Chopade and Zhan *Journal of Big Data* (2015) 2:11
DOI 10.1186/s40537-015-0019-y

Journal of Big Data
a SpringerOpen Journal

RESEARCH Open Access

Structural and functional analytics for community detection in large-scale complex networks

Pravin Chopade[1*†] and Justin Zhan[2†]

*Correspondence:
pvchopad@ncat.edu
†Equal contributors
[1] Department of Computer Science, College of Engineering, North Carolina A and T State University, Greensboro, NC, USA, 305 Cherry Hall, 1601 East Market Street, NC-27411, Greensboro, USA
Full list of author information is available at the end of the article

Abstract

Community structure is thought to be one of the main organizing principles in most complex networks. Big data and complex networks represent an area which researchers are analyzing worldwide. Of special interest are groups of vertices within which connections are dense. In this paper we begin with discussing community dynamics and exploring complex network structural parameters. We put forward structural and functional models for analyzing complex networks under situations of perturbations. We introduce modified adjacency and modified Laplacian matrices. We further introduce network or degree centrality (weighted Laplacian centrality) based on modified Laplacian, weighted micro-community centrality. We discuss its robustness and importance for micro-community detection for social and technological complex networks with overlapping communities. We also introduce 'k-clique sub-community' overlapping community detection based on degree and weighted micro-community centrality. The proposed algorithms use optimal partition of k-clique sub-community for modularity optimization. We establish relationship between degree centrality and modularity. This proposed method with modified adjacency matrix helps us solve NP-hard problem.

Keywords: Community; Big data; Complex network; Laplacian; Centrality; Robustness; Modularity

Introduction

The last decade has witnessed the birth of a new field of interest and research in the study of complex networks, i.e. networks whose structure is irregular, complex and dynamically evolving in time, with the main focus moving from the analysis of small networks to that of systems with thousands or millions of nodes, and with a renewed attention to the properties of networks of dynamical units. Networks are all around us, and we are ourselves, as individuals, the units of a network of social relationships of different kinds and, as biological systems, the delicate result of a network of biochemical reactions. Networks can be tangible objects in the Euclidean space, such as electric power grids, the Internet, highways or subway systems, and neural networks. Or they can be entities defined in an abstract space, such as networks of acquaintances or collaborations between individuals [1].

The network construction from general, real-world data presents several unexpected challenges owing to the data domains themselves, e.g., information extraction and pre-processing, and to the data structures used for knowledge representation and storage. The increased availability of large-scale, real-world sociographic data has ushered in a new era of research and development in social network analysis. The quantity of content-based data created every day by traditional and social media, sensors, and mobile devices provides great opportunities and unique challenges for the automatic analysis, prediction, and summarization in the era of what has been dubbed "Big Data" [2].

Centrality is one of the most studied concepts in social network analysis to characterize social power and structural influence [3]. When studying faults and fault propagation in physical networks, complex networks such as smart grid, communication, highway, traffic networks, centrality plays a somewhat different role than in social networks [4].

In this paper we discuss structural and functional analysis of complex technological and social networks. First we discuss various existing structural analysis parameters. Major contribution of this work is modified relationship between adjacency and Laplacian matrix. We use this modified relationship to define new degree centrality and new modularity. Using these new degree centrality and new modularity we are able to detect micro level overlapping community structures. We introduce network or degree centrality (weighted Laplacian centrality) based on modified Laplacian, weighted micro-community centrality and discuss its robustness and importance for micro-community detection for social and technological complex networks with overlapping communities. We also introduce '*k*-clique sub-community' overlapping community detection based on degree and weighted micro-community centrality. These new matrices and algorithms are helpful for identifying hidden level vulnerabilities. First we review various complex network structural parameters. We further put forward new community detection based on network or degree centrality. In the related work section, we review and discuss existing community detection methods and algorithms. The our approach section discusses about community dynamics, research approach and complex network structural parameters. The Methodology section discusses analysis of unweighted, weighted networks (functional analysis), where we introduce modified relationship between adjacency, degree and Laplacian matrices. Using this we define weighted Laplacian centrality, weighted micro-community centrality and related algorithms. We also discuss and introduce algorithm for *k*-clique sub-community and optimal partition of *k* -clique sub-community for weighted modularity optimization and overlapping community detection. In the "Results and discussion" section, we analyse real world complex networks and carry out comparison of different community detection algorithms. Lastly we discuss computational complexity of our proposed algorithms and conclude the paper with major findings and future works.

Background and literature review

Community detection is a fundamental component of network analysis for sensor systems and is an enabling technology for higher level analytical applications such as behavior analysis, prediction, and identity and pattern-of-life analysis [2]. In both commercial industry and academia, significant progress has been made on problems related to the analysis of community structure; however, traditional work in social networks has focused on static situations (i.e., classical social network analysis) or dynamics in a large-scale sense (e.g., disease propagation) [2].

Communities are of interest for a number of reasons. They have intrinsic interest because they may correspond to functional units within a networked system [5]. The aim of community detection in graphs is to identify the modules and, possibly, their hierarchical organization, by only using the information encoded in the graph topology. Community detection is important for other reasons, too. Identifying modules and their boundaries allows for a classification of vertices, according to their structural position in the modules. So, vertices with a central position in their clusters, i.e. sharing a large number of edges with the other group partners, may have an important function of control and stability within the group; vertices lying at the boundaries between modules play an important role of mediation and lead the relationships and exchanges between different communities [6]. Fortunato [6] discussed various crucial issues of community detection like the significance of clustering and its application to real networks. This paper triggered a big activity in the field, and many new methods have been proposed in the last years.

With the aim at explaining and comprehending common principles and properties in real networks, three general network models have been intensely researched: random network [7], small-world network [8] and scale-free network [9], though these models cannot interpret all phenomena observed in real networks. Random network has binomial or Poisson degree distribution [10], so random network is rather robust since it is a homogeneous network where majority of vertices almost have the same number of edges to be connected. However, real networks do not show random distribution and properties. Small-world is a network between a lattice and random networks. Small-world network has smaller average path length like a random network but larger clustering coefficient like a lattice network. Rather unexpectedly, the degree distribution of small-world network is mathematically explained by binomial distribution that is same as random network. Besides, most of real networks have the degree distribution that is power law [11] rather than Poisson distribution and these networks are called as scale-free network which is sensitive to intentional removal of vertices but robust against randomly removing vertices because the power law distribution shows it is a heterogeneous network where a larger number of vertices have larger edges to be connected and these vertices are called as hubs that play important role in connectivity of networks [12].

Centrality measures the relative importance of a node or a link in terms of the network efficiency and utilization of the network resources. Koschutzki et al. [13] discusses centrality indices based on degree considering distances and neighborhoods as well as shortest paths. Koschutzki et al. presented some of the more influential, 'classic' centrality indices but he did not strive for completeness and provide a catalog of basic centrality indices with some of their main applications.

Borgatti [14] claimed that centrality measures can be regarded as generating expected values for certain kinds of node outcomes (such as speed and frequency of reception) given implicit models of how traffic flows. Borgatti regarded the formulas for centrality concepts like betweenness and closeness as generating the expected values under specific unstated flow models of certain kinds of node participation in network flows.

As such, they do not actually measure node participation at all but rather indicate the expected participation if things flow in the assumed way. One contribution of Borgatti's paper is to make explicit what the assumptions behind each measure are, and then to test each measures deconstruction via simulation. Node-centric measures are more convenient for computation and interpretation, hence more common than edge-centric measures.

The problem of community detection requires the partition of a network into communities of densely connected nodes, with the nodes belonging to different communities being only sparsely connected. Precise formulations of this optimization problem are known to be computationally intractable. Several algorithms have therefore been proposed to find reasonably good partitions in a reasonably fast way [15]. One of the proposed algorithms is by Greedy sketch method for modularity Q optimization [16]. It is an agglomerative hierarchical clustering method, where groups of vertices are successively joined to form larger communities such that modularity increases after the merging. Greedy optimization method attempts to optimize the "modularity" of a partition of the network. The optimization is performed in two steps. First, the method looks for "small" communities by optimizing modularity locally. Second, it aggregates nodes belonging to the same community and builds a new network whose nodes are the communities. These steps are repeated iteratively until a maximum of modularity is attained and a hierarchy of communities is produced.

By assumption, high values of modularity Q indicate good partitions. So, the partition corresponding to its maximum value on a given graph should be the best or at least a very good one. This is the main motivation for modularity maximization, by far the most popular class of methods to detect communities in graphs. An exhaustive optimization of Q is impossible, due to the huge number of ways in which it is possible to partition a graph, even when the latter is small. Besides, the true maximum is out of reach, as it has been recently proved that modularity optimization is an NP-complete problem [17], so it is probably impossible to find the solution in a time growing polynomially with the size of the graph. However, there are currently several algorithms able to find fairly good approximations of the modularity maximum in a reasonable time [6].

Integer linear programming algorithms solve the modularity maximization problem for small graphs [16, 18]. Brandes et al. [18] have given an integer linear programming formulation for modularity clustering and established that the formal problem is – in the worst case – NP-hard.

Gregori et al. [19] presented a novel, parallel k-clique community detection method, based on an innovative technique which enables connected components of a network to be obtained from those of its subnetworks. The novel method has an unbounded, userconfigurable, and input-independent maximum degree of parallelism, and hence is able to make full use of computational resources. Chen et al. [20] introduce two novel fine-tuned community detection algorithms that iteratively attempt to improve the community quality measurements by splitting and merging the given network community structure but they did not consider optimal number of clusters or subnetwork or concept of modularity for community detection.

Considering the importance of the community detection problem this work aim to identify hidden layer micro-community, overlapping communities and related functional dynamics by using concept of modified adjacency and modified Laplacian matrices.

Research design and methodology

Research design

Many social networks exhibit community structure. Communities are groups of nodes that have high connectivity within a group and low connectivity across groups. Communities roughly correspond to organizations and groups in real social networks. Figure 1 shows our community detection research process. We will apply our developed algorithm for large-scale big data networks. This algorithm will explore or extract different community structures which will represent properties of real networks such as random, small world and scale-free network.

Figure 2 shows research methodology which holds true for any type of network. Here, the aim of network analysis is to study how the performance of networks is affected by the removal of vertices and edges, to compare the structure of different networks, and to analyse how the change of structure affects the vulnerability of networks.

Complex network structural parameters

Structural parameters are the tools of Complex Network Analysis which are of useful to understand salient properties of complex systems. Some of the important local and global structural parameters are discussed below

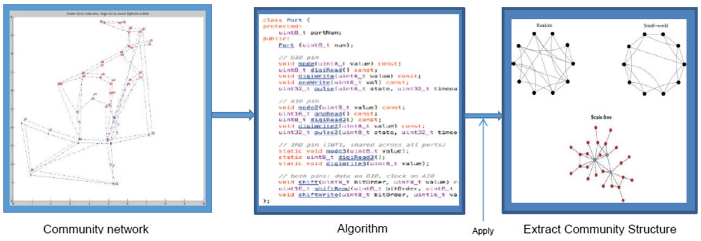
Node degree distributions, correlations and assortativity The degree (or connectivity) k_i of a node i is the number of edges incident with the node. It is defined in terms of the adjacency matrix A as given by $k_i^{in} = \sum A_{ij}$. For directed network total degree is sum of in-degree of node and out-degree of node given with Eqs. 1 to 3.

$$k_i^{in} = \sum_{j \in N} A_{ij} \tag{1}$$

$$k_i^{out} = \sum_{j \in N} A_{ji} \tag{2}$$

$$k_i = k_i^{in} + k_i^{out} \tag{3}$$

The degree distribution, usually denoted by $P(k)$, is the probability that a vertex chosen uniformly at random has degree k, or equivalently, the fraction of vertices in the network with degree k. In many real networks it has been found that the degree distribution follows



```
class Port {
protected:
    uint8_t portNum;
public:
    Port (uint8_t num);

    // DIO pin
    void mode(uint8_t value) const;
    uint8_t digiRead() const;
    void digiWrite(uint8_t value) const;
    void anaWrite(uint8_t val) const;
    uint32_t pulse(uint8_t state, uint32_t timeou

    // AIO pin
    void mode2(uint8_t value) const;
    uint16_t anaRead() const;
    uint8_t digiRead2() const;
    void digiWrite2(uint8_t value) const;
    uint32_t pulse2(uint8_t state, uint32_t timeo

    // IRQ pin (INT1, shared across all ports)
    static void mode3(uint8_t value);
    static uint8_t digiRead3();
    static void digiWrite3(uint8_t value);

    // both pins: data on DIO, clock on AIO
    void shift(uint8_t bitOrder, uint8_t value) c
    uint16_t shiftRead(uint8_t bitOrder, uint8_t 
    void shiftWrite(uint8_t bitOrder, uint16_t va
};
```

Fig. 1 Research process

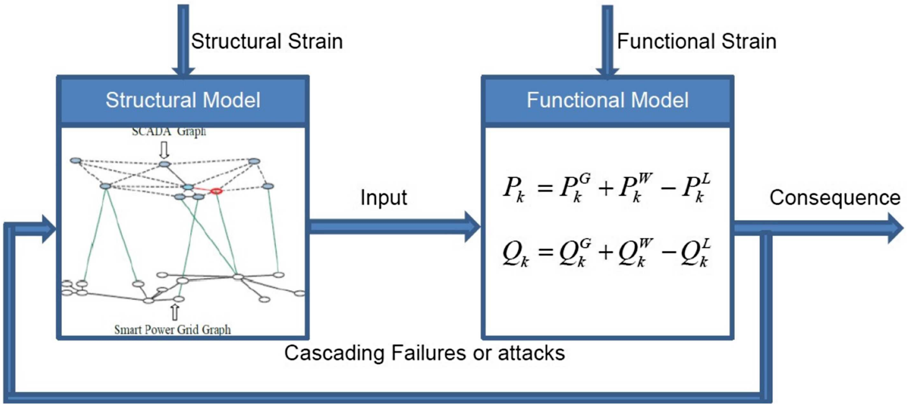


Fig. 2 Research methodology

a power-law, i.e. $P(k)^{\sim}k^{-\alpha}$, where α is the scaling coefficient, it is typically between 1 and 3 [21]. A large number of real networks are correlated in the sense that the probability that a node of degree k is connected to another node of degree, say $k^{'}$ depends on k. The degree correlations are formally characterized by $P(k^{'}|k)$. Some networks (including the Internet and the World Wide Web) have degree distributions in the form of a power law: that is, the probability that a node has degree k is given as $P(k)^{\sim}k^{-\alpha}$ [22]. Assortativity is the correlation between the degrees of connected nodes. Positive assortativity indicates that high-degree nodes tend to connect to each other.

Shortest path lengths or characteristics path length Average path length is the distance between two vertices is defined as the number of edges along the shortest path connecting them. Many complex networks, despite their often-large size, have a relatively short average path length between any two vertices.

Let the community network be represented as a graph $G_n = \{V, E\}$ with N nodes, $V = \{v_i\}$ is the set of vertices and E the set of edges. Denote by $d(v_i, v_j) = d_{ij}$ the shortest path lengths (shortest distance) connecting two nodes i and j in the community network. The average path length l is given by,

$$l = \frac{1}{N(N-1)} \sum_{i,j} d_{ij} \tag{4}$$

The community network is divided into two subcommunities, G_{c_1} representing the subcommunity 1, and G_{c_2} representing the subcommunity 2. Then the interdependent structural efficiency $X(G_{c_1} \cap G_{c_2})$ of the community network can be defined as follows [23]:

$$X\left(G_{c_1} \cap G_{c_2}\right) = \frac{1}{N_{c_1}.N_{c_2}} \sum_{\substack{i \in G_{c_1} \\ j \in G_{c_2}}} \frac{1}{d_{ij}} \tag{5}$$

where N_{c_1} is the number of resource nodes in the subcommunity 1, and N_{c_2} is the number of nodes in the subcommunity 2.

When two nodes are not connected at all, or become disconnected due to attacks, their shortest path length d_{ij} becomes infinite, and then $\frac{1}{d_{ij}}$ is zero. If $X\left(G_{c_1} \cap G_{c_2}\right)$ is large, it is indicated that the network is well connected and has high efficiency [24].

Local and global clustering coefficient If the nearest neighbours of a node are also directly connected to each other they form a cluster. The clustering coefficient quantifies the number of connections that exist between the nearest neighbours of a node as a proportion of the maximum number of possible connections [8]. Interactions between neighbouring nodes can also be quantified by counting the occurrence of small motifs of interconnected nodes [25]. The distribution of different motif classes in a network provides information about the types of local interactions that the network can support [26].

The local clustering coefficient (Cliques): For the modular network cliques (or similar measures) identify interesting sub-components of the network. This metric can help to identify functionally related genes/proteins in the network. The local clustering coefficient, CC_i, of a vertex i is the ratio between the actual number of edges that exist between the vertex and its neighbors and the maximum number of possible edges between these neighbors. The CC_i of the network is defined as:

$$CC_i(local) = \frac{m_i}{k_i(k_i - 1)/2} \tag{6}$$

Here CC_i is the *local clustering coefficient*, m_i is the number of edges that exist between the neighbors of vertex i and k_i is the number of neighbors for vertex i. The denominator $k_i(k_i - 1)/2$ is the maximum possible number of edges that can exist between the neighbors of vertex i.

The global clustering coefficient CC is the ratio of the number of triangles in a network versus the number of paths of length 2. This ratio is typically high in social networks, whose generative processes tend to close triangles. In contrast, the clustering coefficient is close to 0 for random graphs.

The global clustering coefficient, CC of the network is defined as:

$$CC(global) = \frac{1}{N}\sum_{i\in V} CC_i = \frac{1}{N}\sum_{i\in V} \frac{m_i}{k_i(k_i - 1)/2} \tag{7}$$

N represents the number of vertices or the number of nodes in the network. A general problem of network measures, such as the clustering coefficient, is whether sampling or perturbations change the values of these measures. Network measures are frequently used for the classification of different networks [27] or of topological changes (addition or deletion of nodes or edges) within the same network.

Network centrality and robustness The structure of many networks is governed by latent communities or clusters. For example, in a social network, people which are part of the same latent community are more likely to be friends and therefore be connected in the network. Very often it is useful to learn these latent communities in order to better understand the structural composition of a network. The challenge is then to figure out how to use the available network data to find these latent communities. Social networks have the added complexity that very often the users belong to multiple communities, so there is considerable overlap in the communities. For example, in Fig. 3, we show a social network where there are three overlapping communities: people from work, family, and college. This type of overlapping community structure is very common in social networks, so finding the community structure is more complex than simply partitioning the network into disjoint communities.

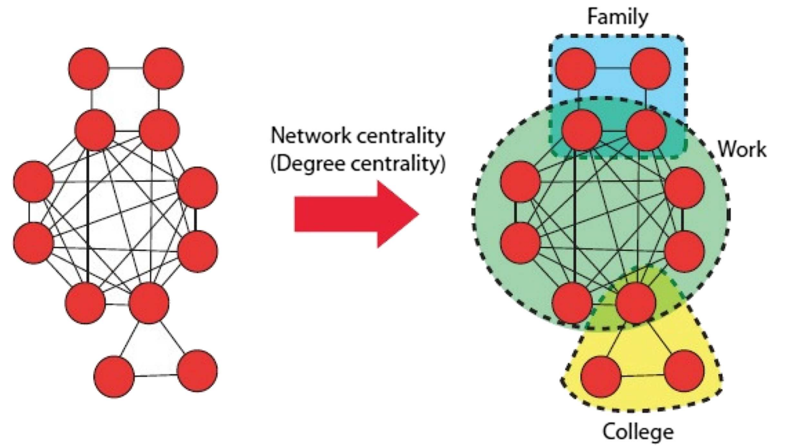


Fig. 3 A social network with overlapping communities

Centrality measures are used in network science to rank the relative importance of vertices and edges in a graph. Within graph theory and network analysis, there are various measures of the centrality of a vertex or an edge. Centrality indices are quantifications of the fact that some nodes/edges are more central or more important in a network than others [28]. Our algorithm uses the network centrality known as degree centrality to find overlapping community structure.

Degree centrality The simplest centrality for a vertex is its node degree, i.e., the total number of edges incident upon a node. This centrality represents the connectivity of a node to the rest of the network and reflects the immediate chance for a node to exert its influences to the rest of the network or to be exposed to whatever is flowing through the network, such as disturbances, shared information, power or traffic flows, or even a virus. For a graph with $G_n = \{V, E\}$, where V represents the set of vertices and E the set of edges, given its Laplacian L the degree centrality of a vertex or node is defined as [28],

$$C_i^D = \frac{D_i}{2n_E} = \frac{L_{ii}}{2n_E} \tag{8}$$

Where $2n_E$ is used as a normalization factor. In order to make better comparisons between graphs of different sizes the degree is standardized by dividing by $2n_E$, the maximum possible degree of any node.

Robustness refers either to the structural integrity of the network following deletion of nodes or edges or to the effects of perturbations on local or global network states. As shown in example network in Fig. 4 node 8 is most central but not robust. Network robustness and centrality plays vital role under circumstances of perturbations [28, 29].

Modularity The modularity is the fraction of edges that fall within communities, minus the expected value of the same quantity if edges fall at random without regard for community structure [30]. Several optimization methods attempt to optimize the " modularity" of a partition of the network [30]. Many complex networks consist of a number of modules. Each module contains several densely interconnected nodes, and there are relatively few connections between nodes in different modules. Hubs can therefore be described

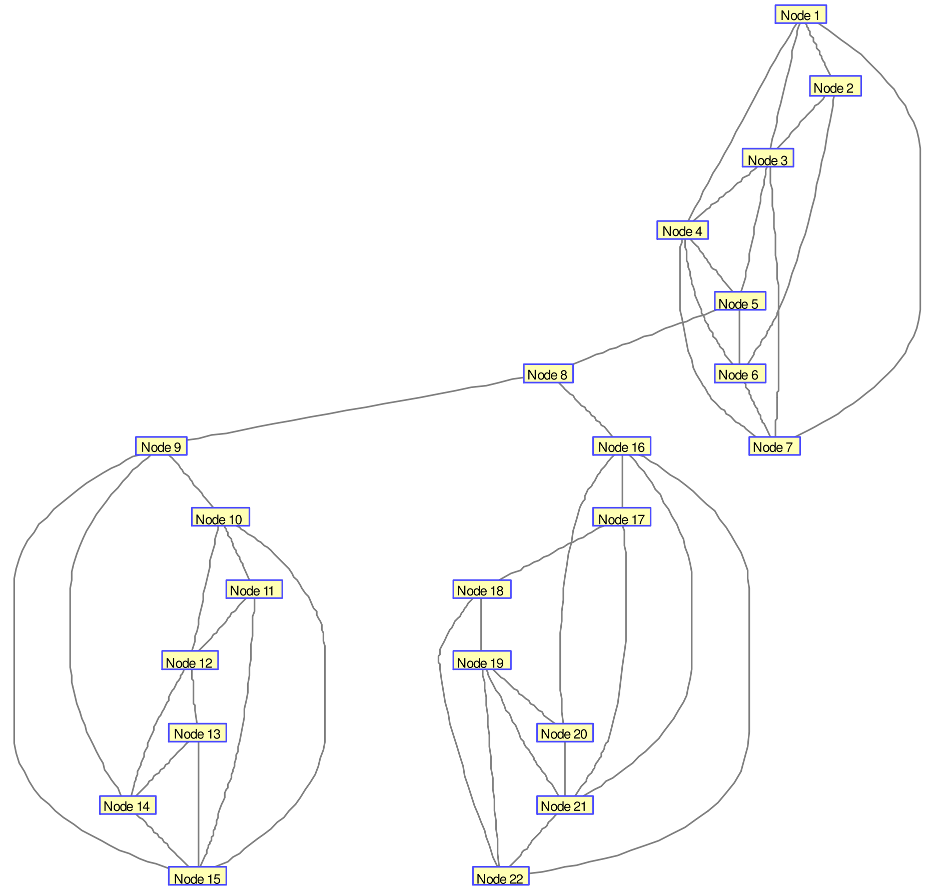


Fig. 4 Example network for indentifying network robustness and centrality. (Central, but not robust or powerful)

in terms of their roles in this community structure [22]. Provincial hubs are connected mainly to nodes in their own modules, whereas connector hubs are connected to nodes in other modules as shown in Fig. 5 [22].

Network density or cost Network or Connection density is the actual number of edges in the graph as a proportion of the total number of possible edges and is the simplest estimator of the physical cost, for example, the energy or other resource requirements, of a network.

We use above discussed complex network structural parameters for supervised community detection. Research methodology and algorithms are discussed in next section.

Methodology

As briefly discussed in the Related Work section, our research process and methodology consists of structural and functional analysis. To account for structural analysis we already discussed various complex network structural parameters in the Our Approach section.

Analysis of unweighted network (structural analysis)

A network can be defined as an object composed of elements and interactions or connections between these elements. A graph, $G_n(V, E)$, made up of node set, V, and link set,

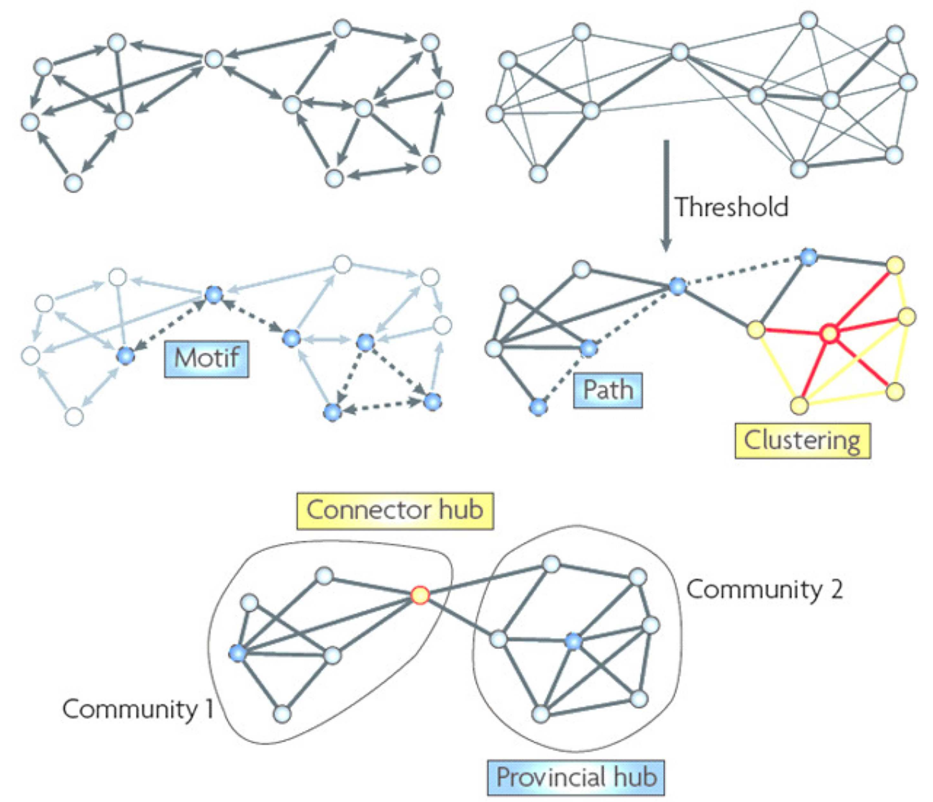


Fig. 5 Supervised community detection process [22]

E, is a natural means to model networks mathematically. Consider a graph with N nodes and m links or edges. The line-node incidence matrix of the network, is an $m \times N$, matrix M where the l^{th} edge is connected between nodes i and j if and only if

$$M : \begin{cases} M_{li} = & 1 \\ M_{lj} = & -1 \\ M_{lk} = & 0, \quad \text{with } k \neq i \text{ or } j \end{cases} \tag{9}$$

The Laplacian matrix L of the network [31], with size $N \times N$, can be obtained as

$$L = M^T M \tag{10}$$

Then

$$L_{ij} : \begin{cases} -1, & \text{if there exists link } i - j, \ \text{for } j \neq i \\ k & \text{with } k = -\sum_{j \neq i} L_{ij}, \qquad \text{for } j = i \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

with $i, j = 1, 2, \cdots, N$. Moreover, L is positive-semidefinite, real symmetric and the elements of every row (or column) add to zero. Alternatively:

$$L = D - A \tag{12}$$

A normalized Laplacian is stated as

$$\overline{L} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} \tag{13}$$

where $D = \operatorname{diag}(L)$ is the diagonal degree matrix of the network, matrix D be defined as

$$D_{ij} = \begin{cases} D(i) & \text{for } i = j \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

and A is the $N \times N$ adjacency matrix. The adjacency matrix can be written as

$$A_{ij} : \begin{cases} 1 & \text{if there is an edge from vertex } i \text{ to vertex } j\,, \\ 0 & \text{if there is no edge from vertex } i \text{ to vertex } j\,. \end{cases} \tag{15}$$

Here $A_{ij} = k$ if there are k parallel edges from i to j. Moreover, $D_i \geq 0$, $D_{ii} =$ number of edges connected to node i. Note that the diagonal elements of the Laplacian are assumed to be positive.

Eigenvalues of matrices in a graph, especially the adjacency matrix, the Laplacian matrix and the normalized Laplacian matrix reflect structural properties about the graph. For instance, adjacency matrix is useful for counting paths of certain length in a graph, number of spanning trees and connected components can be determined from the Laplacian, and the normalized Laplacian enables recognition of connected components and bipartite structures [32].

Analysis of weighted networks (functional analysis)

For a purely topological representation of a simple graph (with no parallel or self loops), the graph-theoretic matrices satisfy the following properties.

- The adjacency matrix A is real, symmetric, and zero on the diagonal, with entries being either 0 or 1. Since the trace is zero, then some of the eigenvalues must be positive and others must be negative, and hence this matrix is not sign-definite. It is obtained from the Laplacian matrix after zeroing its diagonal elements.
- The Laplacian matrix L is real symmetric and the sum of each row is zero. The diagonal elements are nonnegative, and the off-diagonal elements are nonpositive, either 0 or -1.
- The degree matrix D is a matrix with diagonal elements equalling either 0 or 1.

If parallel links are allowed between nodes, then nonzero entries can have integer values higher than 1 but of the same sign. If self loops are allowed, the adjacency matrix can have nonnegative integer diagonal elements. In any case, the matrices are related by Eq. 12, $L = D - A$.

Here we lift the restriction that the elements be binary or integers, thus leading to definitions of the **pseudo-adjacency**, **pseudo-Laplacian**, and **pseudo-degree matrices**. We will use the above guidelines to define **pseudo-adjacency**, **pseudo-Laplacian**, and **pseudo-degree matrices** for the weighted networks. The value of weights considered as power flow in smart grid network, signal or data flow in communication or data networks, information flow in social networks, money flow or transactions in financial networks, traffic flow in internet networks, money, weapons, drugs transactions in terrorists network etc. These matrices are required to maintain the basic structure and property of their graph-theoretic counterparts. In particular:

- The pseudo-adjacency matrix $\tilde{A}$ is real, symmetric, and zero on the diagonal, with nonnegative entries. Since the trace is zero, then some of the eigenvalues must positive and others must be negative, and hence this matrix is not sign-definite.

- The pseudo-Laplacian matrix $\tilde{L}$ is real symmetric and the sum of each row is zero. The diagonal elements are nonnegative, and the off-diagonal elements are nonpositive.
- The pseudo-degree matrix $\tilde{D}$ is diagonal with nonnegative diagonal elements. The sum of the diagonal elements is twice the total susceptance of all the lines in the system.

Similarly we require that

$$\tilde{L} = \tilde{D} - \tilde{A} \tag{16}$$

Note however that entries need not be integers or -1, 1, 0.

A normalized Laplacian is

$$\bar{L} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} \tag{17}$$

$$\bar{L} = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}, \tag{18}$$

where I is an $N \times N$ identity matrix (with ones on the diagonal, other elements being zero).

Normalized Laplacian in matrix form is written as

$$\bar{L}_{ij} : \begin{cases} 1, & \text{if } i = j, \\ -\frac{1}{\sqrt{k_i k_j}} & \text{if } ij \in E, \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

The Laplacian L of a directed network is defined as L_D

$$L_D = D_O^{-\frac{1}{2}} A D_I^{-\frac{1}{2}}, \tag{20}$$

where Do is the diagonal matrix of out-degrees (or row sum of A) and D_I is the diagonal matrix of in-degrees (or column sum of A).

For a **directed weighted** networks Eq. 20 is written as

$$\tilde{L}_D = \tilde{D}_O^{-\frac{1}{2}} \tilde{A} \tilde{D}_I^{-\frac{1}{2}}, \tag{21}$$

where $\tilde{A}$ is weighted adjacency matrix of directed networks, $\tilde{D}_O$ is the weighted diagonal matrix of out-degrees (or row sum of $\tilde{A}$) and $\tilde{D}_I$ is the weighted diagonal matrix of in-degrees (or column sum of $\tilde{A}$).

For incorporating functional analysis in order to consider flow or functional dynamics in the network we proposed modified relationship between adjacency and Laplacian of a graph given by Eq. 16.

This modified relationship turns modularity maximization into a spectral graph partitioning problem using the modified Laplacian matrix. A nice feature of the modified Laplacian is that, for graphs which are not too small, it can be approximated (up to constant factors) by the transition matrix $\tilde{A}_x$, obtained by normalizing $\tilde{A}$ such that the sum of the elements of each row equals one.

Weighted laplacian centrality Using modified relationship obtained in Eq. 16 we then obtained modified network or degree centrality as given by Eq. 22.,

$$(C_i^D)_F = \frac{(\tilde{L})_{ii}}{2n_E} \tag{22}$$

Using Eq. 22 we will get centrality of the functional network. We used this functional degree centrality to determine robustness of the network.

Definition 1. (Micro-Community). The micro-community is a small dense group or a sub-graph or isolated node that consists of one or more connected dense network or pairs with certain energy.

Network energy is defined as: the sum of the absolute values of the real components of the eigenvalues,

$$ne(G_n) = \sum_{i=1}^{N} |\lambda_i| \tag{23}$$

For a network $G_n(V, E)$, the local micro-community μc

$$\mu c = c(l_1) = (V_{l_1}, E_{l_1}, ne_{l_1}) \tag{24}$$

where V_{l_1} vertices of sub or dense network, E_{l_1} edges of sub or dense network and ne_{l_1} is the energy of local sub or dense network.

The micro-community clusters μcc are given by

$$\mu cc = \{c(l_1), c(l_2), c(l_3), ...c(l_n)\} \tag{25}$$

For a given community network, to partition it into a certain number of smaller sub-communities or number of subsets, called *clusters*.

Weighted micro-community or sub-community centrality Smaller sub-communities are given more weight than larger ones, which makes this measure appropriate for characterizing network motifs. The sub-community centrality can be obtained mathematically from the spectra of the weighted adjacency matrix of the network. The sub-community centrality of a node is a weighted sum of closed walks of different lengths in the network starting and ending at the node. This function returns a vector of sub-community centralities for each node of the network [33, 34].

Definition 2. (Micro-Community Centrality).

For a graph, $G_n(V, E)$, let $v_{1,}v_{1,}.....v_N$ be an orthogonal basis of R^N composed by eigenvectors of weighted adjacency matrix $\tilde{A}$ associated to the eigenvalues $\lambda_{1,}\lambda_{2,}.....\lambda_{N.}$ Let v_j^i denote the i^{th} component of $v_{j.}$ For all $i \in V$, the sub-community centrality is expressed as [33],

$$\left(C_i^S\right)_F = \sum_{i=1}^{n} \left(v_j^i\right)^2 e^{\lambda_j} \tag{26}$$

For all methods and approaches discussed above Micro-Community Centrality (MCC) network robustness algorithm is developed. Overall process of MCC is described in Algorithm 1. For any given large-scale community network G_n. First it identifies type of network i.e. Directed Unweighted (DU), Directed Weighted (DW), Undirected Unweighted (UU), Undirected Weighted (UW). As per the type of network then it calculates all required statistical parameters from adjacency A, Laplacian L and degree matrices D and similarly for weighted matrices i.e. $\tilde{A}$, $\tilde{L}$, and $\tilde{D}$ etc. Then it calculates network energy, micro-community and micro-community clusters. With these parameters it then calculate weighted Laplacian centrality and weighted micro-community centrality. Using algebraic connectivity it check for robustness of the network i.e. whether network is strongly connected or weakly connected.

Algorithm 1 : Micro-community centrality and network robustness

```
Input: $G_n$ the initial network (Dataset)
Returns: Micro-Community Centrality, Algebraic Connectivity
Identify type of network
Directed Unweighted (DU), Directed Weighted (DW),
Undirected Unweighted (UU), Undirected Weighted (UW)
As per the type of network
Compute
  Adjacency matrix $A_{ij}$,
  In, Out Node Degree $D_i$ and Degree Distribution$P(k)$,
  Avg. Degree Distribution,
  Degree matrix$D$,
  $L \leftarrow D - A$, and $\tilde{L} \leftarrow \tilde{D} - \tilde{A}$
  $L_D \leftarrow D_O^{-\frac{1}{2}} A D_I^{-\frac{1}{2}}$, and $\tilde{L}_D = \tilde{D}_O^{-\frac{1}{2}} \tilde{A} \tilde{D}_I^{-\frac{1}{2}}$,
  $\lambda_2(L)$, Avg. path length $l$, Shortest path length $d_{ij}$,
  Local clustering coefficient $CC_i(local)$,
  Global clustering coefficient $CC(global)$,
  Network Energy $ne(G_n)$, Micro-community $\mu c$, Micro-community clusters $\mu cc$
  Weighted Laplacian centrality $\left(C_i^D\right)_F = \frac{(\tilde{L})_{ii}}{2n_E}$
  Weighted Micro-Community Centrality $\left(C_i^S\right)_F$
     arranged $v_{1,}v_{1,}.....v_N$ by the descending order of their
    micro-community centrality $C_1^S, C_2^S, , ...C_N^S$.
Check for the Network Robustness
EV=sort $(L,'\ descend')$;
 $N = length(G_n)$
eps=0.01;
if $(N >= 2)\&\&(abs(EV(N)) < eps)$
  Algebraic Connectivity $= EV(N-1)$;
  return;
else
  Algebraic Connectivity $= -1$;
  return;
end
return;
```

K-clique sub-community: degree and weighted micro-community centrality based overlapping community algorithm Most real networks typically contain parts in which the nodes (units) are more highly connected to each other than to the rest of the network. The sets of such nodes are usually called clusters, communities, cohesive groups, or modules [35]. Most real networks are characterized by well defined statistics of overlapping and nested communities. Such a statement can be demonstrated by the numerous communities each of us belongs to, including those related to our scientific activities or personal life (family, work, college) and so on [35], as illustrated in Fig. 3.

Definition 3. A typical community consists of several complete (fully connected) sub-communities that tend to share many of their nodes. Thus, we define a sub-community, or more precisely, a *k-clique-sub community* as a union of all *k-cliques* (complete subgraphs of size *k*) that can be reached from each other through a series of adjacent *k*-cliques (where adjacency means sharing $k - 1$ nodes).

Proposed algorithms (Algorithm 2 and 3) firstly extracts all complete weighted sub-communities of the network that are not parts of larger complete sub-communities. A maximal clique is a clique that is not a subset of any other clique in a community network [36]. These maximal complete subgraphs are simply called cliques, and the difference between *k*-cliques and cliques is that *k*-cliques can be subsets of larger complete sub-communities. Once the cliques are located, the clique-clique overlap matrix is prepared [37]. In this symmetric matrix each row (and column) represents a clique and the matrix elements are equal to the number of common nodes between the corresponding two cliques, and the diagonal entries are equal to the size of the clique. The intersection of two cliques is always a complete sub-communities. The *k*-clique-communities for a given value of *k* are equivalent to such connected clique components in which the neighbouring cliques are linked to each other by at least $k - 1$ common nodes. Advantage of this method is that the clique-clique overlap matrix encodes all information necessary to obtain the communities for any value of *k*, therefore once the clique-clique overlap matrix is constructed, the *k*-clique-communities for all possible values of *k* can be obtained very quickly [35]. Algorithm 2 describes the process of finding maximum *s*-size *k*-cliques in the community network. It uses degree sequence for finding largest possible clique size.

Algorithm 2 : Maximum ***s***-Size ***k***-Cliques in the Community Network

```
Input: $G_n$ the initial network (Dataset)
Returns: Maximum $s-$size $k$-cliques
Number of nodes N =size($G_n$, 1)
Find the largest possible clique size via the degree sequence
Let $\{d_1, d_2, ..., d_k\}$ be the degree sequence of a graph.
The largest possible clique size of the graph is the
maximum value k such that $d_k >= k - 1$
degree_sequence = sort(sum($G_n$, 2) $-$ 1,'descend');
$s_{max} = 0$;
for $i = 1$ : length(degree_sequence)
    if degree_sequence($i$) $>= i - 1$
        $s_{max} = i$;
    else
        break;
    end
end
cliques = cell(0);
for $s = s_{max} : -1 : 3$
    $G_n aux = G_n$;
    for $N = 1 : Nbn$
```

```
        $X = N$;
        $Y$ =setdiff(find($G_n aux(N,:) == 1), N$);
        Enlarging X by transferring nodes from Y
        $Z$ = ttransfer_nodes $(X, Y, s, G_n aux)$;
        if ~ isempty (Z)
            for $i$ = size $(Z, 1)$
                cliques = [ cliques;{$Z(i,:)$}] ;
            end
        end
        $G_n aux(N,:) = 0$;
        $G_n aux(:,N) = 0$;
    end
end
```

For detecting overlapping communities Algorithm 3 is developed. It uses weighted adjacency matrix, weighted micro-community centrality and maximum *s*-size *k*-cliques in the community network (With Algorithm 2). First it generates the clique-clique overlap matrix. Then extracts the *k*-clique matrix *kM* from the clique-clique overlap matrix and *k*-clique sub-communities *cc* from the *k*-clique matrix *kM*.

Algorithm 3 : Overlapping Community Detection

```
Input: $G_n$ the initial network (Dataset)
Returns: k-clique sub-communities cc, all cliques, k-clique matrix kM
    Number of nodes N=size($G_n$)
    Compute Weighted micro-community centrality $(C_i^S)_F$
    $(C_i^S)_s$ =sort($C_i^S$,'descend')
    Find all maximum s-size k-cliques in the community network using Algorithm 2
    Generating the clique-clique overlap matrix
    kM = length(cliques)
    for $c_1 = 1$: length(cliques)
        for $c_2 = c_1$: length(cliques)
            if $c_1 = c_2$
                $kM(c_1, c_2)$ = Number of array elements(cliques{$c_1$});
            else
                $kM(c_1, c_2)$ = Number of array elements(cliques{$c_1$} $\cap$ cliques{$c_2$}));
                $kM(c_2, c_1) = kM(c_1, c_2)$;
            end
        end
    end
Extracting the k-clique matrix kM from the clique-clique overlap matrix
Off-diagonal elements $<= k - 1 \rightarrow 0$
Diagonal elements $<= k \rightarrow 0$
Extracting components (ork-clique sub-communities cc) fromthe k-clique matrix kM
    Sub-community cc =[ ] ;
    for $i = 1$:length(cliques)
    linked_cliques = find ($kM(i,:) == 1$);
    new sub-community $cc_n$ =[ ] ;
    for $j = 1$ : length(linked_cliques)
        new sub-community $cc_n = (cc_n \cup$ cliques{linked_cliques(j)});
    end
    found = false;
    if ~ isempty ($cc_n$ )
        for $j = 1$ : length(cc)
            if all(ismember($cc_n$, cc{j}))
                found = true;
            end
        end
        if ~found
            $cc = [\,cc; \{cc_n\}]$ ;
        end
    end
end
```

Modified weighted modularity: optimal partition of *k*-clique sub-community We then used weighted adjacency matrix $\tilde{A}$ to derive functional modularity of the network. For a simple, undirected graph G_n and a partition C with a given number of groups or number of communities n, the modularity measure $Q(G_n, C)$ is defined as [38]:

$$Q(G_n, C) = \sum_{i=1}^{n} (e_{ii} - a_i^2) \tag{27}$$

where the network is fully subdivided into a set of nonoverlapping communities n, and e_{ij} is the proportion of all links that connect nodes in community i with nodes in community j.

with

$$e_{ij} = \frac{\sum_{v_x \in C_i} \sum_{v_y \in C_j} A_{ij}}{2\,|E|} \tag{28}$$

where A adjacency matrix which is symmetric and set of edges E. With modified adjacency matrix $\tilde{A}$, Eq. 28 changes to

$$e_{ij} = \frac{\sum_{v_x \in C_i} \sum_{v_y \in C_j} (\tilde{A})_{ij}}{2\,|E|} \tag{29}$$

and the proportion of edges with at least one node in the community i is given by

$$a_i = \sum_j e_{ij} \tag{30}$$

e: The $N \times N$ symmetric weighted matrix of the partition C.

e_{ij}: The fraction of edges between clusters C_i and C_j.

e_{ii}: The fraction of edges in cluster C_i. (i.e. the portion of edges that connect vertices inside community C_i).

Assuming the network is divided into n communities. Let us define C_i and C_j be the communities which belong to vertices i and j respectively. Node i belong to community

C_i, the fraction of edges that fall between community i and community j is defined as:

$$e_{ij} = \frac{1}{2m} \sum_{ij} A_{ij} \delta(C_i, i) \delta(C_j, j), \tag{31}$$

where the δ is a function $\delta(i, j)$ and m is again the number of edges in the network.

$$\delta(i, j) = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are the same community} \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

Let P_{ij} be the expected number of edges between i and j.

$$P_{ij} = \frac{k_i k_j}{2m} \tag{33}$$

where $k_i k_j$ are degrees of vertex i and vertex j.

The actual number of edges falling between a particular pair of vertices i and j is A_{ij}. The modularity matrix is defined as

$$B_{ij} = A_{ij} - \frac{k_i k_j}{2m} \tag{34}$$

Alternatively Eq. 34 can be written as

$$B_{ij} = A_{ij} - P_{ij} \tag{35}$$

Important property of modularity matrix is that all rows (and columns) of the modularity matrix sum to zero i.e.,

$$\sum_j B_{ij} = \sum_j A_{ij} - \sum_j P_{ij} = k_i - k_i = 0 \tag{36}$$

Like Laplacian matrix for any network the vector $(1, 1, 1, \ldots)$ is an eigenvector of the modularity matrix with eigenvalue zero but the eigenvalues of the modularity matrix are not necessarily all of one sign i.e. matrix has both positive and negative eigenvalues [39].

Modularity measures the non-randomness of a graph partition. Higher values of the modularity indicate stronger community structures. The modularity maximization problem is then:

$$\max_{P \in \Omega} Q(G_n, C) \tag{37}$$

Then the MaxModularity can be written

$$Q_{\max} = \frac{1}{2m} \sum_{ij} [A_{ij} - P_{ij}] \, \delta(C_i, C_j), \tag{38}$$

$$Q_{\max} = \frac{1}{2m} \sum_{ij} [B_{ij}] \, \delta(C_i, C_j), \tag{39}$$

Using weighted adjacency matrix weighted modularity can be written as

$$Q^W_{\max} = \frac{1}{2m} \sum_{ij} \left[\tilde{A}_{ij} - \frac{\overset{\sim}{k_i} \overset{\sim}{k_j}}{2m} \right] \delta(C_i, C_j), \tag{40}$$

where $\overset{\sim}{k_i}$ and $\overset{\sim}{k_j}$ are weighted degrees.

For directed weighted network modularity is given as

$$\overset{\rightarrow W}{Q}_{\max} = \frac{1}{2m}\sum_{ij}\left[\tilde{A}_{ij} - \frac{\overset{\sim}{k_i^I}\overset{\sim}{k_j^O}}{2m}\right]\delta(C_i, C_j), \tag{41}$$

A measure for the modified modularity is proposed to quantify the overlapping community structure referred as Q_M^W (Weighted modified modularity). With the measure Q_M^W, the overlapping community structure can be identified by finding an optimal partition of k-clique sub-community, i.e., the one with the maximum Q_M^W. The Q_M^W is based on a maximal clique view of the original network. A maximal clique is a clique (i.e. a complete subgraph) which is not a subset of any other clique in a network. The maximal clique view is according to a reasonable assumption that a maximal clique cannot be shared by two communities due to that it is highly connective. To find an optimal partition, we construct a maximal clique network from the original network. We then prove that the optimization of Q_M^W on the original network is equivalent to the optimization of the modularity on the maximal clique network. Thus the overlapping community structure can be identified through partitioning the maximal clique network with an efficient modularity optimization Algorithm [40].

The proposed overlapping community structure based on optimal partition of k-clique sub-community is stated as

$$\overset{\rightarrow W}{Q}_{M\max} = \frac{1}{2m}\sum_{ij}\frac{1}{C_i, C_j}\left[\tilde{A}_{ij} - \frac{\overset{\sim}{k_i^I}\overset{\sim}{k_j^O}}{2m}\right], \tag{42}$$

where C_i, C_j are the number of overlapping communities to which node i and node j belongs. High value of $\overset{\rightarrow W}{Q}_{M\max}$ indicates a significant overlapping community structure.

In our implemented Algorithm 4 given below we used Fast Newman Greedy algorithm for modularity optimization [41] with modified functional parameters. In order to efficiently detect community structure using complex network structural and functional parameters listed above we developed an Algorithm 5 for modified modularity for overlapping community detection.

Algorithm 4 : Modularity Maximization

```
$G_n(V, E)$ the initial network
repeat
Put each node of $G_n$ in its own community
Calculate $Q^W$ from pairs of connected communities
  while some nodes are moved do
     for all $N$ node of $G_n$ do
     place N in its neighboring community including its own
      while maximal $Q^W > 0$ do
       select the maximal $Q^W$, join the pair of communities with the maximal $Q^W$
      which maximizes the modularity gain $Q^W$
      update the $Q^W$ matrix
    end while
```

```
        end for
    end while
      if the new modularity is higher than the initial
      then
    $G_n$ = the network between communities of $G_n$
    else
      Terminate
    end if
    until $Q^W = 0$.
```

It measures modularity variation for each candidate partition where pair of clusters are merged. It merges the pair of clusters by maximizing modularity Q using Algorithm 4. So for each formed clusters it splits community and then updates corresponding Q. For each sub-community then it measures sub-community energy *ne*, micro-community centrality using overlapping community detection Algorithm 3. Then it selects sub-community with highest Q and highest *ne* to find k-cliques sub-community network to form micro-community clusters μcc. These micro-community cluster formation continues till value of Q is 0 i.e. leading eigenvalue is zero which means that subgraph is indivisible. Overall process of modified modularity for overlapping community is described in Algorithm 5.

Algorithm 5 Modified modularity for overlapping community detection (MMOC)

```
Input: $G_n$ the initial network (Dataset)
Returns: Community Clusters cc
Identify type of network
Directed Unweighted (DU), Directed Weighted (DW),
Undirected Unweighted (UU), Undirected Weighted (UW)
As per the type of network
Compute Community c and Modularity $Q^W$
  Apply overlapping community detection (Algorithm 3)
  Apply modularity optimization (Algorithm 4)
  Measure modularity variation $Q^W$ for each candidate
  partition where a pair of clusters are merged
Compute $\overset{\to W}{Q}_{O\max}$
Repeat
for each clusters
    $\overset{\to W}{Q}_{O\max}$ ←Merge the pair of clusters maximizing $Q^W$
    Update $\overset{\to W}{Q}_{O\max}$ ← Split c
        if $Q^W = 0$
        (leading eigenvalue is zero, subgraph is indivisible) then
            break;
        end if
end for
return cc
```

This modified modularity for overlapping community algorithm has several advantages. First, its steps are intuitive and easy to implement. Moreover, the algorithm is extremely fast, i.e., network simulations on large-scale ad-hoc modular networks found that its complexity is linear on typical and sparse data. Experimental evaluation of these algorithms for complex technological networks and social networks are discussed in next Section result and discussion on analysis of real-world large-scale complex networks.

Results and discussion

Analysis of real-world large-scale complex networks

In this section we analyze real world large-scale complex network using proposed algorithms discussed above. We used MATLAB version R2015a [42] with Intel, Xeon(R) 2.60 GHz, 256 GB RAM 2 processors, GPU Quadro K6000 and Tesla K20c for running these algorithms. In a simple random graph G_n, degree will have a Poisson distribution, and the nodes with high degree are likely to be at the intuitive center. Deviations from a Poisson distribution suggest non-random processes, which is at the heart of current "scale-free" work on networks. Figure 6 shows degree distribution of directed weighted Facebook social network with 1899 nodes and 20296 links [43].

Figure 7 shows degree centrality for directed Amazon product co-purchasing network from March 2, 2003 with 262111 nodes and 1,234,877 links [43]. As shown in this figure network follows the power law of scale free network.

Centrality measures and power have become common emphasis for world city network research and frequently serve as tools for describing cities' position or status in the system. We experimented weighted bipartite graph of world city network. Figure 8 shows the out degree distribution for world city system. Figure 9 shows degree distribution of directed web graph from Google (Data obtained in 2002) with 916428 nodes and 4333051 links [43]. Node represent web pages and directed edges represent hyperlinks between them.

Then we experimented with other real-world complex networks including complex critical infrastructure U.S. WECC power grid with 4941 nodes and 6594 links [44, 45]

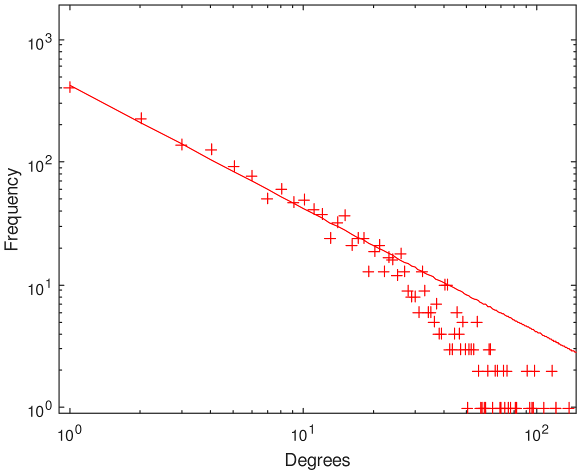


Fig. 6 Degree distribution of Facebook social network

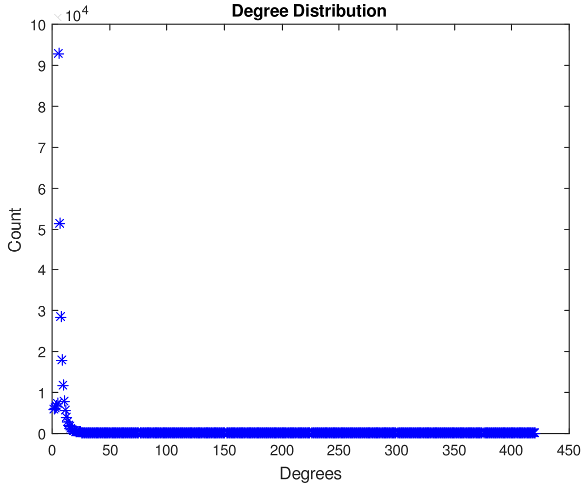


Fig. 7 Degree centrality for directed amazon product co-purchasing network

and other social community, as well as citation networks. For these networks we applied MMOC and other algorithms (Algorithms 1 to 5 and parameter $k = 3$). Table 1 shows analysis for these networks. As seen from the results MMOC algorithm plays major role in overlapping community detection in complex networks. We can see relationship of algebraic connectivity *ac* and network energy *ne*. These values shows how strong or how weak the overall network is. This MMOC algorithm plays decisive role for directed weighted, unweighted as well as undirected weighted networks. As seen from results in case of Facebook network with 1899 nodes. (1899 users that sent or received total of 59,835 online messages over 20,296 directed ties among these users) MMOC algorithm identifies 512 communities and 353 overlapping communities based on topics, social areas of interests,

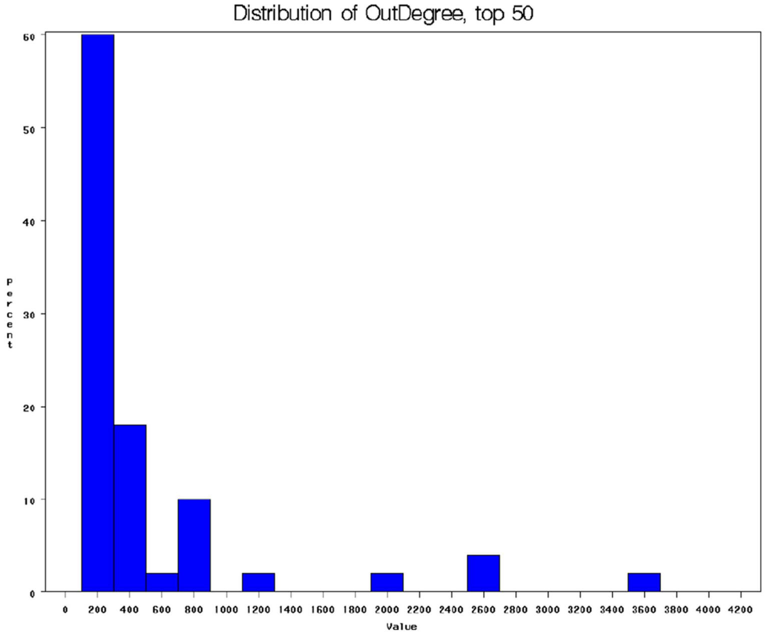


Fig. 8 Out degree distribution for world city system

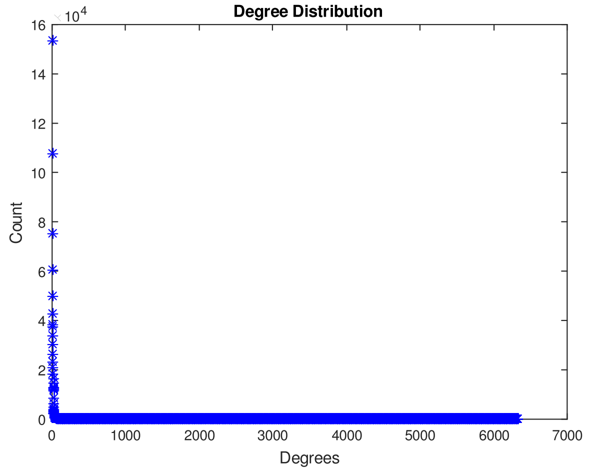


Fig. 9 Web graph from Google

etc. Similarly for PhD in Computer Science it identifies 189 communities and 4 overlapping communities and for SciMet citation network directed multigraph with 3084 nodes and 10413 links it identifies 650 overlapping communities. Overlapping community clusters (*cc*) obtained with MMOC algorithm clearly shows how this algorithm identifies dense, deeper and hidden community structures. For social community networks also we can see the same relationship of algebraic connectivity *ac* and network energy *ne*.

Comparison of different community detection algorithms

We compared modularity optimization values obtained using MMOC algorithm with other existing algorithms for real world large-scale big data networks as shown in Table 2.

We plotted values obtained using different modularity optimization algorithms as shown in Table 2. Figure 10 shows modularity comparison for existing algorithms and with MMOC algorithm (Algorithms 4 and 5).

Table 1 Analysis of complex social and technological networks

Networks →	PhD's CS	Facebook	SciMet	U.S. Power Grid
Analysis Parameters ↓				
Type of Network	Directed	Directed Weighted	Directed multigraph	Undirected
V	1882	1899	3084	4941
E	1740	20296	10399	6594
Avg k	40.913	5.6962	16.6402	260.0526
CC_{global}	0.0051	0.1107	0.1703	0.0801
ac	20.2106	115.9189	77.3748	15.0674
ne	34.18	109.3126	96.1957	35.5106
c	189	512	391	35
cc (oc)	**4**	**353**	**650**	**307**

V: Number of nodes or vertices, *E*: Number of links or edges, *Avg k*: Average node degree, CC_{global} : Global clustering coefficient, *ac*: Algebraic connectivity, *ne*: Network energy, *c*: Sub-communities, *cc*: Overlapping community clusters (For *k*-cliques = 3)., **PhD's CS:** PhD's in computer science, directed graph with 1882 nodes and 1740 links. **Facebook:** The Facebook-like Social Network originate from an online community for students at University of California, Irvine. The dataset includes the users that sent or received at least one message (1,899). A total number of 59,835 online messages were set over 20,296 directed ties among these users. **SciMet:** SciMet citation network directed multigraph with 3084 nodes and 10413 links. **US Power Grid:** US power grid undirected graph with 4941 nodes and 6594 links *(Note: Datasets obtained from iLab Big Data Center, North Carolina A&T State University* [43]*)*

Table 2 Modularity comparison

Algorithms→		FN	DGA	FD	MSTAB	MMOC
Networks↓	Size↓	*Q* FN	*Q* DGA	*Q* FD	*Q* MSTAB	(*Q* Our method)
PhD's in CS	1882	0.9610	0.9610	0.9295	0.9601	0.9755
Facebook	1899	0.2717	0.2567	0.3751	0.3742	0.3860
SciMet	3084	0.5469	0.5949	0.6146	0.6146	0.6502
US Power Grid	4941	0.9341	0.9358	0.9347	0.9348	0.9587

FN: Fast Newman based on a greedy agglomerative method. **DGA:** Modularity optimization based on Danon greedy agglomerative method. **FD:** Fast detection of communities using modularity optimization. **MSTAB:** Modularity based on stability. **MMOC**: Modified Modularity for Overlapping Community Detection (Our method).

Modularity maximization achieved with MMOC algorithm helps for detection of dense, hidden micro level communities. These results clearly indicate the importance of modularity maximization even though it is NP-complete problem.

Also for overlapping community analysis shown in Table 1, we plotted comparison based on communities obtained with existing algorithm and overlapping communities clusters obtained with MMOC algorithm. Figure 11 shows overlapping community detection analysis for complex technological and social community networks. These values clearly shows significant difference between base communities and overlapping communities for both complex technological networks as well as social community networks. The parameter *k* affects the constituent of the overlapping regions between communities. The choice of the parameter *k* depends on the specific networks. Observed from many real world networks, the typical value of *k* is often between 3 and 6 [40].

From these results it showed that community centrality appears to have relation with vertices that are central in their local communities. The centrality is correlated with degree, for few overlapping communities they are not perfectly correlated and in particular some vertices have quite high centrality while having relatively low degree. High centrality is an indicator of individuals who have more connections than expected within their neighborhood and hence potentially make a large contribution to the modularity, rather than simply having a lot of connections.

Computational complexity

The determination of the full set of cliques of a network is widely believed to be non-polynomial problem. In spite of this, proposed algorithm proves to be very efficient when

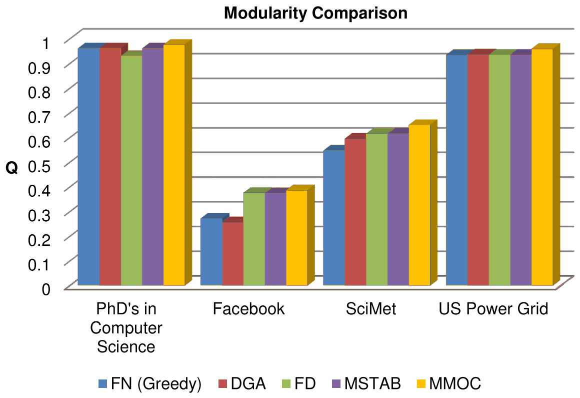
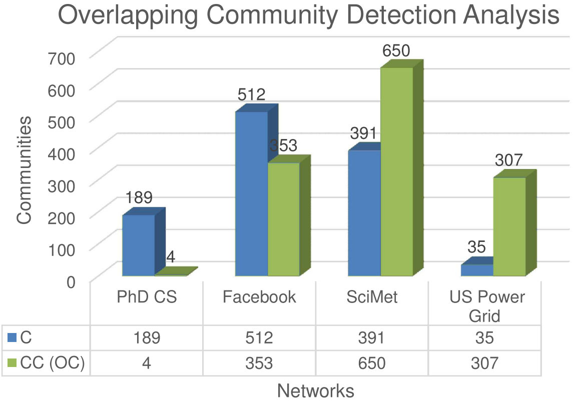


Fig. 10 Modularity comparison



	PhD CS	Facebook	SciMet	US Power Grid
🟦 C	189	512	391	35
🟩 CC (OC)	4	353	650	307

Fig. 11 Overlapping community detection analysis

applied to the graphs of the investigated real systems. Our method consists of five stages, finding degree sequence, micro-community centrality, finding out the maximal cliques, constructing the maximal clique network and overlapping community network matrix and partitioning the maximal clique network based on the modularity maximization and then finding overlapping communities.

We analyze the computational complexity of MMOC and other algorithms (Algorithms 1 to 5). Finding an exact solution to a partitioning task of this kind is believed to be an NP-complete problem, making it prohibitively difficult to solve for large-scale networks, but a wide variety of heuristic algorithms have been developed that give acceptably good solutions in many cases. The first algorithm of the modern age of community detection introduced by Newman and Girvan has a complexity $O(N^3)$ on a sparse networks and other mentioned existing algorithms for detecting community structures gives qualitatively similar results. Fast implementation of Newman algorithm (Fast Newman algorithm) has worst-case running time of $O((m + N)N)$, or $O(N^2)$ on sparse network with N nodes and m edges. Experimental evaluation on the real-world complex technological and social community networks show that MMOC algorithm achieves the best performance when compared with other existing methods discussed in Table 2. Efficient time complexity of MMOC algorithm and other algorithm is $O(N \log N)$ which is scalable in nature. For MMOC algorithm running time is consumed by maximizing modularity and forming overlapping community matrix based on sub-community energy. Also in case of directed weighted networks running time is also consumed by computation of large eigen values. Our method is very efficient on real world networks. In our future work we will work for modifying our MMOC algorithm for better run time performance.

Conclusions

In this paper we have discussed community dynamics and reviewed complex network structural parameters. We highlighted the importance of network centrality or degree centrality and network robustness for community detection. Centrality is correlated with degree. We discussed network or degree centrality (weighted Laplacian centrality) based on modified Laplacian, weighted micro-community centrality. We also discussed and introduced algorithm for k-clique sub-community and optimal partition of k-clique sub-community for weighted modularity optimization and overlapping community detection based on degree and weighted micro-community centrality. These new matrices and algorithms are helpful in identifying hidden level vulnerabilities. We analyzed real-world large-scale complex networks and carried out comparison of different community detection algorithms. Our results indicated certain relationship between degree centrality and modularity optimization. Network centrality and robustness will help for supervised community detection in overlapping communities. Proposed algorithms will be useful for finding communities of densely connected vertices in network data. Computational complexity of our proposed algorithms is better as compared to other existing algorithms. Scalable nature of this algorithm is valuable for analyzing more complex large-scale networks.

It is also an interesting problem about the selection of the parameter k in our method. We will further investigate how to determine an appropriate k for a given network later. In our future work we will put forward functional dynamics of complex network by incorporating network centrality and weighted clustering coefficient for identifying micro level communities and their associated relationship.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

PC performed the primary literature review, mathematical modelling, research design, large-scale data collection, programming and experimental evaluation, and also drafted the manuscript. JZ supervised PC to develop the research methodology and computational complexity. Both authors read and approved the final manuscript.

Authors' information

Dr. Pravin Chopade is a Research Scientist at iLab, Department of Computer Science, College of Engineering, North Carolina A and T State University, Greensboro, NC, USA. His project is funded by Department of Defense (DoD). He completed his Ph.D. in the Department of Computational Science and Engineering (CSE) at the North Carolina Agricultural and Technical State University, Greensboro, USA in December 2013. His Ph.D. research work was on "Robustness and Survivability of Smart Power Grid and SCADA Networks when Subjected to Severe Emergencies, Vulnerability and WMD Attacks", funded by The Defense Threat Reduction Agency (DTRA) and Pennsylvania State University, USA. His major field of interest and research includes Big Data/Large-scale networks, Smart Grid, IntiGrid, GridStat, SCADA-EMS system design, protection and survivability of interconnected large-scale networks. He published and presented 40 research papers at various national and international peer reviewed journals and conferences. Pravin also received number of research grants and awards.

Dr. Justin Zhan is the director of ILAB. Department of Computer Science, College of Engineering, University of Nevada-Las Vegas, Las Vegas, NV, USA, 4505 S. Maryland Pkwy., Las Vegas, USA. His research interests include Big Data, Information Assurance, Social Computing, and Health Science. He is a steering chair of ASE/IEEE International Conference on Social Computing (SocialCom), ASE/IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT), and ASE/IEEE International Conference on BioMedical Computing (BioMedCom). He is currently an editor-in-chief of International Journal of Privacy, Security and Integrity, International Journal of Social Computing and Cyber-Physical Systems, and managing editor of SCIENCE journal and HUMAN Journal. He has served as a conference general chair, a program chair, a publicity chair, a workshop chair, or a program committee member for 160 international conferences and an editor-in-chief, an editor, an associate editor, a guest editor, an editorial advisory board member, or an editorial board member for 30 journals. He has published 180 articles in peer-reviewed journals and conferences and delivered above 30 keynote speeches and invited talks. He has been involved in a number of projects as a PI or a Co-PI, funded by the National Science Foundation, Department of Defense, National Institute of Health, etc.

Acknowledgment

We are thankful to The United States Department of Defense (DoD) for their support and finance for this project. This work is supported by The United States Department of Defense (DoD grants #W911NF-13-1-0130).

We are also thankful to The United States National Science Foundation (NSF) and The United States National Consortium for Data Science (NCDS). This research is partially supported by the following grants: NSF No. 1137443, NSF No. 1247663, NSF No. 1238767, DoD No. W911NF-14-1-0119, and the Data Science Fellowship Award by the National Consortium for Data Science.

Author details

[1] Department of Computer Science, College of Engineering, North Carolina A and T State University, Greensboro, NC, USA, 305 Cherry Hall, 1601 East Market Street, NC-27411, Greensboro, USA. [2] Department of Computer Science, College of Engineering, University of Nevada-Las Vegas, Las Vegas, NV, USA, 4505 S. Maryland Pkwy., NV-89154, Las Vegas, USA.

Received: 8 May 2015 Accepted: 12 June 2015

References

1. Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang DU (2006) Complex networks: Structure and dynamics. ELSEVIER, Physics Reports 424:175–308
2. Campbell W, Dagli C, Weinstein C (2013) Social Network Analysis with Content and Graphs. MIT Lincoln Laboratory Journal 20:62–81
3. Marsden P (2002) Egocentric and sociocentric measures of network centrality. Social Networks 24:407–422
4. Zio E, Piccinelli R (2010) Randomized flow model and centrality measure for electrical power transmission network analysis. Reliability engineering and System Safety 95:379–385
5. Newman M (January 2012) Communities, modules and large-scale structure in networks. Nature Physics 8:25–31. doi:10.1038/NPHYS2162
6. Fortunato S (2010) Community detection in graphs. Physics Reports 486, no no. 3-5:75–174
7. Bollobas B (2001) Random graphs. Cambridge university press 2:1–496
8. Watts D, Strogatz S (June 4 1998) Collective dynamics of 'small-world' networks. Nature 393:440–442
9. Barabasi A (Oct 15 1999) Emergence of scaling in random networks. Science 286:509–512
10. Newman M (2003) The structure and function of complex networks. Siam Review 45:167–256
11. Clauset A (2009) Power-Law distributions in empirical data. Siam Review 51:661–703
12. Crucitti P (Sep 1, 2004) Error and attack tolerance of complex networks. Physica a-Statistical Mechanics and Its Applications 340:388–394
13. Koschutzki D, Lehmann K, Peeters L, Richter S, Renfelde-Podehl D, Zlotowski O (2005) Centrality Indices. Network Analysis: Methodological Foundations Springer-Verlag Book Chapter:16–61. doi:10.1007/978-3-540-31955-9-3, Chapter 3
14. Borgatti S (2005) Centrality and network flow. Social Networks 27:55–71
15. Blondel V, Guillaume J, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiments:P–10008. doi:10.1088/1742-5468/2008/10/P10008
16. Geyer-Schulz A, Ovelönne M (2014) The Randomized Greedy Modularity Clustering Algorithm and the Core Groups Graph Clustering Scheme. Springer Book Chapter. ISBN: 978-3-319-01263-6, eBook ISBN:978-3-319-01264-3, doi: 10.1007/978-3-319-01264-3, http://www.springer.com/978-3-319-01263-6
17. Brandes U, Delling D, Gaertler M, Gorke R, Hoefer M, Nikoloski Z, Wagner D (30 Aug 2006) Maximizing Modularity is hard. Cornell University Library, physics data an:1–10. arXiv:physics/0608255v2
18. Brandes U, Delling D, Gaertler M, Gorke R, Hoefer M, Nikoloski Z, Wagner D (Feb 2008) On Modularity Clustering. Knowledge and Data Engineering, IEEE Transactions on 20 no.2:172–188. doi: 10.1109/TKDE.2007.190689
19. Gregori E, Lenzini L, Mainardi S (August 2013) Parallel k-Clique Community Detection on Large-Scale Networks. IEEE Transactions on Parallel and Distributed Systems 24, no.8:1651–1660. doi: 10.1109/TPDS.2012.229
20. Chen M, Kuzmin K, Szymanski BK (March 2014) Community Detection via Maximization of Modularity and Its Variants. Computational Social Systems, IEEE Transactions on 1, no.1:46–65. doi: 10.1109/TCSS.2014.2307458
21. Sun K (2005) Complex Networks Theory: A New Method of Research in Power Grid. 2005 IEEE PES Transmission and Distribution Conference and Exhibition: Asia and Pacific Dalian, China:1–6. doi:10.1109/TDC.2005.1547099
22. Bullmore E, Sporns O (March 2009) Complex brain networks: graph theoretical analysis of structural and functional systems. Nature Reviews Neurosci 10,no.3:186–198. doi:10.1038/nrn2575
23. Chopade P, Bikdash M, Kateeb I (April 2013) Interdependency modeling for survivability of Smart Grid and SCADA network under severe emergencies, vulnerability and WMD attacks. Southeastcon, 2013 Proceedings of IEEE, ISBN: 978-1-4799-0052-7:1–7. doi:10.1109/SECON.2013.6567510
24. Chopade P, Bikdash M (November 2013) Structural and functional vulnerability analysis for survivability of Smart Grid and SCADA network under severe emergencies and WMD attacks. Technologies for Homeland Security HST, 2013 IEEE International Conference, ISBN: 978-1-4799-3963-3:99–105. doi:10.1109/THS.2013.6698983
25. Milo R (2002) Network motifs: simple building blocks of complex networks. Science 298:824–827
26. Sporns O, Kötter R (2004) Motifs in brain networks. PLoS Biol 2:1910–1918
27. Amaral L, Scala A, Barthelemy M, Stanley H (10 Octomber 2000) Classes of small-world networks. Proc Natl Acad Sci USA 97 no. 21:11149–11152
28. Chopade P (2013) Robustness and survivability of smart power grid and scada networks when subjected to severe emergencies, vulnerability and WMD attacks. Doctoral Dissertation, North Carolina Agricultural and Technical State University ACM,ISBN: 978-1-303-68490-6:1–194. http://dl.acm.org/citation.cfm?id=2604359
29. Chopade P, Bikdash M (2012) Analyzing smart power grid and SCADA network robust-ness using the node degree distribution and algebraic connectivity under vulnerability and WMD attacks. Homeland Security (HST), 2012 IEEE Conference on Technologies for IEEE,ISBN: 978-1-4673-2708-4:365–372. doi:10.1109/THS.2012.6459876
30. Chopade P, Zhan J (May 2014) Community Detection in Large-Scale Big Data Networks. ASE International Conference 2014 on BIGDATA, SOCIALCOM, CYBER SECURITY, Stanford University, CA, USA ASE, ISBN: 978-1-62561-000-3:1–7. http://www.ase360.org/handle/123456789/64
31. Biyikoglu T, Leydold J, Stadler P (2007) Laplacian Eigenvectors of Graphs. Springer Publications Springer, ISBN: 978-3-540-73509-0:1–120
32. Baltz A, Kliemann L (2005) Spectral Analysis, in Network Analysis: Methodological Foundations. Springer Publications Verlag Berlin Heidelberg Springer, ISSN 0302-9743, ISBN 3-540-24979-6:373–416
33. Ernesto E, Juan R (2005) Subgraph centrality in complex networks. Physical Review E, American Physical Society 71:056103–056103. doi: 10.1103/PhysRevE.71.056103
34. Estrada E, Desmond H (2010) Network Properties Revealed through Matrix Functions. Society for Industrial and Applied Mathematics SIAM REVIEW 52 no. 4:696–714
35. Palla G, Derenyi I, Farkas I, Vicsek T (9 June 2005) Uncovering the overlapping community structure of complex networks in nature and society. Nature 435:814–818. doi:10.1038/nature03607
36. Shen H, Cheng X, Guo J (2009) Quantifying and identifying the overlapping community structure in networks. J Stat Mech 7:P07042–P07042

37. Everett M, Borgatti S (1998) Analyzing clique overlap. Connections INSNA 21 no. 1:49–61
38. Newman M (6 June 2006) Modularity and community structure in networks. PNAS 103 no. 23:8577–8582. doi:www.pnas.org/cgi/doi/10.1073/pnas.0601602103
39. Newman M (2013) Spectral methods for community detection and graph partitioning. Physical Review E 88:042822-1–042822-10. doi:10.1103/PhysRevE.88.042822
40. Shen H (2013) Detecting the Overlapping and Hierarchical Community Structure in Networks. Springer, Community Structure of Complex Networks XIV 120 e-ISBN 978-3-642-31821-4:042822-1–10. doi:10.1103/PhysRevE.88.042822
41. Newman M (2004) Fast algorithm for detecting community structure in networks. APS, Physical Review E 69 np. 6:P066133–P066133
42. MATLAB (2015) The Mathworks Inc. USA R2015a:0–1. http://www.mathworks.com/
43. iLab (2015) iLab Big Data Center. North Carolina A and T State University 1:0–1. http://www.ilabsite.org
44. NERC (2015) The North American Electric Reliability Corporation. USA 1:0–1. http://www.nerc.com/
45. WECC (2015) US Power Grid Data. USA 1:0–1. http://www.wecc.biz/