

Detection of Overlapping Communities in Social Network

Sumana Maity



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India
June 2014

Detection of Overlapping Communities in Social Network

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Technology

in

Computer Science and Engineering

(Specialization: Software Engineering)

by

Sumana Maity

(Roll No.- 212CS3374)

under the supervision of

Prof. Santanu Kumar Rath



Department of Computer Science and Engineering

National Institute of Technology Rourkela

Rourkela-769 008, Odisha, India

June 2014



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

Certificate

This is to certify that the work in the thesis entitled *Detection of Overlapping Communities in Social Network* by *Sumana Maity* is a record of an original research work carried out by him under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Master of Technology with the specialization of Software Engineering in the department of Computer Science and Engineering, National Institute of Technology Rourkela. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Place: NIT Rourkela
Date: June 1, 2014

(Prof. Santanu Ku. Rath)
Professor, CSE Department
NIT Rourkela, Odisha

Acknowledgments

I am grateful to many people who have helped towards shaping this thesis. It gives me immense pleasure and satisfaction to express my heart-felt gratitude to my guide, Prof. Santanu Ku. Rath, Head-CSE, for accepting me as his project student and providing me with excellent guidance and constant encouragement throughout my project duration. He devoted his valuable time towards discussions, and offered viewpoints and insights which went far beyond the narrow domain of work and helped me embark on new ideas. I am very much grateful to him for his invaluable suggestions, able guidance, during this period and above all constant encouragement throughout my work.

I would like to express my sincere thanks to all my teachers for providing sound knowledge base and cooperation.

I would like to thank all my classmates and friends for helping me in one or other way. I should not forget to thank my lab-mates; Jyoti, Prerna, Amar, Lov and my lab seniors Mukesh Kumar, Y. Suresh, Shashank M. Satpaty, Ashis Dwivedi and Abinash Tripathy for their endless support and enthusiasm. Without them this work would not be successful.

I must acknowledge the academic resources that I have got from NIT Rourkela. I would like to thank administrative and technical staff members of the Department who have been kind enough to advise and help in their respective roles.

Last, but not the least, I would like to dedicate this thesis to my family, for their love, patience, and understanding.

Sumana Maity

Roll-212cs3374

Abstract

Community detection in a social network is an emerging issue in the study of network system as it helps to realize the overall network structure in depth. Communities are the natural partition of network nodes into subgroups where nodes within the subgroup are densely connected but between the subgroups connections are sparser. Real world networks, including social networks have been found to partition themselves naturally into communities. A member of a social network can be part of more than one group or community. As a member of a social network can be overlapped between more than one group, overlapping community detection technique need to be considered in order to identify the overlapping nodes. This topic of research has many applications in various fields like biology, social sciences, physics etc.

In literature, most of the proposed community detection approaches are able to detect only disjoint communities. Recently few algorithms has been emerged which are capable of discovering overlapping communities. In this work two different types of algorithms have been proposed which efficiently detect overlapping communities. A novel approach has been introduced which overcomes the short-falls of clique percolation method, an overlapping community detection algorithm mostly used in this area. Another algorithm which is based on Genetic Algorithm is also used to discover overlapping communities. Modularity measure is generally used to determine the quality of communities for the particular network. The Quality of the communities detected by the algorithms is measured by several different overlapping modularity measures. Standard real world networks used as benchmark for community detection, have been used to judge the algorithms.

Keywords: Social Network, Overlapping Community Detection, Graph Theory, Clique, Modularity Measure, Genetic Algorithm.

Contents

Certificate	ii
Acknowledgement	iii
Abstract	iv
List of Figures	viii
List of Tables	ix
1 Introduction	1
1.1 Literature survey	2
1.2 Motivation of Work	4
1.3 Objectives of Work	5
1.4 Experimental Datasets	5
1.4.1 Zachary’s Karate Club Network	5
1.4.2 Dolphin Social Network	6
1.4.3 Les-Miserables	6
1.5 Thesis Organization	6
2 Related Concepts	7
2.1 Network	7
2.2 Adjacency Matrix	8
2.3 Degree of a Node	8
2.4 Average Degree	8
2.5 Degree Distribution:	9
2.6 Small World	9
2.7 Component	9

2.8	Community	10
2.9	Community Structure	10
2.10	Overlapped and Non-Overlapped node	11
2.11	Modularity Measure	11
2.11.1	Modularity by Girvan and Newman	11
2.11.2	Modularity Measure by Nepsuz et al.	13
2.11.3	Modularity Measure by Chen et al.	14
2.11.4	Modularity by Shen et al.	14
2.11.5	Partition Density by Ahn et al.	15
2.11.6	Modularity by Lazar et al.	16
2.11.7	Modified Version of Lazar et al.	17
2.12	Summary	18
3	Overlapping Community Detection Based on Fuzzy Assignment of Nodes	19
3.1	Introduction	19
3.2	Research background	20
3.2.1	Clique and k-clique	20
3.2.2	Clique Percolation Method (CPM)	21
3.2.3	Community Similarity	22
3.2.4	Belonging Coefficient	23
3.3	Proposed Method: Extended Clique Percolation Method (ECPM) .	23
3.4	Implementation	28
3.4.1	Preprocessing of Network	28
3.4.2	Implementation	28
3.5	Result and Analysis	29
3.5.1	Zachary's Karate Club Network	29
3.5.2	Dolphin Social Network	32
3.5.3	Les-Miserable Network	33
3.6	Summary	34

4	Overlapping Community Detection Using Genetic Algorithm	35
4.1	Introduction	35
4.2	Research Background	35
4.2.1	Adjacent Edge	35
4.2.2	Line Graph	36
4.3	Genetic Algorithm for Community Detection	37
4.3.1	Objective Function	37
4.3.2	Genetic Representation	37
4.3.3	Operators	39
4.3.4	Algorithmic Description	41
4.4	Result and Analysis	42
4.4.1	Zachary's Karate Club Network	42
4.4.2	Dolphin Social Network	43
4.4.3	Les-Miserable Network	44
4.4.4	Comparison with CPM and ECPM	44
4.5	Summary	47
5	Conclusion and Future Work	48
5.1	Conclusion	48
5.2	Future Work	49
	Bibliography	50

List of Figures

2.1	Community Structure in a network	10
3.1	An example network	20
3.2	Corresponding Clique Graph of example network	22
3.3	Community Structure detected by CPM with $k=4$	24
3.4	Modularity values for karate club network	30
3.5	Communities detected using CPM in Karate Club Network	31
3.6	Communities detected using proposed method in Karate Club Network	31
3.7	Modularity values for dolphin network	33
3.8	Modularity values for Les-Miserable network	34
4.1	An example graph, G	36
4.2	Line Graph, $L(G)$ corresponding to graph, G	36
4.3	Components to corresponding individual	39
4.4	Communities in Graph G	39
4.5	Communities detected using Genetic Algorithm in Karate Club Network	43
4.6	Comparison of modularity values for karate club network	45
4.7	Comparison of modularity values for dolphin network	45
4.8	Comparison of modularity values for les-miserable network	46

List of Tables

2.1	Overlapping Modularity Measures Summarized	18
3.1	Belonging Coefficients to initial communities	27
3.2	Belonging Coefficients to updated communities	27
3.3	Community Structure Details of Karate Club Network	30
3.4	Community Structure Details of Dolphins Social Network	32
3.5	Community Structure Details of Les-Miserable Network	33
4.1	Community Structure details of Karate Club Network, GA	42
4.2	Community Structure details of Dolphin Social Network, GA	43
4.3	Community Structure details of Les-Miserable Network, GA	44

Chapter 1

Introduction

Real world complex systems can be represented in the form of networks. To understand the in-depth structure and detail function of those systems, it is important to study and analyze the networks [1–3]. A trivial property of these networks is community structure obtained by partitioning the network into several groups, within which connection between nodes are more dense than the rest of the network. This type of grouping is commonly referred as communities, but also known as clusters, cohesive groups, or modules as there is no globally accepted unique definition [1]. The concept of community detection is related to graph partitioning in some way; though it is very much dissimilar from graph partitioning. In case of graph partitioning, number of groups and the approximate size of those groups are known priory and the task is usually to divide the network into these many numbers of disjoint sub-graphs of almost same size, irrespective of whether a partition even exists. But in case of community detection, it is not known that how many communities are present in the network and it is not at all mandatory for them to be of same size. The community detection approach assumes that most of real world networks, divide naturally into groups of nodes (community) with dense connections internally and sparser connections between groups, and the experimenter’s job is only to detect these already formed groups [12]. The number of partitions and size of them are settled by the network itself and not set by the experimenter. So community detection is the technique which aims at discovering natural divisions of (social) networks into groups based on strength of

connection between vertices's [1, 4].

Basically, community can be subdivided into two types; disjoint communities and overlapping communities. In disjoint communities nodes can be part of only a single community, but in overlapping communities partitions are not necessarily disjoint. There could be nodes that belong to more than one community [5–7].

A social network is a collection of finite set of members (nodes) which can be a single person, a group, an organization; and relations (edges) among them may represent friendship, influence, affection or conversely, dislike, conflict or many other similar entities. In a social network, a community could be a group of people with common interest or location [8, 9]. Generally in any social network a person may be part of more than one different group or community, like a person can be part of his/her professional group and simultaneously can be part of his/her family group indicating overlap between the professional and family group. So for social networks, overlapping community detection technique should be considered over disjoint community detection technique.

1.1 Literature survey

Community detection is a stimulating field of research. There are various community detection algorithms available but most of the algorithms are able to detect disjoint communities only. As overlapping community detection is comparatively new approach less algorithms are present for this approach. Some of these work are described below.

In 2004, Newman [1] proposed a disjoint community detection algorithm. Communities are found using edge betweenness. In this work first modularity measure is introduced. In the year of 2005, Clauset et al. [10] proposed another community detection algorithm based on a local modularity measure proposed by them. At each step the algorithm adds a node into a partial community and update the neighbors of that community. It discovers only disjoint communities. Palla et al. [11] presented the first overlapping community detection algorithm in 2005. In

this approach communities are identified based on the adjacent cliques. This algorithm allows a node to be part of more than one community, resulting overlapping community structure. In 2006, Newman proposed another community detection algorithm [12]. The algorithm works by using eigenvector of matrices. Here he has proposed a new quality function, modularity matrix which has been used to detect the community structure. This algorithm results disjoint communities. Nepsuz et al. [13] in 2008 proposed a new approach to evaluate modularity for overlapping communities. Lancichinetti et al. [7] in 2009 have proposed an algorithm which works based on local optimization of a fitness function. This algorithm finds overlapping communities by maximizing the fitness value. Shen et al., 2009 [2] proposed another overlapping community detection approach. Overlapping communities are detected based on maximal cliques. An overlapping modularity measure is also proposed here based on number of maximal cliques. Gregory, 2009 [14], proposed a two phase method for overlapping community detection. In the first phase of this method, the network is transformed into a new one by splitting the nodes using split betweenness. In the second phase of the method, a disjoint community detection algorithm has been applied to process the transformed network. Ahn et al., 2010 [15] presented another overlapping community detection algorithm based on link partition. Using hierarchical clustering links are partitioned to link dendrogram. Overlapping communities are detected by cutting this dendrogram at some threshold point. Here one more modularity measure partition density is introduced. Chen et al., 2010 [16] proposed another algorithm to detect overlapping communities in weighted network. It detects overlapping communities using a local algorithm which works by expanding a partial community which is started from a special single node. They have introduced another overlapping modularity measure. Lazar et al. 2010 [17] proposed a overlapping modularity measure for overlapping communities based on difference between inward and outward edges. Nguyen et al., 2011 [18] proposed a two-phase framework which detect the overlapping community structure in a dynamic network by quickly and adaptively updating the network structure only based on its history without recomputing

from scratch. Coscia et al., 2012 [19] presented an overlapping community discovery algorithm. Each node vote for the communities it sees surrounding it using a label propagation algorithm and finally, the local communities are merged into a global collection. Tooth et al., 2013 [20] studied various modularity measures by applying them on community structures obtained using clique percolation method.

Those are some of graph theory based overlapping community detection approaches. Few author has attempted to detect overlapping communities using genetic algorithm by maximizing the modularity value. Some of these type of approach are described here. Clara Pizzuti, 2008 [21] proposed an genetic algorithm based disjoint community detection approach which uses node-based clustering. A simplified objective function is also proposed here. Clara Pizzuti, 2009 [22] has proposed first genetic algorithm based overlapping communities detection algorithm. It uses edge-clustering approach instead of node-clustering. They have used line graph concept to achieve that. Cai et al., 2011 [23] proposed another genetic algorithm based approach to detect overlapping communities. The algorithm first finds the link communities by optimizing objective function partition density, and then map the link communities to overlapping node communities. They have used the concept of bridge node to adjust the node membership of overlapped nodes. Dickinson et al., 2013 [24] also attempted to detect overlapping communities using genetic algorithm. Overlapping communities are detected using two different approach Label Rank algorithm and genetic algorithm. For genetic algorithm edge based clustering has been used and Modularity by Shen et al. [2] is used as the objective function.

1.2 Motivation of Work

The ability to find and analyze communities present in the network can provide help in understanding and visualizing the structure of networks. Social creatures interact in diverse ways by forming groups, sending messages, sharing items, joining in group discussion etc. Some of the interactions are accidental while others

are a consequence of the underlying explicit or implicit social structures. In order to understand social interactions and the low-level structure of the network, it is therefore crucial to identify these social structures or communities. The well-known structure of the network can have significant applications. For effective online marketing, such as placing online ads or deploying viral marketing strategies, known community structure in social network could often lead to more accurate targeting and better marketing results.

1.3 Objectives of Work

The objective of this work is to effectively explore and identify the naturally formed communities in a network so that the in-depth structure of the network becomes understandable. Diverse overlapping community detection algorithms need to be studied and implemented to investigate for the effective and better approach for this purpose. To identify the best and effective approach, they need to be evaluated by some means. Various overlapping modularity measures are needed to be considered to measure effectiveness of the approaches.

1.4 Experimental Datasets

Standard real world networks generally considered as benchmark for community detection, are used as experimental dataset for all the implemented algorithms.

1.4.1 Zachary's Karate Club Network

This is a social network of friendships between 34 members of a karate club at an American university in the 1970. It is an undirected, unweighted network having 34 nodes and 78 edges. Due to some reason their was a disagreement between the administrator and instructor and the network was divided into two groups. Node 1 and 33 represents the administrator and instructor respectively [25].

1.4.2 Dolphin Social Network

It is a relation-network between bottlenose dolphins. It describes the associations between 62 dolphins living in Doubtful Sound, New Zealand, with ties between dolphin pairs. This Network was built by Lusseau et al [26]. It is an undirected network consisting 62 nodes and 159 edges .

1.4.3 Les-Miserables

This is a weighted network of co-appearances of characters in Victor Hugo's novel "Les Miserables". Nodes represent characters and edges connect any pair of characters that appear in the same chapter of the book. The weights on the edges are the number of such co-appearances. The data on co-appearances were taken from [27]. In this work as only unweighted and undirected links are considered, here links are considered as un-directed. Network consist of 77 nodes and 254 edges.

1.5 Thesis Organization

The rest of the thesis is organized as follows.

In Chapter 2 related concepts have been discussed which are used later in this work. Chapter 3 describes the proposed overlapping community detection algorithm based on fuzzy assignments of nodes. In Chapter 4 another overlapping community detection algorithm, based on genetic algorithm has been discussed. Finally, Chapter 5 concludes the thesis with future work.

Chapter 2

Related Concepts

In this chapter fundamental concepts related to overlapping community detection are described which have been used throughout the thesis.

2.1 Network

A network represents the actors and relations among them. A network is generally presented as a graph $G(V, E)$, where V is set of n nodes and E is set of m edges. Actors in the network are called as nodes, vertex and relations can be called as edge, arc, link, tie, bond or connections.

The nodes and edges in the network may have different properties depending on the concerned system. For example edges or connections can be weighted, directed or undirected and nodes can have a size, color or many other attributes generally termed as labels. For example, in a social network, one might need to consider the gender or age of the considered population. In the corresponding network, this translates into labels associated to each node. Links labels are more common, and generally take the form of numerical values called weights, expressing the strength of the corresponding to relationships. Another important link attribute is the direction, which introduces a distinction between the connected nodes: one is the tail, the other is the head, and the link represents an asymmetrical relationship from the tail towards the head. Depending on the links, a network is said to be (un)weighted and (un)directed. In this work networks, with attribute-less nodes and unweighted, undirected links has been considered.

2.2 Adjacency Matrix

The information about a network can be represented by some mathematical expressions, vectors, matrices etc. One of the most common way of network representation is adjacency matrix. It describes how the nodes are connected.

$$A_{ij} = \begin{cases} 1 & \text{if a link exists between node } i \text{ and node } j \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

If the network is weighted then the weights are used in place of binary values.

2.3 Degree of a Node

Degree of a node is the number of connections, a node is having with other nodes. In another words, number of neighbors a node is directly associated with. For a directed network a node has two type of degree which are in-degree and out-degree. These are distinguished by the direction of links. In-degree of a node is the number of incoming links where out-degree is the count of outgoing links. Nodes with no links that is of zero degree are called as isolated nodes [28].

2.4 Average Degree

Average degree of a network is the mean degree processed over all of the nodes of the network. It depends on the considered system, the number of links and nodes. The average degree of an undirected, unweighted network can be calculated as

$$d_{avg} = \frac{1}{n} \sum_i d_i \quad (2.2)$$

Here, d_{avg} is average degree of network, d_i symbolizes the i^{th} node's degree and n is the total node number.

2.5 Degree Distribution:

A significant property of real world network is power law degree distribution. The degree distribution, $p(k)$ of a considered network is defined as the fraction of nodes in the network having degree of k . If there are n nodes in total in a network and n_k of them have degree k , then $p(k) = n_k/n$. The networks which follows power-law degree distribution is known as scale free network. It has been seen that degree distribution of real world network follows power law [28,29].

2.6 Small World

If the average path length between two nodes is small then the considered network is having small world property. The famous experiment of Milgram's named as, "six degrees of separation" [28] presents the idea, that any two person in earth is at most six steps away from each other. A chain of, "a friend of a friend" can be made which will be able to connect any two person in six steps or less distance. The empirical studies are shown that many real networks has small world property.

2.7 Component

A component is a sub-network in which any node is reachable from any other node by a walk. Putting concisely, it is a maximal connected subgraph. For an undirected network, a component is a set of connected nodes with no links with other nodes from the same network. But for directed networks, it is less straightforward. A component is said to be strongly connected if there is a directed walk between each pair of nodes It is called weakly connected if there is at least an undirected walk between each pair of nodes. A network with only one component is said to be connected. An isolated node (i.e. a node with a degree zero) is a component of its own. [30,31]

2.8 Community

A community is a group of actors or nodes, and connections or links between these nodes where nodes are clustered into tightly knit groups with high density of within-groups edges and low density of between-group edges [32]. Communities have been studied in diverse domains which results several names to refer them like modules, partitions, clusters or cohesive subgroups. In Figure 2.1, nodes 1, 2 and 3 combinedly represents a community. Again Nodes 3, 4 and 5 represents another community.

Basically communities can be subdivided into two types; disjoint communities and overlapping communities. In disjoint communities nodes can be part of only single community but in overlapping communities partitions are not bounded to be disjoint. There could be nodes that belong to multiple communities [5–7].

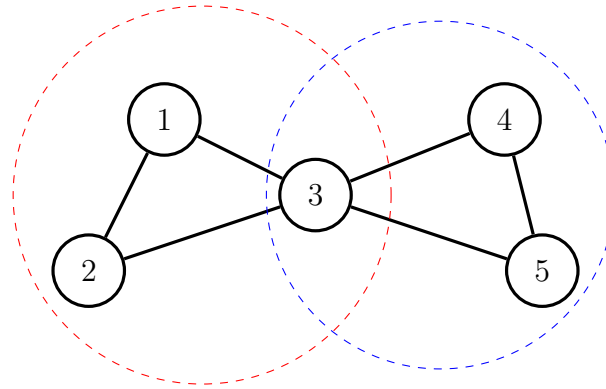


Figure 2.1: Community Structure in a network

2.9 Community Structure

Set of communities discovered in a network combinedly referred as the community structure or cover, represented as $C = \{c_1, c_2, c_3, \dots, c_k\}$. Here C is the cover or community structure and c_1, c_2, c_k are communities. The size of cover represented by $|C|$ indicate number of communities in the cover. For example in Figure 2.1 communities are $c_1 = \{1, 2, 3\}$ and $c_2 = \{3, 4, 5\}$ and cover $|C| = \{c_1, c_2\}$.

2.10 Overlapped and Non-Overlapped node

An overlapped node is shared by more than one community. Type of community structure where some nodes are overlapped is known as overlapping community structure. For example in Figure 2.1 node 3 is an overlapped node.

An non-overlapped node belongs to only one community. Type of community structure where all nodes are non-overlapped is known as non-overlapping or disjoint community structure.

2.11 Modularity Measure

Actual community structure or cover in a network is not always fixed. It corresponds to the arrangement of edges. Modularity is a quantitative measure of the quality of the community structure or inversely it defines how good the founded community structure is [4]. It can be either positive or negative, where positive value indicates possible presence of community structure [1,4].

Presently most commonly used modularity measure for overlapping community structure is Normalized Mutual Information (NMI) [5]. The problem with this modularity measure is on ground truth about the network which needs to be known priori. But it is nearly not possible for real world networks. Recently few modularity measures for overlapping community structure have been developed which does not require the ground truth information [20]. In most of the papers researchers have considered a single modularity measure to rate the community structure. Every modularity measure has its own strength and weakness. It will not be fair enough to declare a single modularity measure as ideal one; as there is no such universal modularity. In this work a number of modularity measures have been considered for quality measure.

2.11.1 Modularity by Girvan and Newman

The concept of modularity is proposed by Girvan and Newman [4]. According to them a good community division in network is not that where there are fewer edges

between the communities; it is where there are more edges within communities than between the communities [4]. Modularity is defined as the difference between number of edges falling inside the communities of the network and the number of edges expected to fall inside the communities in an equivalent random network. The reason of this definition is that number of connections inside the community should be larger than the what is expected for a random network. So the algorithm works by comparing the fraction of connections or edges inside the communities to the expected fraction of edges in a random graph where the degrees of individual nodes are equal to the degrees of nodes in the original network [4, 20].

In a random graph having equal number of edges, m of original network, the probability for having a connection between two nodes i and j with degrees d_i and d_j respectively is $d_i d_j / 4m^2$. Accordingly, the expected fraction of edges inside a community c is $(d_c / 2m)^2$, where $d_c = \sum_{i \in c} d_i$ is the sum of node degrees in community c .

As modularity is measured by comparing fraction of edges inside the community with expected fraction of edges in random model, the modularity contribution by community c is given in Eq. (2.3).

$$Q_c = \frac{l_c}{m} - \left(\frac{d_c}{2m} \right)^2 \quad (2.3)$$

where, m is total number of edges in the network and l_c is total number of edges within the community c . So l_c/m is fraction of edges or connections in community c . Based on the above, the modularity of the cover can be computed as the sum of contribution for all communities, as mentioned below:

$$Q = \sum_{c \in C} Q_c = \sum_{c \in C} \left[\frac{l_c}{m} - \left(\frac{d_c}{2m} \right)^2 \right] \quad (2.4)$$

Eq. (2.4) can also be written in a different form with summation over the individual nodes as:

$$Q = \frac{1}{2m} \sum_{c \in C} \sum_{ij} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \delta_{ic} \delta_{jc} \quad (2.5)$$

where, A_{ij} stands for the corresponding value in adjacency matrix. δ_{ic} indicates whether the node i belongs to community c or not. $\delta_{ic} = 1$, if node i is part of community c and $\delta_{ic} = 0$, when it is not.

The modularity measure suggested by Newman and Girvan may be considered as appropriate for disjoint communities as, the value of δ_{ic} in Eq. (2.5) can only be 1 or 0, which indicates a node can be part of a single community or none. But in case of overlapping communities a node may belong to more than one community. So δ_{ic} needs to be evaluated in a way so that it can measure how much fraction of node i is dedicated to community c . Recently few modularity measures are proposed which takes this overlapping case into consideration [2, 15–17, 20, 33].

2.11.2 Modularity Measure by Nepsuz et al.

Modularity measure proposed by Nepsuz et al. [13] generalizes the concept of Newman and Girvan for overlapping communities by proposing a fuzzy modularity measure, where a node can be member of several communities. This type of assignment of nodes are called fuzzy assignment. If a node is member of more than one community, then it is distributed among those communities.

In this type of assignment, each node has a total membership degree of 1. If a node belongs to a single community, it's membership for that community is 1 and if it belongs to more than one community, it's membership value is divided among those communities. Maximum membership value of a particular node to a community can be 1, if it belongs to only that community and minimum membership value to a community can be 0, if it is not at all a part of that community. If a node belong to more than one community, then it will be having membership values between 1 and 0 to those communities it belongs to. If α_{ic} represents the membership value of node i to community c , then constraints for α_{ic} can be:

$$0 \leq \alpha_{ic} \leq 1, \quad \forall c \in C, i \in V \quad (2.6)$$

and

$$\sum_{c \in C} \alpha_{ic} = 1 \quad (2.7)$$

For disjoint community structure, value of α_{ic} will be 0 or 1. So this α_{ic} is counter part of δ_{ic} in Eq. (2.5) for overlapping community structure, which indicates that how much fraction of node i is included in community c . So the modularity for overlapping community structure is evaluated as:

$$Q = \frac{1}{2m} \sum_{c \in C} \sum_{ij} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \alpha_{ic} \alpha_{jc} \quad (2.8)$$

The idea proposed by Nepsuz et al. [13] to calculate α_{ic} is to divide the membership values of a node equally among communities the node is part of, independently of the underlying network as per following equation

$$\alpha_{ic} = \frac{1}{q_i} \quad (2.9)$$

where, q_i denotes the number of communities, node i is associated with.

2.11.3 Modularity Measure by Chen et al.

One more approach is proposed by Chen et al. [16] to compute the membership value α_{ic} . In this approach, connections within the communities i.e., between community members and connections between the communities are also considered to compute α_{ic} which is calculated as,

$$\alpha_{ic} = \frac{k_{ic}}{\sum_{c \in C} k_{ic}} \quad (2.10)$$

where, $k_{ic} = \sum_{j \in c} A_{ij}$ and A_{ij} is the value in adjacency matrix corresponding to node i and j . Modularity is calculated using Eq. (2.8) where α_{ic} is computed using Eq. (2.10).

2.11.4 Modularity by Shen et al.

An approach to calculate α_{ic} is proposed by Shen et al. [2]. They have considered maximal cliques present in the network to compute α_{ic} . Since maximal cliques

are highly connected, if two community shares a maximal clique then they must represent one community. So it is assumed that each and every maximal clique is part of only one community. Here α_{ic} is calculated considering the maximal cliques as follows,

$$\alpha_{ic} = \frac{1}{\alpha_i} \sum_{j \in V} \frac{O_{vw}^c}{O_{vw}} A_{ij} \quad (2.11)$$

with,

$$\alpha_i = \sum_{c \in C} \sum_{j \in V} \frac{O_{vw}^c}{O_{vw}} A_{ij} \quad (2.12)$$

where, O_{vw} denotes total number of maximal cliques containing edge (i, j) and O_{vw}^c denotes number of maximal cliques present within the community c and also contains edge (i, j) . A_{ij} is adjacency matrix value. The term α_i is used for the normalization purpose, so that α_{ic} satisfies the constraints described in Eq. (2.6) and Eq. (2.7)

2.11.5 Partition Density by Ahn et al.

Another modularity measure for overlapping communities has been recommended by Ahn et al. [15]. It is founded on link density of the communities. A network consisting m links (edges), n nodes with a cover, C consisting of $|C|$ communities is division or partition of network's links into $|C|$ groups. Link or partition density of community c is calculated as:

$$D_c = \frac{l_c - (n_c - 1)}{n_c(n_c - 1)/2 - (n_c - 1)} \quad (2.13)$$

where l_c and n_c denotes number of nodes and edges within the community c respectively.

The link or partition density, D for the entire network is calculated as average of all D_c , weighted by the fraction of edges inside the communities as

$$D = \sum_{c \in C} \frac{l_c}{m} D_c \quad (2.14)$$

2.11.6 Modularity by Lazar et al.

One more overlapping modularity measure has been proposed by Lazar et al. [17]. According to them in a well structured cover, members of a community should dedicate most of their links to the community they are part of instead of dedicating more of it's links to other communities they are not part of [17]. And communities are expected to be dense i.e., they should have more connections within themselves than between themselves. [11, 17]. So necessary conditions of a good community structure are following:

1. Links of a node must primarily go inward to the community, it is part of.
2. Communities need to be dense.

The first condition tells how excusable it is to assign the node $i(\in c)$ to the community c . It is the difference between the edges going inward to the community and the edges going outward from the community, associated with node i divided by the node degree, d_i . So every node i contributes to the community c with which it is associated by following value:

$$\frac{\sum_{j \in c, i \neq j} A_{ij} - \sum_{j \notin c} A_{ij}}{d_i} \quad (2.15)$$

where, A_{ij} denotes the adjacency matrix value. An overlapped node contribute with positive values to more than one communities. So because of overlapping areas the whole network's modularity value increases [17].

For the second condition; communities need to be dense, density of each community or module is interpreted as $\frac{l_c}{\binom{n_c}{2}}$. Where, l_c is number of edges and n_c is number of nodes within the community c . So $\binom{n_c}{2}$ indicates maximum number of edges, possible with n_c nodes.

To avoid the problem of resulting cover with only a few communities with high modularity values, one more criteria has been added. The criteria is that each and every node must belong to at least one community. To fulfill this criteria all left-out nodes together are considered as a separate community at the end [17].

In case of overlapping cover, nodes may belong to multiple communities. So Eq. (2.15) need to be divided by the number of communities node i is associated with, i.e., q_i [17]. So, the modularity of a community, c is measured by the average of Eq. (2.15) over the community members, multiplied by the link density inside c as given in equation below:

$$Q_c = \left[\frac{1}{n_c} \sum_{i \in c} \frac{\sum_{j \in c} A_{ij} - \sum_{j \notin c} A_{ij}}{d_i \cdot q_i} \right] \frac{l_c}{\binom{n_c}{2}} \quad (2.16)$$

The overall modularity of a cover is the average of Q_c over all the communities in the cover as given by Eq. (2.17). Where $|C|$ is number of communities consisted in cover C .

$$Q = \frac{1}{|C|} \sum_{c \in C} Q_c \quad (2.17)$$

2.11.7 Modified Version of Lazar et al.

The modularity measure proposed by Lazar et al. described above has a drawback which is eliminated by Toth et al. [20] by proposing a slight modification of the above approach. The drawback is that it does not consider the size of the communities. All communities are treated equally irrespective of their different sizes. This may create problem when a huge size community rises propagating the entire network. The individual modularity for the huge size community given in Eq. (2.16) will be surely low, as the link density will not be higher than the overall link density in the network [20]. But modularity value should not be low because of the less contribution from the huge sized community irrespective of presence of few small good quality communities. To prevent this another version of Q in Eq. (2.17) is proposed by Toth et al. [20]. Rather than treating all communities equally, they are weighted by the fraction of links in the community, as given by:

$$\hat{Q} = \sum_{c \in C} \frac{l_c}{m} Q_c \quad (2.18)$$

2.12 Summary

In this chapter few concepts related to community detection have been discussed. These concepts have been used throughout the thesis. Overlapping modularity measures addressed here are used rigorously to evaluate the quality of community structure or cover. Considered overlapping modularity measures are summarized in Table 2.1 with the abbreviation used in later chapters.

Table 2.1: Overlapping Modularity Measures Summarized

NP	Modularity proposed by nepsuz et al. [13] given in Eq. (2.8), where membership value, α_{ic} is assessed according to Eq. (2.9)
CHN	Modularity by chen et al. [16] given in Eq. (2.8). α_{ic} is computed using Eq. (2.10)
SHN	Modularity proposed by shen et al. [2] in Eq. (2.8). Here α_{ic} is computed using Eq. (2.11)
AHN	partition density proposed by ahn et al. [15] given in Eq. (2.14)
LZR	overlapping modularity measure proposed by Lazar et al. [17] given in Eq. (2.17)
MLZR	Modularity proposed by Lazar et al. [17] and modified by Toth et al. [20] given in Eq. (2.18)

Chapter 3

Overlapping Community Detection Based on Fuzzy Assignment of Nodes

3.1 Introduction

In general, communities can be divided into two types based on the assignments of nodes into communities. Assignments of nodes can be crisp (non-fuzzy) assignment or fuzzy assignment [14]. In case of crisp assignment, binary relationship is being held between a node and a community. A node can belong to at-most 1 community and at-least 0 community (none). But in case of fuzzy assignment, a node may belong to more than one community. A node associated with a community by some factor. Type of community structure where the assignment of nodes is fuzzy is known as overlapping community structure.

Members in social network are usually part of multiple communities. For example a person is supposed to have connections to several communities like his/her family, friends, office colleagues etc. So in social network an individual may belong to unlimited number of communities as a person can simultaneously be a member of as many groups as he/she wants. So when considering social networks, attention should be given to overlapping community detection algorithms which discover a set of clusters that are not required to be disjoint.

Clique percolation method (CPM) is one of the first and most commonly used algorithm for overlapping community detection. This is a simple and efficient

algorithm which has been used often in this area. But this algorithm suffers from a disadvantage. In this work a new approach is proposed which extends CPM to cover its disadvantage.

3.2 Research background

In this section, concepts related to the proposed work are described.

3.2.1 Clique and k-clique

Clique in a graph is a subset of nodes where each pair of node is connected through an edge, that is a complete sub-graph. Finding all cliques with a given size in a graph, is an NP-hard problem [34].

The notation, *k-clique* used in the community detection technique is completely different from the *k-distance* clique in graph theory. Here *k-clique* indicates size of the clique i.e., the clique consist of *k* nodes e.g. a 3-clique indicate a complete sub-graphs having 3 nodes [34].

Figure 3.1 shows an example network having 3-clique and 4-clique. Six *3-cliques* are (1, 2, 3), (1, 2, 8), (2, 4, 5), (2, 4, 6), (2, 5, 6) and (4, 5, 6) and one *4-clique* is (2, 4, 5, 6).

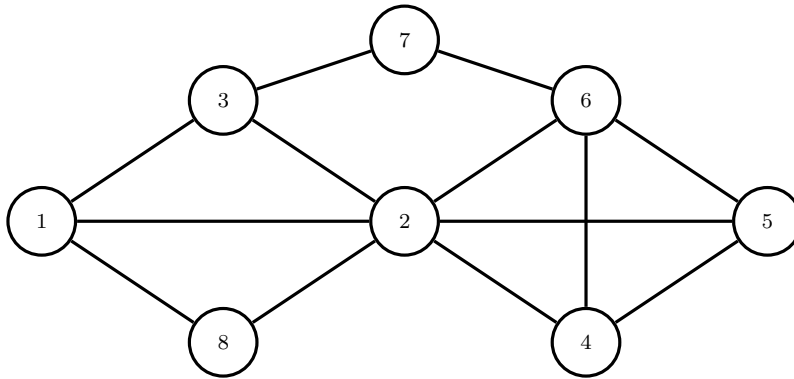


Figure 3.1: An example network

3.2.2 Clique Percolation Method (CPM)

CPM is a clique based overlapping community detection algorithm used mostly in this field. [11].

As the connection between the nodes within the community is dense, it is obvious that edges within a community form cliques (complete subgraph) due to their high density. But it is unlikely that edges between communities i.e., inter-community edges form cliques. The assumption based on which clique percolation method works is that a community comprises of overlapping sets of fully connected subgraphs. So this algorithm detect communities by searching for adjacent cliques [11]. It begins by exploring all the k -cliques (clique of size k) in the network. When all the k -cliques have been found a new graph commonly referred as clique-graph is constructed where each vertex represents a k -clique. Two nodes in this clique-graph is connected or adjacent if they share $(k - 1)$ members. Each connected component in the clique-graph represents a community [11]. The overall process of the approach is described below in Algorithm 1.

Algorithm 1 Clique Percolation Method

Input: The network, G and the clique size, k

Output: Community structure, C

Step 1: All k -cliques present in the network G are identified.

Step 2: A new network, referred as clique-graph, G_C is formed where each node represents an identified clique and two nodes (clique) in the network, G_C are connected by an edge, if they share $k - 1$ members.

Step 3: Connected components in G_C are identified.

Step 4: Each connected component in G_C represents a community. Set of communities forms the identified community structure for the network, G .

For example, the network shown in Figure 3.1 have six 3-cliques which represents individual node in clique graph. Six cliques are,

$a:(1, 2, 3)$; $b:(1, 2, 8)$; $c:(2, 4, 5)$; $d:(2, 4, 6)$; $e:(2, 5, 6)$; $f:(4, 5, 6)$

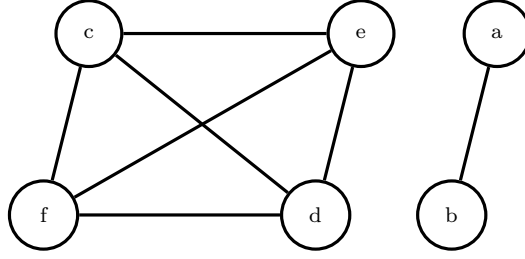


Figure 3.2: Corresponding Clique Graph of example network

If two cliques share minimum of 2 nodes (as $k=3$ here) then they are connected by an edge. Clique a and b share two nodes (node 1 and 2). So these two clique nodes will be connected through an edge. In the same way other cliques also make connection with each other to form the clique graph, G_C as shown in Figure 3.2. Connected components in G_C are (a, b) and (c, d, e, f) . Connected components represent communities. So, in this case two connected components correspond to two communities which are,

$$c_1 : (1, 2, 3, 8)$$

$$c_2 : (2, 4, 5, 6)$$

The *community structure* or *cover* is $C = \{c_1, c_2\}$ and node 2 is overlapped between these two communities. Node 7 is not included to any community as it is not a part of any 3-cliques.

3.2.3 Community Similarity

Community similarity defines how much similar are two communities. There are various ways to measure similarity between two entities. In this paper Jaccard index has been considered as similarity measure [32]. If c_i and c_j are two communities in community structure C then *similarity* between these two communities is defined as,

$$s = \frac{|c_i \cap c_j|}{|c_i \cap c_j| + |c_i - c_j| + |c_j - c_i|} \quad (3.1)$$

$|c_i \cap c_j|$ indicate number of common members i.e., members present in both the communities. $|c_i - c_j|$ and $|c_j - c_i|$ indicate the numbers of members present only in one or the other community.

3.2.4 Belonging Coefficient

This measure defines how strongly a node is associated with a community. For a community c and a node i , *belonging coefficient* is measured as

$$B(i, c) = \frac{\sum_{i \in c} A_{ij}}{k_i} \quad (3.2)$$

with constraint,

$$0 \leq B(i, c) \leq 1$$

Where, k_i is defined as,

$$k_i = \sum_{c \in C} \sum_{j \in c(v)} A_{ij} \quad (3.3)$$

A_{ij} is appropriate element of adjacency matrix. If all neighbor nodes of node i are inside the same community c , then $B(i, c) = 1$ and if none of it's neighbor is included in the community c , then $B(i, c) = 0$.

For example, in Figure 3.3, three communities are (1, 2, 3, 4), (7, 8, 9, 14) and (11, 12, 13, 15) as represented in different colors. Node 6 has one one neighbor node in two communities. So it is having *belonging coefficient* 0.5 to both the communities, (1, 2, 3, 4) and (7, 8, 9, 14). Accordingly node 5 has *belonging coefficient* 1 to communities (1, 2, 3, 4). Node 10 has *belonging coefficient* 0.5, 0.25 and 0.25 to community (1, 2, 3, 4), (7, 8, 9, 14) and (11, 12, 13, 15) respectively.

3.3 Proposed Method: Extended Clique Percolation Method (ECPM)

The k -clique method only considers the fully connected subgraphs of size k . It may happen that a node is associated to a community by some edges, but it does not form any clique of size k . So CPM algorithm leads to a community structure which may not include many member of the network, though they are linked with some communities. The proposed approach aims to cover all the connected members of the network by including them to at least one community.

The approach is based on fuzzy assignments of nodes. A node can take part in multiple communities and is associated with each community by some fraction.

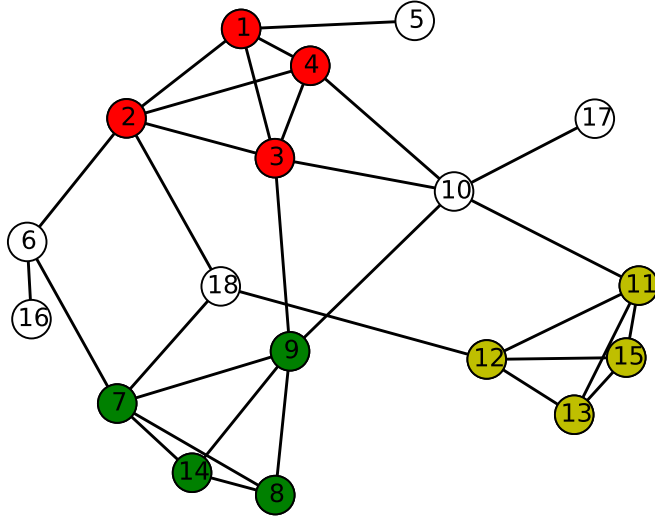


Figure 3.3: An example of network showing three initial communities detected by CPM with $k=4$. Communities are represented by different colors.

This fraction is the *belonging coefficient*. The idea is a node can participate in several communities and is associated with each communities with some value. It is assumed that each node can have maximum attachment value of 1 and minimum attachment value of 0. And it can be attached with a community with an attachment value between 0 to 1. If the node is not part of any community then attachment values of this node to all the communities is 0. If the node is part of few communities then sum of all the attachment values need to be 1. So if there are two communities in a cover and a node is part of both the communities then, a node can have attachment values to two communities as 0.2 and 0.8 as $(0.8+0.2=1)$ but can not be 0.3 and 0.8 (as $0.3+0.8 \neq 1$). Here this attachment value is described as *belonging coefficient*. So one more constraint for *belonging coefficient* is

$$\sum_{c \in C} B(i, c) = 1 \quad (3.4)$$

where c indicates the communities in cover C and i indicate a particular node.

The method proposed here comprises of three major components which are,

- a. **Finding the initial community structure:** The initial communities are detected for networks using CPM for the given value of k , the clique size.
- b. **Updating the initial communities:** If initial community structure includes all the members in the network into some community then there is no need of this step. When some nodes are not included in any of the initial communities because of not forming a clique of size k in the network, this step includes those left out nodes into at least one community. *Belonging coefficient* is used to decide the deserving membership of left out nodes.
- c. **Merging similar communities:** When all nodes are included into some community, it may be possible that two or more communities are very much similar. If two or more communities are similar more than a threshold then these communities are merged into a single one. Similarity is measured using Eq. 3.1.

The overall algorithm has been described in Algorithm 2 where, G is the considered network, k is the clique-size and t is the threshold value for community similarity.

Algorithm 2 Extended Clique Percolation Method

Require: The network G , the clique size, k and the similarity threshold, t

Ensure: Community structure or cover, C

```

1:  $C \leftarrow$  Compute initial cover using CPM for given  $k$ .
2:  $|C| \leftarrow$  Number of initial communities.
3: {Updating communities by adding left out nodes}
4:  $L_f \leftarrow$  Compute list of nodes which does not belong to any initial community.
5: while  $L_f \neq \emptyset$  do
6:   for each  $i$  in  $L_f$  do
7:      $\forall c \in C$ , Find belonging coefficient ,  $B(i, c)$ .
8:     Find the maximum  $B(i, c)$ .
9:     if maximum  $B(i, c) = 0$  then
10:       $N = 0$ 
11:    else
12:       $N \leftarrow$  Count the number of  $B(i, c)$  with maximum value.
13:    end if
14:    if  $N = 1$  then
15:      Include node,  $i$  to the community,  $c$  for which  $B(i, c)$  is maximum and
      remove  $i$  from  $L_f$ . {both  $L_f$  and  $C$  is updated.}
16:    else if  $N > 1$  then
17:      Include  $i$  to all those communities for which  $B(i, c)$  is maximum and
      remove  $i$  from  $L_f$ . { $i$  is overlapped between multiple communities.}
18:    else
19:      do nothing
20:    end if
21:  end for
22: end while
23: {Merging of similar communities}
24: for each two communities  $\in C$  do
25:   Find similarity.
26:   if similarity  $\geq$  threshold,  $t$  then
27:     Merge these two communities and update the cover.
28:   end if
29: end for
30: return  $C$ 

```

For example, in the network shown in Figure 3.3, three communities found initially by CPM with $k = 4$ are

$$c_1 : (1, 2, 3, 4)$$

$$c_2 : (7, 8, 9, 14)$$

$$c_3 : (11, 12, 13, 15)$$

with Cover, $C = \{c_1, c_2, c_3\}$

Left out nodes are $L_f = (5, 6, 10, 16, 17, 18)$

Table 3.1: Belonging Coefficients to initial communities

Node	$B(i, c)$		
	c_1	c_2	c_3
5	1	0	0
6	0.5	0.5	0
10	0.5	0.25	0.25
16	0	0	0
17	0	0	0
18	0.33	0.33	0.33

Belonging coefficient of nodes in L_f to three initial communities are given in Table 3.1. Node 5 having maximum $B(5, c)$ with $c = c_1$ will be included to community c_1 . Node 6 has $B(6, c_1) = 0.5$ and as well as $B(6, c_2) = 0.5$. So node 6 will be included to both c_1 and c_2 . Node 10 shares it's maximum to community c_1 , $B(10, c_1) = 0.5$. So node 10 is included to c_1 . In the same way node 18 is included to all the three communities c_1, c_2 and c_3 as all three communities having same *belonging coefficient* of 0.33. Node 16 and 17 are not included to any community in this step as they are having *belonging coefficient* 0 to all the three communities. In next step the communities are updated as

$$c'_1 : (1, 2, 3, 4, 5, 6, 10, 18)$$

$$c'_2 : (6, 7, 8, 9, 14, 18)$$

$$c'_3 : (11, 12, 13, 15, 18)$$

and left out nodes are $L_f = (16, 17)$

Table 3.2: Belonging Coefficients to updated communities

Node	$B(i, c)$		
	c'_1	c'_2	c'_3
16	0.5	0.5	0
17	1	0	0

In next step again, *belonging coefficients* are computed for *nodes* $\in L_f$ as given in Table 3.2. So node 16 is included to both community c'_1 and c'_2 and node 17 is

included to community c'_1 resulting communities

$$c''_1 : (1, 2, 3, 4, 5, 6, 10, 16, 17, 18)$$

$$c''_2 : (6, 7, 8, 9, 14, 16, 18)$$

$$c''_3 : (11, 12, 13, 15, 18)$$

The final cover, $C = \{c''_1, c''_2, c''_3\}$, with overlapped nodes as 6, 16 and 18. As all the nodes are covered the algorithm stops here. This algorithm ensures that all connected nodes will be included to at least one community.

3.4 Implementation

3.4.1 Preprocessing of Network

In any real-world network there may have some isolated nodes that is members having no connection. Surely these nodes are not going to be part of any community as they are not connected to any one else in the network. In the proposed method all left out nodes have been considered to include in community structure. For isolated nodes *belonging coefficient* will always be zero and they will not be included to any community. The algorithm described in Algorithm 2 will stop only when there will be no left out nodes. The algorithm will never stop if isolated nodes are considered as these nodes will always be in left out list. So isolated nodes are removed from the network before searching for communities in the network.

3.4.2 Implementation

Communities are detected for networks using the proposed approach for different value of k , the clique size. Minimum permitted value for k is 3. In literature it is found that generally value of k ranges from 3 to 6 [5, 20]. Here k value is taken as 3 and 4. Here threshold value, t is taken as 0.8 that is two communities are similar to the extent of 80% will be merged.

NetworkX package and Pyplot library of Python language has been used for network manipulation. implementation of algorithms and visualization.

3.5 Result and Analysis

The method, proposed in this paper has been experimented on three classic real world networks which are used as benchmark networks for community detection [16]. Details of the experimental dataset is given in section 1.4.

3.5.1 Zachary’s Karate Club Network

The method proposed in this chapter is evaluated on this network. Initially with $k = 4$, CPM detects three communities which are (1, 2, 3, 4, 8, 14), (24, 30, 33, 34) and (9, 31, 33, 34) as represented in Figure 3.5 with different colors. Node 33 and 34 are overlapped between last two communities. Total 22 nodes are not included in any community, though they are connected to the network i.e., their friends are part of some community. Proposed method includes these nodes into initial communities on the basis of their *belonging coefficient* as discussed in Algorithm 2. The updated communities are (1, 2, 3, 4, 8, 14, 5, 6, 7, 10, 11, 12, 13, 18, 20, 22, 17), (24, 33, 34, 30, 10, 15, 16, 19, 21, 23, 26, 27, 28, 29, 32, 25) and (33, 34, 9, 31, 10, 15, 16, 19, 21, 23, 29). Here it can be observed that second and third detected communities are very much similar as they share most of the members. Similarity between these two communities is found as 0.81. So these two communities are integrated into a single community as (9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34). Finally according to the proposed method, two communities are found in the network which are (1, 2, 3, 4, 8, 14, 5, 6, 7, 10, 11, 12, 13, 18, 20, 22, 17) and (9, 10, 15, 16, 19, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34). This indicates the actual partition in the network [25]. Details of community structure for both CPM and ECPM is given in Table 3.3. The cover detected by ECPM is presented in Figure 3.6.

With $k = 3$, three communities have been found initially. Node 1 and 32 are found as overlapped nodes. This community structure does not include node 10 and 12 into any community. Proposed method covers these two nodes by including node 10 and 12 to initial communities. Node 10 included as overlapped between

two communities as it shares one connection to both the communities. Proposed method results three communities with three overlapped nodes. Detailed outcome of both the algorithm, CPM and ECPM is given in Table 3.3.

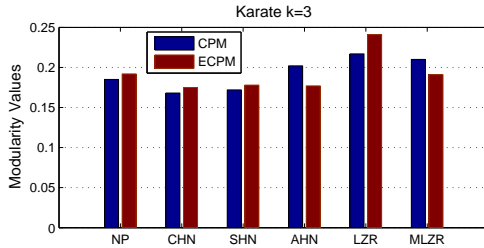
Table 3.3: Community Structure Details of Karate Club Network

k	CN(%)		UNC		C		OV	
	CPM	ECPM	CPM	ECPM	CPM	ECPM	CPM	ECPM
3	94 %	100 %	2	0	3	3	2	3
4	35 %	100 %	22	0	3	2	2	1

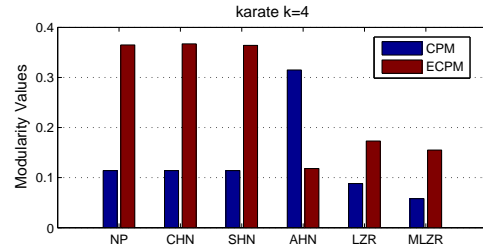
CN: % of nodes covered; UNC: Number of nodes uncovered; |C|: Number of communities

OV: Number of overlapped nodes; ECPM: Proposed algorithm (Extended CPM)

The details of community structures detected by CPM and ECPM for Karate network for k value 3 and 4 are summarized and compared in Table 3.3. It is visible that proposed method ECPM covers all nodes in connected network resulting 100% node coverage i.e 0 uncovered node.



(a) Community structure with $k=3$



(b) Community structure with $k=4$

Figure 3.4: Modularity values for community structures detected using CPM and ECPM in karate club network.

As modularity defines the quality of the detected community structure, modularity is measured for each community structures. As there are no universal modularity measure, more than one overlapping modularity measures are considered to assess the quality of the covers. Here six overlapping modularity measures have been considered. For both $k = 3$ and $k = 4$, overlapping modularity measures have been computed for detected community structures using CPM and ECPM. Modularity values are shown in Figure 3.4. Meaning of NP, CHN, SHN, AHN, LZR, MLZR is defined in Table 2.1. For both the cases with k value as 3 and 4,

modularity values for covers, detected using ECPM are greater than the modularity values for CPM cover. It indicates ECPM results better quality detection.

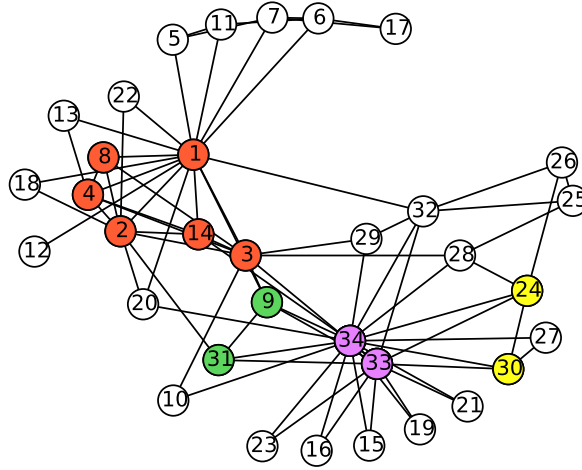


Figure 3.5: Communities detected using CPM in Karate Club Network with $k = 4$. Node colored white are uncovered that is not part of any community. Node 33 and 34 are overlapped.

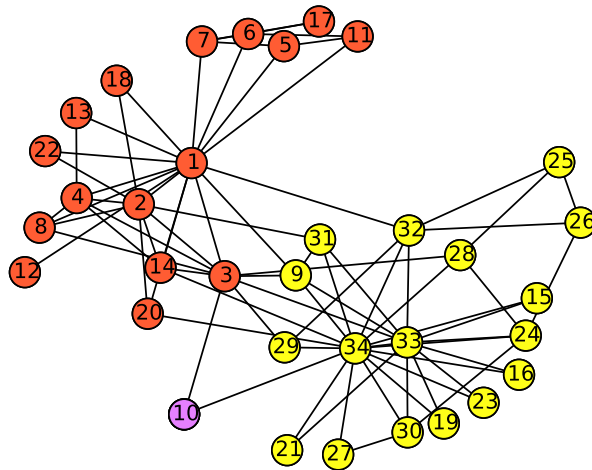


Figure 3.6: Communities detected using proposed method in Karate Club Network with $k = 4$. Node 10 is overlapped between two communities.

3.5.2 Dolphin Social Network

The method proposed in this paper is also evaluated on dolphin network. With $k = 4$, initially CPM detects four non-overlapping communities which are (0, 10, 42, 47), (33, 34, 37, 38, 40, 43, 14, 16, 50), (5, 6, 9, 13, 17, 54, 41, 57), (45, 15, 18, 51, 21, 24, 29). Only 45% nodes are covered in this initial cover. Total of 34 nodes are not covered, which are covered when proposed method is applied resulting 100% coverage. Proposed method detects four overlapping communities with three overlapped nodes. Identified communities are (33, 34, 37, 38, 40, 43, 14, 16, 50, 3, 7, 12, 20, 46, 49, 52, 53, 58, 44, 28, 61, 8, 36), (5, 6, 9, 13, 17, 54, 41, 57, 1, 7, 22, 25, 31, 32, 39, 48, 56, 60, 19, 26, 28, 27, 30), (45, 15, 18, 51, 21, 24, 29, 4, 11, 23, 35, 55, 59) and (0, 10, 42, 47, 2, 28, 30) and overlapped nodes are 28, 30 and 7. Details of community structure for both CPM and ECPM is given in Table 3.4.

For $k = 3$, 74% nodes are covered using CPM with 16 uncovered nodes, six overlapped nodes and four communities. Where, proposed method gives 100% coverage of node with four communities having nine overlapped nodes. Table 3.4 summarizes the outcomes of CPM and ECPM for dolphins network.

Table 3.4: Community Structure Details of Dolphins Social Network

k	CN(%)		UNC		C		OV	
	CPM	ECPM	CPM	ECPM	CPM	ECPM	CPM	ECPM
3	74 %	100 %	16	0	4	4	6	9
4	45 %	100 %	34	0	4	4	0	3

CN: % of nodes covered; UNC: Number of nodes uncovered; |C|: Number of communities

OV: Number of overlapped nodes; ECPM: Proposed algorithm (Extended CPM)

Modularity measures for both $k = 3$ and $k = 4$, have been computed for detected community structures using CPM and ECPM. Modularity values are shown in Figure 3.7. Meaning of NP, CHN, SHN, AHN, LZR, MLZR is defined in Table 2.1. For the covers with $k = 3$, ECPM performs better than CPM in case of modularity NP, CHN, SHN but CPM performs better in other three cases. For the covers with $k = 4$, ECPM gives better modularity value for all the measures

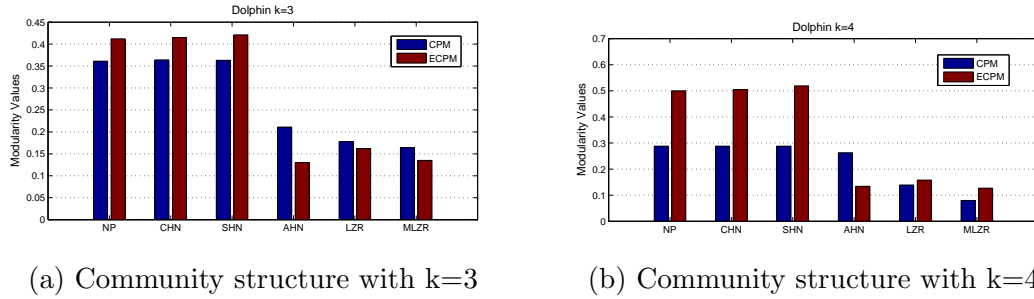


Figure 3.7: Modularity values for community structures detected using CPM and ECPM in dolphin network.

except AHN. So if considered overall, ECPM gives greater modularity values than CPM, which indicates ECPM results better quality detection.

3.5.3 Les-Miserable Network

This is another benchmark network on which proposed method is tested. With $k = 4$, CPM identifies 4 overlapping communities with 4 overlapped nodes. But it covers only 74% nodes with 20 uncovered nodes. ECPM effectively place 20 uncovered nodes into initial communities resulting 100% coverage. ECPM includes some node as overlapped node resulting total 11 overlapped nodes. Again for $k = 3$ also CPM is able to cover only 62% nodes with total 29 uncovered nodes. ECPM includes all uncover nodes to initial communities ensuring 100% coverage. The detail comparison of CPM and ECPM for both $k = 3$ and $k = 4$ is given in Table 3.5.

Table 3.5: Community Structure Details of Les-Miserable Network

k	CN(%)		UNC		C		OV	
	CPM	ECPM	CPM	ECPM	CPM	ECPM	CPM	ECPM
3	74 %	100 %	20	0	4	4	4	11
4	62 %	100 %	29	0	4	4	3	8

CN: % of nodes covered; UNC: Number of nodes uncovered; |C|: Number of communities

OV: Number of overlapped nodes; ECPM: Proposed algorithm (Extended CPM)

Modularity values for community structures detected by CPM and ECPM for both $k = 3$ and $k = 4$ is also computed. Modularity value for different modularity measures for covers of Les-Miserable network, detected by CPM and ECPM is

compared in Figure 3.8. Covers detected by ECPM gives better modularity values which indicates better detection.

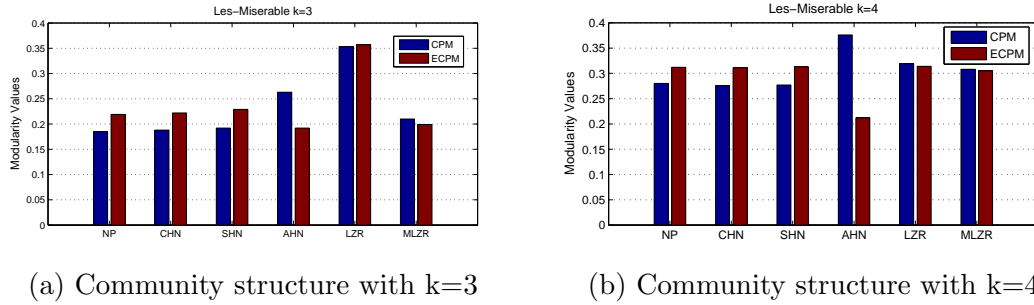


Figure 3.8: Modularity values for community structures detected using CPM and ECPM in Les-Miserable network.

3.6 Summary

In this chapter a new approach to detect overlapping communities has been proposed which extends the existing clique percolation method algorithm. The proposed algorithm includes all connected members of the network to at least one community and still maintain the properties of community. Three classic real world networks are tested by the proposed method. The proposed method efficiently detects overlapping communities and assures 100% node coverage for connected networks. The quality of the community structures is evaluated using modularity measures. For Karate club network proposed method gives the original partition with increased modularity where detection using CPM gives lower modularity with 22 uncovered nodes. Other two tested network; Dolphin network and Les-Miserable network also gives comparable results. It is observed that based on the modularity measure and node coverage, proposed method gives better community structure compared to existing Clique Percolation Method.

Chapter 4

Overlapping Community Detection Using Genetic Algorithm

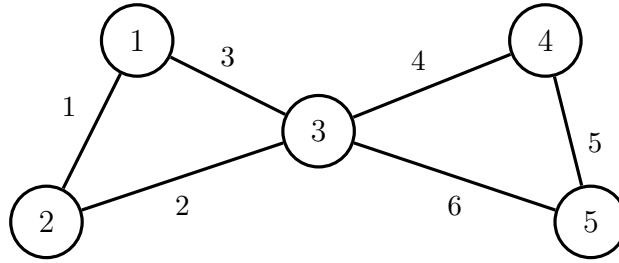
4.1 Introduction

In this chapter a different approach has been used to detect overlapping communities. The method is based on genetic algorithm which is completely different from the graph based approaches used in previous chapter. Conventional genetic algorithm based approaches work by node clustering but those lead to disjoint community structure. The approach used here is based on edge clustering. Each edge represents an unique relationship between two nodes. Edge clustering will discover similar edges. So nodes will automatically belong to multiple communities resulting overlapping communities.

4.2 Research Background

4.2.1 Adjacent Edge

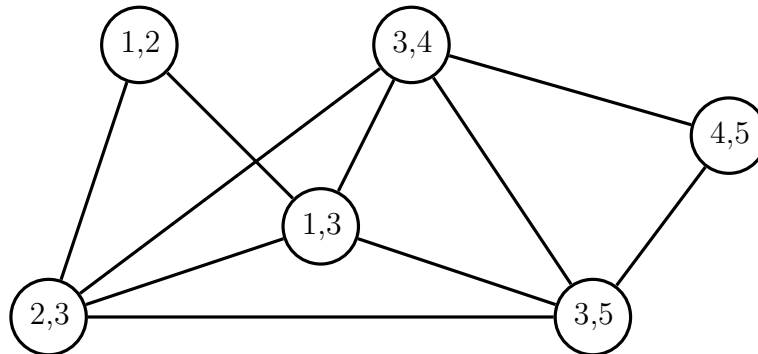
In undirected graph two edges are adjacent if they share one node. For example adjacent edges of edge (1,2) in Figure 4.1 are (1,3) and (2,3).

Figure 4.1: An example graph, G

4.2.2 Line Graph

Line graph, $L(G)$ corresponding to an undirected graph G is another graph where each node in line graph, $L(G)$ represents an edge of G , and two nodes of $L(G)$ are adjacent if their corresponding edges share a node in G . That is in line graph, links represent the adjacent edges in original graph. The original network with all the information content can be recovered easily from the line graph.

For example, an example graph, G is shown in Figure 4.1 where edge labels indicates the edge numbers. The line graph, $L(G)$ of G is shown in Figure 4.2. Line graph, $L(G)$ contains six nodes corresponding to six edges in original graph, G . In the line graph, two nodes are connected by an edge if they share a node in graph, G . For example, there is an edge between the nodes in $L(G)$ labeled $(3, 4)$ and $(4, 5)$ because they share node 4 in G .

Figure 4.2: Line Graph, $L(G)$ corresponding to graph, G

4.3 Genetic Algorithm for Community Detection

Genetic algorithm is an artificial intelligence technique generally used to optimization and search problems. It is derived from evolutionary biology which generates exact or approximate solutions to optimization problems. Genetic algorithm uses the techniques which are inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

To detect overlapping communities effectively by using genetic algorithm, here a special kind genetic representation has been used. The representation include a specific encoding schema to encode the edges of the network and corresponding decoding schema. This encoding and decoding schema reduces the search space effectively and helps to find the solution effectively.

4.3.1 Objective Function

Genetic algorithm solves an optimization problem. In case of community detection an optimum solution gives highest modularity value. So here the problem is to maximize the modularity value. As discussed in section 2.11 there are few overlapping modularity measures. These modularity measures can be used as objective function. Here modularity measure be Chen et al. discussed in section 2.11.3 and summarized in Table 2.1 has been used as objective function.

4.3.2 Genetic Representation

In this section, the genetic representation schema used here has been discussed. The schema includes encoding of edges and decoding them to identify the communities.

4.3.2.1 Encoding Phase

Conventional genetic algorithm representations for community detection is node-based. Here unlike node-based genotype edge-based representation has been used.

In this representation, each individual of the population comprises of m genes where, m is the number of edges present in the network. An individual is represented as $\{g_0, g_1, \dots, g_i, \dots, g_{m-1}\}$. Where i identify the edge number. Value assigned to g_i is restricted to the adjacent edges of i^{th} edge. To obtain edge-based genotype, line graph is used where each node represents an edge.

For example, for the graph G in Figure4.1 and it's corresponding line graph L(G) in Figure 4.2, one possible individual can be

Position	1	2	3	4	5	6
Genotype	2	3	1	6	4	5

where position indicates the edge number. Edges are numbered in the Figure 4.1. Edge (1,2) is edge 1, edge (2,3) is edge 2 and so on. For $i = 1$ that is for edge 1 possible value for g_i can be 2(edge 2 that is (2,3)) and 3(edge 3 that is (1,3)). It can be seen from the line graph that possible value for edge (1,2) is the nodes directly connected to it in line graph. In above individual, for edge 1(edge (1,2)) g_i is taken as 2. Accordingly for other edges value of g_i is assigned to build an individual. So in this encoding schema edges are encoded unlike nodes in node-based schema.

4.3.2.2 Decoding Phase

In the decoding phase an genotype is transfered to partition consist of link communities. In this phase all components corresponding to the genotype are found. The nodes participating to the same component represents one community.

Position	1	2	3	4	5	6
Genotype	2	3	1	6	4	5

In the above individual, $g_1 = 2$ indicates edge 1 is connected with edge 2. $g_2 = 3$ indicates edge 2 is connected with edge 3. $g_3 = 1$ indicate edge 3 is connected with edge 1. Accordingly edge 4 connected with edge 6, edge 5 connected with edge 4 and edge 6 connected with edge 5. This results two components which are (1, 2, 3) and (4, 5, 6). For the first component participating nodes are 1, 2 and 3 and for the second component participating nodes are 3, 4 and 5. This results two communities which are (1, 2, 3) and (3, 4, 5) and node 3 is an overlapped node.

The components and the community structure corresponding to this individual is shown in Figure 4.3 and Figure 4.4 respectively.

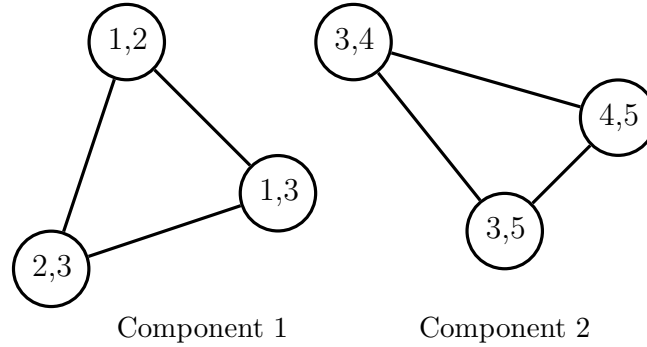


Figure 4.3: Components to corresponding individual

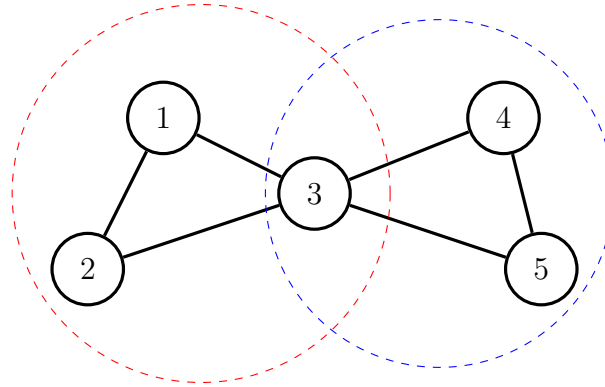


Figure 4.4: Communities in Graph G

4.3.3 Operators

In this work networks are considered as simple undirected connected graph. According to the genetic representation, the operators are as follows

4.3.3.1 Initialization

The initialization process takes in account the effective connections of the nodes in the social network. Generally initial generation of individual is done by random generation. But in this case random generation of individuals may generate components which are not at all connected in the original graph. As discussed, the value $g_i = j$ in the i^{th} position indicates edge i and j are adjacent to each

other. A randomly generated individual may produce the value of $g_i = j$ in the i^{th} position, irrespective of i and j are not adjacent.

To avoid this problem, once an individual is generated randomly, it is repaired. A check is done to verify whether there is a connection between i and g_i in the line graph, that is i and g_i are adjacent links in original graph. If not then value of g_i is substituted with one of the neighbors of i . Individual generated by this procedure is a safe individual as it will not consist of connections which does not exist in the original network. Using of safe individual reduces the search space.

4.3.3.2 Uniform Crossover

Uniform crossover has been used. Because of the repairing step each individual in the population is safe that is if an i^{th} gene contain value j then i^{th} and j^{th} edges are adjacent or inversely (i, j) link exist in link-graph. Thus crossover is performed with two selected safe parents. A random binary vector with same length as the individual is generated. Uniform crossover then exchanges the genes of two parents where the vector value is 0 to create two safe children. At each position, i of the individual, both the child contains the value of gene, $g_i = j$ from either the first parent or the second. Thus adjacency between edge (i, j) exists which implies two safe children is generated from two safe parents.

Position	1	2	3	4	5	6
Parent 1	3	1	4	5	6	3
Parent 2	2	3	1	6	4	5
Vector	1	0	1	0	0	1
Child 1	2	1	1	5	6	5
Child 2	3	3	4	6	4	3

4.3.3.3 Mutation

The mutation operator randomly change the value of g_i . But again the value of g_i is restricted to the neighbors of i in link-graph that is only the adjacency edges of edge i is allowed. For mutation, randomly a gene is selected and it's value is replaced by one of permitted values. This repaired mutation guarantees the generation of a safe mutated child.

4.3.4 Algorithmic Description

The algorithm works by generating a population which is initialized at random and then repairing each individual to produce safe-individual. After this the fitness for each individual is evaluated. The fitness is evaluated on the original graph, G , not on the line graph, $L(G)$. Thus each individual translated into an overlapping division of G and then fitness value is computed using the objective function. The modularity value used as objective function is need to be maximized. Variation operators have been used to optimize the modularity value, used as objective function. Overall framework of the process is defined in Algorithm 3.

Algorithm 3 Framework of GA Based Overlapping Community Detection

- 1: Compute the line graph $L(G)$ associated with G .
 - 2: Create an initial population of random individuals. Where length of each individual equals the number of edges in G .
 - 3: **while** generation number $<$ iterations required **do**
 - 4: for each individual $I = \{g_1, g_2, \dots, g_i, \dots, g_m\}$ of the population, components are identified and corresponding node communities are found.
 - 5: evaluate the fitness of individuals by applying fitness function on the communities for the individual.
 - 6: create a new population of individuals by applying the GA operators
 - 7: **end while**
-

4.4 Result and Analysis

Implementation of the above algorithm has been done in Python language. NetworkX package has been used for graph manipulation. Networks are considered as simple, undirected and unweighted. Parameters considered for the genetic algorithm are, crossover rate 0.8, mutation rate 0.2, elite reproduction 10% of the population size. The population size is taken as 50 and the number of generations is taken as 60.

4.4.1 Zachary’s Karate Club Network

The GA based algorithm is effective enough to detect overlapping communities. For karate club network this algorithm identifies two overlapping communities with six overlapped nodes. Two detected communities are (2, 31, 1, 3, 4, 8, 22, 6, 17, 11, 20, 7, 18, 14, 13, 5, 32, 9, 12) and (24, 30, 9, 34, 20, 26, 28, 25, 32, 3, 33, 19, 10, 29, 21, 14, 15, 23, 16, 31, 27). Six overlapped nodes are 32, 3, 9, 14, 20 and 31. The algorithm detects two communities nearly correct. Though it identifies several nodes as overlapped nodes than the actual overlapped nodes, but it correctly discover two partitions. The algorithm results 100% node coverage i.e., all nodes are part of some community. The detected community structure is shown in Figure 4.5.

As modularity quantifies the quality of the community structure, various modularity are measured for the detected cover. Overlapping modularity measures as summarized in Table 2.1 are evaluated with the detected cover.

Details of the detected community structure and value of different modularity measures are described in Table 4.1.

Table 4.1: Community Structure details of Karate Club Network, GA

CN	$ C $	OV	NP	CHN	SHN	AHN	LZR	MLZR
100 %	2	6	0.306	0.331	0.365	0.130	0.154	0.160

CN: % of nodes covered; $|C|$: Number of communities; OV: Number of overlapped nodes

NP, CHN, SHN, AHN, LZR, MLZR are modularity measures described in Table 2.1

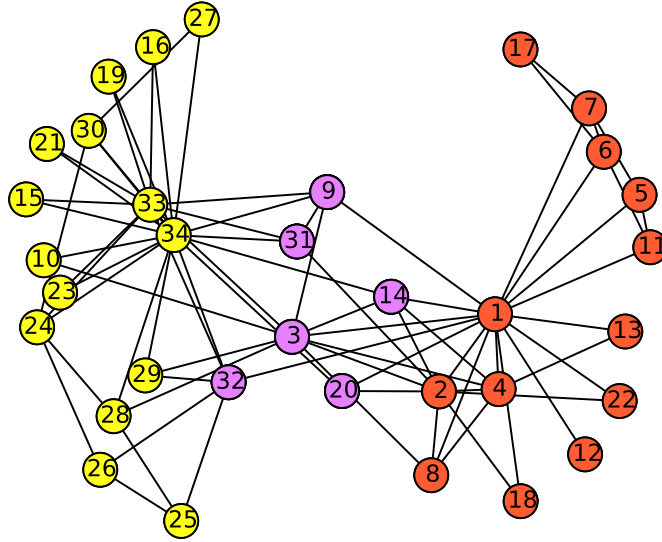


Figure 4.5: Communities detected using genetic algorithm in Karate Club Network. Red and Yellow colored nodes represent two different community. Nodes colored in magenta are overlapped nodes

4.4.2 Dolphin Social Network

Dolphin social network, another benchmark network for community detection, is also tested with the genetic algorithm based overlapping community detection algorithm. Genetic algorithm based algorithm detect three communities in dolphin network with twelve overlapped nodes. Various modularity are measured with detected community structure. Details of detected cover and modularity values for detected cover are given in Table 4.2. This algorithm covers all nodes for this networks also.

Table 4.2: Community Structure details of Dolphin Social Network, GA

CN	$ C $	OV	NP	CHN	SHN	AHN	LZR	MLZR
100 %	3	12	0.327	0.354	0.372	0.094	0.043	0.109

CN: % of nodes covered; $|C|$: Number of communities; OV: Number of overlapped nodes

NP, CHN, SHN, AHN, LZR, MLZR are modularity measures described in Table 2.1

4.4.3 Les-Miserable Network

One more benchmark network, Les-Miserable network is also used to evaluate the algorithm. The algorithm detects eight overlapping communities with thirty five overlapped nodes. Modularity measures are evaluated with detected cover. Details of the cover and value of various modularity measures are enlisted in Table 4.3. Again for this network also node coverage is 100%. This indicate that this GA based algorithm covers all nodes in the network while finding communities.

Table 4.3: Community Structure details of Les-Miserable Network, GA

CN	$ C $	OV	NP	CHN	SHN	AHN	LZR	MLZR
100 %	8	35	0.229	0.310	0.411	0.282	0.084	0.136

CN: % of nodes covered; $|C|$: Number of communities; OV: Number of overlapped nodes
NP, CHN, SHN, AHN, LZR, MLZR are modularity measures described in Table 2.1

4.4.4 Comparison with CPM and ECPM

In this section the genetic algorithm based overlapping community detection algorithm is compared with two previously discussed algorithm, Clique Percolation Method (CPM) and Extended Clique Percolation Method (ECPM).

Algorithms are compared on the basis of modularity measures. As modularity measure quantify the quality of detected community structure or cover. Maximum value of modularity for the cover indicates the better community structure. No modularity measure is proved globally ideal. So quality of the cover should not be quantified based on a single modularity. Here six overlapping modularity measures have been considered to evaluate the quality of the covers.

Figure 4.6 shows various modularity measure values for three different covers in Karate club network detected by CPM, ECPM and GA based algorithm respectively. From the figure it is visible that modularity NP, CHN, SHN and LZR maximizes for ECPM. CPM produces maximum value only for modularity AHN. The GA based algorithm gives better modularity value than CPM algorithm for all cases except AHN. But GA based algorithm does not gives better value than ECPM. only in case of AHN and MLZR, GA based algorithm outperforms ECPM.

As in most of the cases ECPM results better modularity value, it can be concluded that ECPM algorithm gives better results for karate club network.

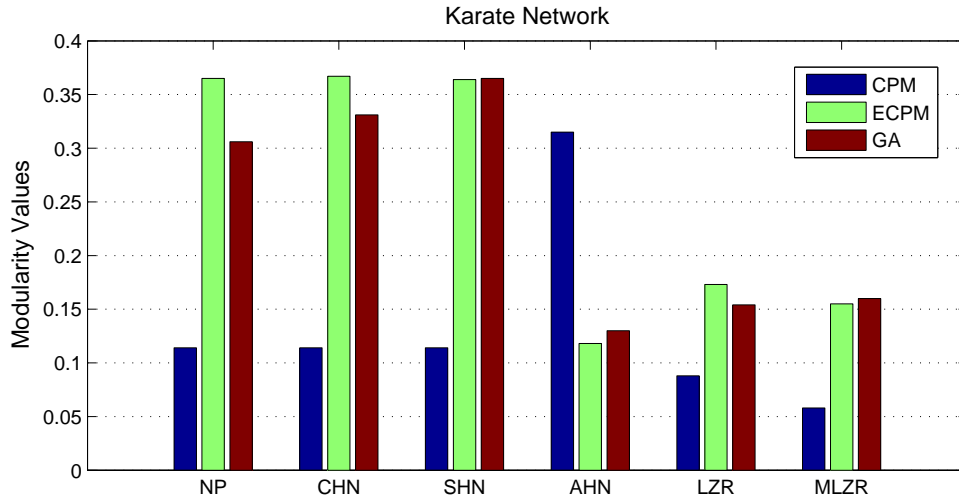


Figure 4.6: Comparison of modularity values for community structures detected using CPM with $k = 4$, ECPM and GA based Algorithm in karate club network.

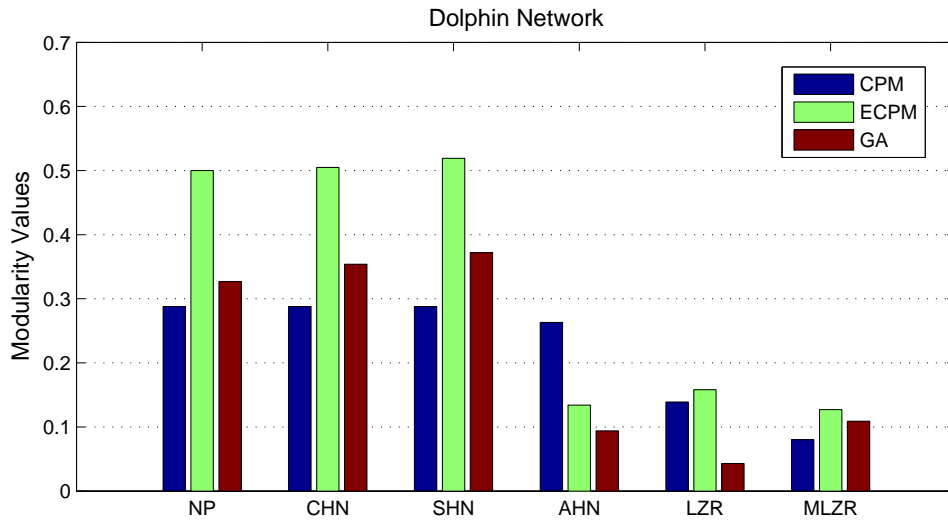


Figure 4.7: Comparison of modularity values for community structures detected using CPM with $k = 4$, ECPM and GA based Algorithm in dolphin network.

Figure 4.7 shows various modularity measure values for three different cover for the Dolphin network. Modularity NP, CHN, SHN, LZR, and MLZR maximizes for ECPM algorithm. Only modularity AHN does not gives maximum value for

ECPM. Modularity AHN gives maximum value for CPM and CPM produces maximum modularity for one and only AHN. GA based approach performs better than CPM as it gives better result than CPM in most of the cases. Again for this network also ECPM performs better than both CPM and GA. GA performs better than CPM but not better than ECPM.

Figure 4.8 shows various modularity measure values for three different cover for the Les-Miserable network. For modularity measure SHN, GA based algorithm gives better value and for modularity AHN, it gives better value than ECPM algorithm though not better than CPM and for CHN it gives better value than CPM. ECPM algorithm gives better modularity than GA based algorithm for modularities NP,CHN,LZR,MLZR. So for this network ECPM perform better than GA based algorithm. Again CPM produces better result with modularity AHN, LZR, MLZR. ECPM also performs better than CPM in all cases except AHN and LZR. So if considered overall, again ECPM performs better for this network as well.

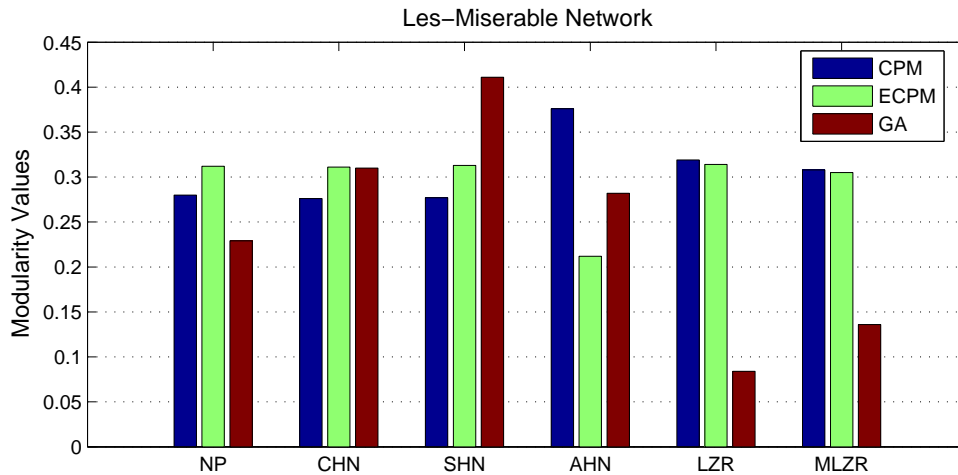


Figure 4.8: Comparison of modularity values for community structures detected using CPM with $k = 4$, ECPM and GA based Algorithm in les-miserable network.

4.5 Summary

In this chapter another approach has been considered to detect overlapping communities. the algorithm is based on genetic algorithm. Encoding and decoding of network to perform GA operations has been discussed. Three classic real world networks are tested with the implemented approach. This GA based approach can detects overlapping communities and assures 100% node coverage for connected networks. Quality of the community structures are evaluated using modularity measures. The algorithm performs better than Clique Percolation Method algorithm with respect to both node coverage and modularity. Though the algorithm does not perform better than ECPM, but still it is effective enough to detect overlapping communities.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this work overlapping communities are identified in social networks. Two overlapping community detection algorithm has been studied. To quantify the discovered community structure various overlapping modularity measure has been considered. Three classic real world networks are used to test the algorithms.

In the first part of work, a method is proposed which overcomes the poor-coverage problem of one traditional method, clique percolation method. The proposed method efficiently detects overlapping communities and assures 100% node coverage for connected network. It is observed that based on the modularity measure and node coverage, proposed method gives better community structure compared to the clique percolation method.

In the second part, another proposed method, genetic algorithm has been used to detect overlapping communities. From the results it has been observed that this approach also detects overlapping communities effectively. Though this approach does not always give better results than previous one (proposed in the first part) but it also covers all nodes for the connected network.

It can be concluded from the result that the community structure depends only on the considered network and vary from network to network.

5.2 Future Work

Overlapping community detection is still a challenge. Though there are several proposed methods, but most of them take a huge amount of processing time. So emphasis should be given to effective algorithms which will be able to detect communities in a huge social network in allowable time. In this work only unweighted and undirected networks has been taken into consideration. In future weighted and directed networks are needed to be considered for community detection. Now days almost all social networks are dynamic that is some members are joining and some are leaving every moment. So it will be great if communities can be detected in dynamic networks.

Bibliography

- [1] M. E. Newman and M. Girvan, “Finding and evaluating community structure in networks,” *Physical review E*, vol. 69, no. 2, p. 026113, 2004.
- [2] H.-W. Shen, X.-Q. Cheng, and J.-F. Guo, “Quantifying and identifying the overlapping community structure in networks,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, no. 07, p. P07042, 2009.
- [3] H. Shen, X. Cheng, K. Cai, and M.-B. Hu, “Detect overlapping and hierarchical community structure in networks,” *Physica A: Statistical Mechanics and its Applications*, vol. 388, no. 8, pp. 1706–1712, 2009.
- [4] M. E. Newman, “Modularity and community structure in networks,” *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [5] J. Xie, S. Kelley, and B. K. Szymanski, “Overlapping community detection in networks: the state of the art and comparative study,” *arXiv preprint arXiv:1110.5813*, 2011.
- [6] S. Fortunato, “Community detection in graphs,” *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.
- [7] A. Lancichinetti, S. Fortunato, and J. Kertész, “Detecting the overlapping and hierarchical community structure in complex networks,” *New Journal of Physics*, vol. 11, no. 3, p. 033015, 2009.
- [8] Z. Liu, J.-L. He, and J. Srivastava, “Cliques in complex networks reveal link formation and community evolution,” *arXiv preprint arXiv:1301.0803*, 2013.

- [9] Z. Liu, J.-L. He, K. Kapoor, and J. Srivastava, “Correlations between community structure and link formation in complex networks,” *PLoS ONE*, vol. 8, no. 9, p. e72908, 2013.
- [10] A. Clauset, “Finding local community structure in networks,” *Physical review E*, vol. 72, no. 2, p. 026132, 2005.
- [11] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, “Uncovering the overlapping community structure of complex networks in nature and society,” *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.
- [12] M. E. Newman, “Finding community structure in networks using the eigenvectors of matrices,” *Physical review E*, vol. 74, no. 3, p. 036104, 2006.
- [13] T. Nepusz, A. Petróczy, L. Négyessy, and F. Bazsó, “Fuzzy communities and the concept of bridgeness in complex networks,” *Physical Review E*, vol. 77, no. 1, p. 016107, 2008.
- [14] S. Gregory, “Finding overlapping communities using disjoint community detection algorithms,” in *Complex Networks*, pp. 47–61, Springer, 2009.
- [15] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, “Link communities reveal multi-scale complexity in networks,” *Nature*, vol. 466, no. 7307, pp. 761–764, 2010.
- [16] D. Chen, M. Shang, Z. Lv, and Y. Fu, “Detecting overlapping communities of weighted networks via a local algorithm,” *Physica A: Statistical Mechanics and its Applications*, vol. 389, no. 19, pp. 4177–4187, 2010.
- [17] A. Lázár, D. Ábel, and T. Vicsek, “Modularity measure of networks with overlapping communities,” *EPL (Europhysics Letters)*, vol. 90, no. 1, p. 18001, 2010.
- [18] N. P. Nguyen, T. N. Dinh, S. Tokala, and M. T. Thai, “Overlapping communities in dynamic networks: their detection and mobile applications,” in

- Proceedings of the 17th annual international conference on Mobile computing and networking*, pp. 85–96, ACM, 2011.
- [19] M. Coscia, G. Rossetti, F. Giannotti, and D. Pedreschi, “Demon: a local-first discovery method for overlapping communities,” in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 615–623, ACM, 2012.
- [20] B. Tóth, T. Vicsek, and G. Palla, “Overlapping modularity at the critical point of k-clique percolation,” *Journal of statistical physics*, vol. 151, no. 3-4, pp. 689–706, 2013.
- [21] C. Pizzuti, “Ga-net: A genetic algorithm for community detection in social networks,” in *Parallel Problem Solving from Nature–PPSN X*, pp. 1081–1090, Springer, 2008.
- [22] C. Pizzuti, “Overlapped community detection in complex networks,” in *GECCO*, vol. 9, pp. 859–866, 2009.
- [23] Y. Cai, C. Shi, Y. Dong, Q. Ke, and B. Wu, “A novel genetic algorithm for overlapping community detection,” in *Advanced Data Mining and Applications*, pp. 97–108, Springer, 2011.
- [24] B. Dickinson, B. Valyou, and W. Hu, “A genetic algorithm for identifying overlapping communities in social networks using an optimized search space,” *Social Networking*, vol. 2, p. 193, 2013.
- [25] W. Zachary, “An information flow model for conflict and fission in small groups,” *Journal of anthropological research*, vol. 33, no. 4, pp. 452–473, 1977.
- [26] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, “The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations,” *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, 2003.

- [27] D. E. Knuth, *The Stanford GraphBase: a platform for combinatorial computing*, vol. 37. Addison-Wesley Reading, 1993.
- [28] J. Travers and S. Milgram, “An experimental study of the small world problem,” *Sociometry*, vol. 32, no. 4, pp. 425–443, 1969.
- [29] M. Scholz, “Node similarity as a basic principle behind connectivity in complex networks,” *arXiv preprint arXiv:1010.0803*, 2010.
- [30] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, “Complex networks: Structure and dynamics,” *Physics reports*, vol. 424, no. 4, pp. 175–308, 2006.
- [31] M. E. Newman, “The structure and function of complex networks,” *SIAM review*, vol. 45, no. 2, pp. 167–256, 2003.
- [32] S. Engen, V. Grøtan, and B.-E. Sæther, “Estimating similarity of communities: a parametric approach to spatio-temporal analysis of species diversity,” *Ecography*, vol. 34, no. 2, pp. 220–231, 2011.
- [33] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, “Extending the definition of modularity to directed graphs with overlapping communities,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, no. 03, p. P03024, 2009.
- [34] M. Van Steen, *Graph Theory and Complex Networks: An Introduction*. Maarten van Steen, 2010.