Finding Overlapping Clusters in a Highly Connected Graph from a Given Difference Density

Kittichai Lavangnananda [1] and Muhammad Aslam HemmatQachmas [2]

Data and Knowledge Engineering Lab.
School of Information Technology (SIT)
King Mongkut's University of Technology Thonburi (KMUTT)
126 Pra-cha-u-tid Road, Bangmod, Thung-Kru, Bangkok 10140, Thailand
E-mail: [1] kittichai.lav@gmail.com ; [2] muhammad.aslam@mail.kmutt.ac.th

2020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/BigData50022.2020.9378329

***Abstract*— In many fields, there may exist relationship among data or information used. Social network is a good example where relationship among members can be high and complex. Their interaction can be represented as a graph. Dividing such examples into smaller groups with similar characteristics can be translated into Graph Clustering. It is sub-field of clustering with numerous applications ranging from analysis of social network to computer network to bioinformatics. This work is an attempt to implement a novel overlapping clustering of a highly connected undirected unweighted graph. The main objective is to determine overlapped clusters in such a graph under a constrain of a specified different density. The approach starts with finding a spanning tree of the graph in order to discover all minimum cliques within the graph. These cliques are utilized by considering a clique with and an average sum of degrees of all three nodes. Such average cliques are the basis for expansion of clusters. Any clique or node that may already be a member of a cluster is allowed to be re-considered in the formation of other clusters. Hence, this enables overlapped clusters to appear. The process is carried out iteratively until all cliques and edges are considered. The two popular quality graph metrics, conductance and coverage, are selected to evaluate the overlapping clustering. Two examples of highly connected graphs are chosen for demonstration. The proposed approach is able to discover overlapping clusters from a given specific difference density. Result from this work can be used as a starting point analysis of highly connect graph, especially in the recent popularity of social network analysis.**

Keywords*— *Clique, Difference Density, Graph Clustering, Overlapping Clustering, Spanning Tree, Social Network

I. INTRODUCTION

Information in this Big Data era is highly complex. Relationship among information exists in different forms. Information and relationship among them can often be represented as graphs of different types [1]. One of such important tasks in the field of Big Data Analysis is to decompose the information into several groups with similar characteristics where they can then be further analyzed. This technique is commonly known as clustering [2]. It has become a technique within a bigger field of Data Mining and attracts much attention recently. Data or information to be clustered can be of many forms too, ranging from simple records comprising several attributes to highly inter-connected in protein-protein structure in bio-informatics.

Situations where relationship exists among the data is where where graphs can be used to represent the data and their relationship. Computing network is an obvious candidate for the application of graph representation. Social network is another popular area where it can be represented as graphs. An ability to divide a social network into smaller but similar groups lies in a special field of clustering known as Graph Clustering [3]. Social network analysis has further emphasized the advantage of graph representation, and therefore, graph clustering. A good example of graph clustering in social network is in finding smaller but similar sub-social groups (i.e. subsets of the whole social network) that have similar characteristics. A simple way of dividing a social network from demographic perspective, such as villages and districts, is far from realistic to represent communication among members in the social network, as travelling across different areas have become a norm in this era of modern communication technology.

Another good example of an application of graph clustering in a social network is the recent COVID-19 pandemic. A graph which represents such a social network is an undirected unweighted graph (i.e. two-way interaction and with no specific priority in any interaction). Edges can represent interaction between members in the network. Monitoring such network with constant and limited resources can be done efficiently if its graph representation is clustered into smaller and manageable groups. Nodes (i.e. members) which may belong to more than one groups are of particular interest in such monitoring, as they can be assigned as representatives of in the network, or on the contrary they can represent possibility of being super-spreaders during the pandemic.

The situation described above can be translated into overlapping clustering of a graph. This work is concerned with implementing algorithms which can be applied to determine overlapped clusters in a highly connected graph. Unlike many previous works, it allows users to specify density value (i.e. Difference Density), in the overlapping clustering. The work presents a novel approach to determine overlapping clusters in highly connected undirected and unweighted graphs, where cliques are used as the basis for forming and expanding of clusters

II. GRAPH CLUSTERING AND RELATED WORK

In its simplest term, clustering can be defined as grouping of data in in such a way that similarity within a group (i.e. intra-similarity) is highest and similarity among groups is lowest (i.e. inter-similarity). As stated in Section I, data to be clustered can be of various forms. This work is concerned with clustering of data in a highly connected undirected un-weighted graph. Graph clustering has numerous applications and attracts much attention recently. This Section briefly introduces the concepts and covers some significant related works.

A. Graph Clustering

A graph G = (V, E) consists of two sets of elements V and E, where V refers to a set of vertices (nodes) and E refers to a set of edges [1]. A simple classification of graphs can be done by noting the type of their edges (i.e. with respect to direction and weight). Similar to clustering, graph clustering shares the same objectives of maximizing intra-similarity of a cluster and minimizing inter-similarity among clusters. Unlike data clustering, graph clustering can be categorized in to three types as follows :

a) Partitioning Clustering: Partitioning clustering is a method of clustering which divides a graph into small subgraphs based on some similarities. In this type of clustering, a node in a graph can be a member of only one cluster. A graph depicted in Fig. 1 is used to demonstate these there types of grapg clustering with an intention to show that all possible types of cluustering are posiible in an arbitrary graph. Fig. 1, depicts an example of a possible partitioning clustering of the graph into two partitioning clusters, noting that each node is a member of a particular partition only.

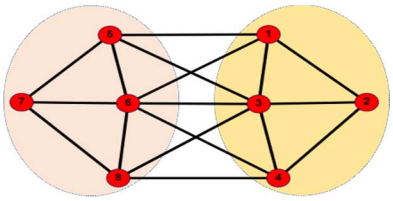

Fig. 1. An example of a possible partitioning clustering

b) Overlapping Clustering: Overlapping clustering is also a method of clustering a graph similarly to partitioning clustering. However, this type of clustering allows any node to be a member of more than one clusters. Fig. 2 depicts a possibility of overlapping clustering of the same graph in Fig. 1. Note that nodes '3' and '6' belong to three different clusters, hence overlapping of clusters occurs.

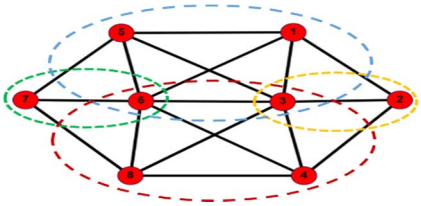

Fig. 2. An example of possible overlapping clustering of the same graph shown in Fig. 1

c) Hierarchical Clustering: Hierarchical Graph Clustering is a special kind of graph clustering. In this type of clustering, the nodes of a graph are grouped into similar clusters where each cluster can contain subcluster(s) inside itself, and recursively a subcluster may be divided further to form hierarchy of clusters within the original graph. Hierarchical clustering can be performed in two ways, Agglomerative (i.e. bottom-up) and Divisive (top-down) approaches. Fig. 3 depicts a possible of hierarchical clustering of the same graph in Fig. 1. Note that there are two layers of clusters appearing from the overall graph.

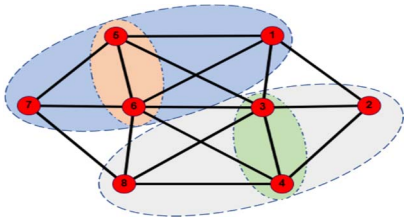

Fig. 3. An example of a possible hierarchical clustering of the same graph shown in Fig. 1

It is worth bearing in mind not only different types of clustering can be performed in a graph but also the result of each type of clustering is not unique. Different results are possible depending on the criteria used in clustering. Graph clustering is known to be an *NP* class problem in general. Also, graph clustering is often referred to as community detection in social network analysis context [4], [5].

B. Related Work

Considerable amount of researches have been done in graph clustering to several types of graphs ranging from un-directed unweighted to directed weighted graphs. While it may not be the first, such research can be found as early as in 1972 [6], where typical depth-first search is applied to is applied to is applied to biconnected component of graphs. A good literature surveys in graph clustering in general can be founded in [7]. Partitioning clustering of graphs has been carried out on over 200 million web pages are clustered to discover local and global property [8]. The recent work in this type of clustering is in [9] where a spanning tree and cliques are applied. Research on hierarchical clustering has attracted some interest too. An agglomerative modularity-based clustering techniques had been extended to hierarchical clustering in [10]. An algorithm based on shared neighbours and links between clusters had also been applied to deter-mine hierarchical clusters in graphs [11]. A good source of hierarchical clustering can be found in [12].

As this work is concerned with overlapping clustering, therefore,more attention is spent on this in this Section. Research in this area had given higher priority to edges rather than vertices for finding overlapping clusters. The work in [13] suggested that detecting overlapping clusters may occur more times when edges were considered. An algorithm known as BIGCLAM [14] was proposed for speedy and scalable model-based community detection to determine overlapping in big and dense networks of millions of nodes and edges. Density-based clustering had been applied to arbitrary shape of graphs where user specified criterion was required [15]. Two approaches known as RaRe (starts the clustering process from high ranking nodes) and IS (uses candidate clusters as the starting point) were developed in [16] in attempt to find an optimal solution in overlapping clustering on synthetic and real world graphs. Relationship between nodes were used to discover overlapped clusters in a social network [17], where information among users (user data) and tags (connections between users) were utilized. A soft clustering approach based on a genetic algorithm [18] has an application in this type of clustering too. The work determined communities by a certain size and made clusters closer by reducing the number of overlapping clusters. The concepts of betweenness centrality had also been adopted in extended Girvan and Newman's method [19] for both hierarchical and partitioning clustering and letting them to

overlap. The work in [20] was a good illustration of application of clique-based clustering (a popular method of clustering), in overlapping clustering.

The description above has shown that various approaches to all three types of graph clustering exist. Nevertheless, plenty of improvement in this field is still possible, especially where different required criteria of clustering is of interest. The work in the article is based on application of spanning tree and minimum clique. It is intended to determine overlapped clusters with specific difference density (specified by users). To date, this work is the first attempt in overlapping clustering which a kind of density is used as the objective of the process.

III. Implementation of Overlapping Clustering in a Highly Connected Graph

Before As stated in the Section I, the application of overlapping clustering in the work is intended for highly connected undirected unweighted graph. Clustering of any type in a sparse graph is not a challenging task, as it is almost possible to consider every possibilities as the ratio between edges and nodes is relatively low. A graph is considered highly connected, if $\lambda(G) >= n/2$, where $\lambda(G)$ denotes the number of edges in the graph and n is the number of nodes [20]. Even with this condition, a cluster can be very visible, especially where cutting a graph into a set of subgraphs can be done when degrees (i.e. number of edges) between a pair (or only few pairs) of nodes is very low while degrees of other nodes are high. In such situations, formation of clusters can happen naturally.

Graph Clustering becomes necessary when formation of clusters are not apparent. The process can be classified mainly into two types. The first is to perform minimum cutting at appropriate places [3], [21]. The second is to utilize shapes, which are commonly known as motifs and cliques [3], [22]. Technically, a clique is a fully connected graph, however these two terms are sometimes used interchangeably in graph clustering. Fig. 4 depicts examples of common motifs used in graph clustering.

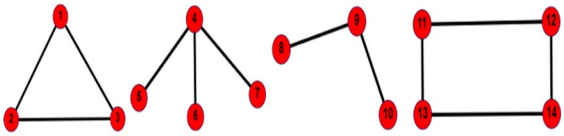

Fig. 4. Example of motifs used in graph clustering

This work adopts the use of minimum cliques (i.e. triangles) together with a spanning tree in a highly connected graph as a basis for overlapping clustering. Using minimum cliques (i.e. a triangle) ensure that all edges are considered. For example, a clique with 4 nodes comprises 4 minimum cliques, so some edges may be rejected during expansion of clusters if a clique of 4 nodes is used. Hence forth, in this article, sum of degrees in all three nodes of a minimum clique is referred to as 'the degrees of clique' for brevity.

Graph clustering can be seen into two perspectives. The first is to determine clusters according to a desired number of clusters (k-mean clustering is a good example of this, where the value if k is pre-defined). Another is to determine appropriate number of clusters from given specific constrain(s). The work adopts the latter approach, as stated earlier, 'Difference Density' is required in overlapping clustering intentionally. This concept mimics the data clustering where Inter-Intra value is commonly used. Nevertheless, this term 'Difference Density' is not yet a commonly used in Graph Clustering.

In practice, this value is given by users since the objective of this work is to discover communities (i.e. clusters) with a specific density in a complex network. Difference Density can be determined by equations 1 - 3 as bellow:

$$\delta\text{int.}(C) = (\text{No. of internal edges of } C)/(n_c (n_c - 1)/2) \quad (1)$$

$$\delta\text{ext.}(C) = (\text{No. of external edges of } C)/(n_c (n - n_c - 1)) \quad (2)$$

$$\text{Difference Density of the cluster } C = \delta\text{int}(C). - \delta\text{ext}(C) \quad (3)$$

Where : n is the total no. of nodes in the graph

: n_c the total no. of nodes in the cluster C

This is worth bearing in mind that this index is independent of cluster size. It uses number of internal and external edges as a guide to indicate the density of a cluster within the overall graph. Another possible index is the Clustering Coefficient [23]. This index indicates the density of the overall graph before clustering. Nevertheless, it may be adapted to measure each cluster a formation occurs during the clustering too.

A. Determination of all Cliques

In determining overlapping clusters, all cliques in the graph must be discovered first. The technique of discovering a spanning tree in a graph has an application here, since it is the starting point in discovering all cliques within the graph. The pseudo code in discovering all cliques in a given highly connected graph is given in Fig. 5.

```
Finding a Spanning Tree from the graph;
Number of Fundamental Cycles (Fn) = 0;
Number of Cliques (Cn) = 0;
Repeat
        Add an edge that is in the graph but not in the Spanning Tree
        If cycle occurs (i.e., Fundamental Cycle is found)
        Then [record this as a Fundamental Cycle;
                Increment the value of (Fn) by 1;
                If this cycle is a Clique
                Then [record this cycle as a Motif;
                        Determine the total degrees of this Clique;
                        Increment the value of (Cn) by 1;
                        ]
                ]
Until (no more edges to consider);
Number of Combinations (C) = 2;
Repeat
        Perform "Exclusive OR" to all Fundamental Cycles of combinations C;
        IF ( The "Exclusive OR" operation results in a Clique)
        Then [ Record this Clique;
                Determine total degrees of this Clique;
                Increment the value of (Cn) by 1;
                ]
Increment the value of C by 1;
Until (C=Fn)
```

Fig. 5. The Pseudo Code to determine all cliques in a highly connected graph

B. Determination of Overlapping Clusters

Once all cliques and their degrees within the graph are discovered, the overlapping clustering can commence. While nothing is conclusive, previous work in [9] had suggested that starting the process and continue considering cliques with highest degrees are likely to result in descending order of clusters size. The objective of this work is to discover clusters of similar sizes as they are more likely to share similar commonality and also any action to each cluster is likely to require similar amount of resources. Therefore, this work selects cliques with average degrees as a stating point also considers cliques with average degrees among the remaining unconsidered cliques during the overlapping clustering process.

Overlapping clustering commences with the clique with an average degrees as the initial cluster. The cluster expands by adding adjacent cliques and edges to the cluster with the specified different density in mind. Once expansion is no longer possible, then a cluster is found. The same process resumes with remaining cliques and edges. Any clique and edge that have been considered but adjacent to the expansion of the current cluster may be reconsidered. The fact that cliques and edges already belong to any cluster are allowed to be reconsidered for formation of a new cluster is the key to discover overlapped nodes, and hence overlapping clustering in this work. Fig. 6 summarizes the pseudo code in overlapping clustering in this work.

```
Do
    Find the unconsidered cliques with the average total degrees
      As the initial cluster (Cl) (* i.e. starting a new cluster *)
    Repeat
      Find adjacent clique (Cq) with the average total degrees
      For each edge of Cq
       Add the edge that is connected to Cl
       Calculate the 'Difference Density' of the new Cl
      If 'Difference Density' of the new Cl
        is [(higher) OR (equal to specified 'Difference Density')]
      Then A new edge is added to Cl (* i.e. the cluster is expanded* )
      Else A new edge is rejected (*Cl remained unchanged *)
     Until ( No more adjacent clique to consider )
    Repeat
        Find adjacent edge(s) to Cl
        For each edge
          Repeat the same process as for a clique.
    Until ( No more adjacent edge to consider )
  (* a cluster with 'Difference Density' higher
      or the same as specified value is found *)
While ( There is/are unconsidered clique(s) to be considered )
```

Fig. 6. The Pseudo code for the overlapping clustering

C. Clustering Quality Metrics

Evaluation of a result from a clustering method in essential as it assesses the quality of clusters. Following the principle of clustering, a cluster in a graph is of good quality if internal nodes are densely connected among themselves and external nodes are sparse to the remaining of the graph. Several metrics exist in evaluation of clustering [24]. This work adopts the two well known metrics in graph clustering, these are conductance and coverage. Their brief description are given below :

a) Conductance: Conductance in a graph measures how densely the nodes are connected in a graph are and defines the nodes converging to a specific cluster. The higher value of conductance indicates better clustering. This metric can be determined by the Equation 4 below :

$$\text{Conductance} = 1 - \frac{1}{K}\sum_K \phi(C_K) \qquad (4)$$

Where: K = the number of clusters

$\phi(C_K)$ = (No. of external edges in cluster C) / (No. of internal edges in cluster C)

b) Coverage: Coverage value ranges from 0 to 1. Higher values of coverage mean that there are more edges inside the clusters than edges linking different clusters, which in turn, translates to a better clustering. This metric can be determined by the Equation 5 below :

$$\text{Coverage} = \frac{\text{Internal edges of all clusters in a graph}}{\text{Number of edges in a graph}} \qquad (5)$$

IV. Experimentation and Results

In practice, social network data may be company properties. So in order to do extensive study in this work, graphs are randomly generated and visualized by means of Python Language together with the Networkx Functions. Numerous experiments have been carried out on graphs with different sizes and difference density values ranging from 20 nodes with 50 edges to 500 nodes with 1,000 edges. Due to limited space, it is infeasible to show them all in this Section. Also their details can be difficult to comprehend and distract crucial findings. For illustration purposes, two examples of highly connected graphs are selected and for discussion. They may represent sufficiently complex social networks. The first is a highly connected graph comprises 50 nodes and 300 edges and the second example comprises 50 nodes and 500 edges. The different density value selected for illustration is 0.4 for both examples for ease of visualization.

A. Result of Example 1

Fig. 7 depicts an arbitrary highly connected graph of 50 nodes and 300 edges in Example 1.

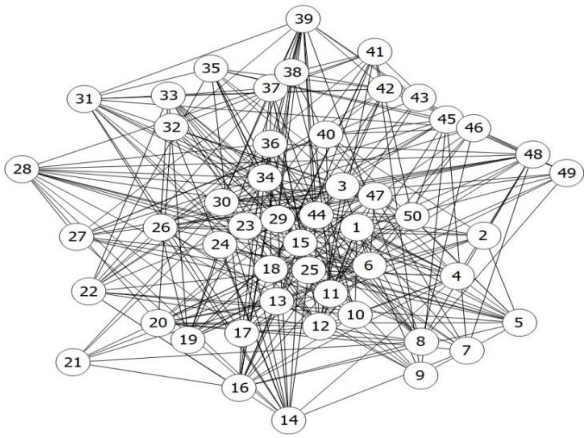

Fig. 7. An arbitrary highly connected graph of 50 nodes and 300 edges (Example 1)

For the different density of 0.4, the overlapping clustering approach described in Section 3 yields the result in Table 1.

TABLE I. RESULT OF OVERLAPPING CLUSTERING IN EXAMPLE 1

No. of edges	No. of nodes	No. of Cliques	Specified Minimum Difference Density (DD)
50	300	284	0.4

Cluster	No. of Nodes	Nodes	DD
1	13	2 3 6 9 14 17 26 28 32 41 44 45 48	0.42
2	13	5 8 9 17 20 21 26 28 35 36 45 46 48	0.47
3	11	4 7 10 12 15 18 28 31 33 35 47	0.43
4	12	8 11 16 20 28 30 31 32 39 45 46 49	0.41
5	12	2 6 7 8 16 17 27 29 39 40 43 48	0.43
6	11	5 9 12 22 26 28 33 41 46 48 50	0.42
Unclustered	9	1 13 19 23 24 25 34 37 38 42	N/A
Overlapped Nodes	21		N/A

Clustering Quality Metrics	
Conductance	Coverage
0.595	0.52

Displaying the results in Table 1 comprehensively becomes a difficult task. Good data visualization may be necessary if indepth analysis is required. This is because a node or an edge can be unclustered or a member single clusters or more than one clusters. Drawing a contour to each cluster is not possible either as nodes within a cluster may spread apart from each other within the graph.

Figure 8 (a) depicts the result of three overlapped clusters (clusters 2, 4 and 5). A colour is used for a cluster. However, some edges may be members of two or more clusters. In such case, a colour of a possible cluster it belongs to is chosen to keep the visualization simple. In this example, the node '8' overlaps with all three clusters (note that edges adjacent to this node have three colours, joining three clusters). Similarly, Figure 8 (b) depicts the result of five overlapped clusters (clusters 1, 2, 3, 4 and 6). In this example, the node '28' overlaps with all five clusters (note that edges adjacent to this node have five colours, joining five clusters).

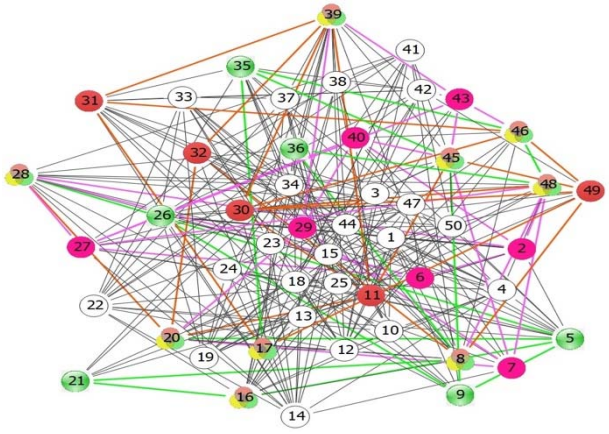

Fig. 8. (a) Three overlapping clusters (Ex. 1)

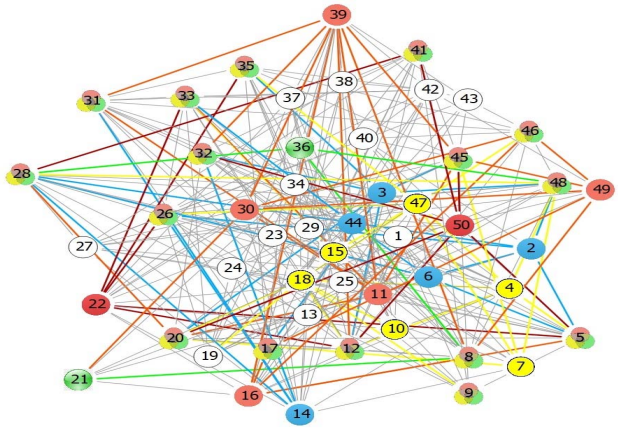

Fig. 8. (b) Five overlapping clusters (Ex. 1)

B. *Result of Example 2*

Fig. 9 depicts an arbitrary highly connected graph of 50 nodes and 500 edges in Example 2.

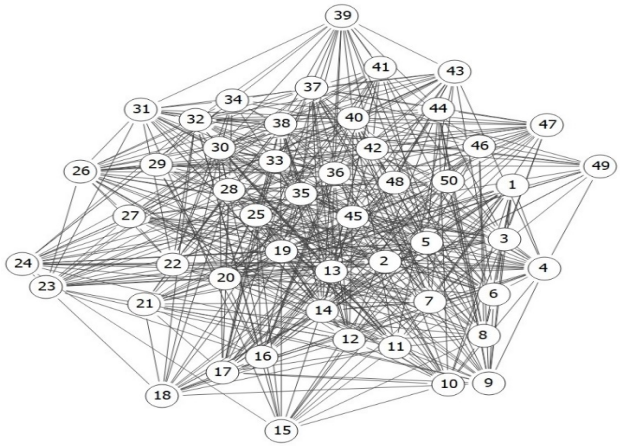

Fig. 9. An arbitrary highly connected graph of 50 nodes and 500 edges (Example 2)

For the different density of 0.4, the overlapping clustering approach described in Section III yields the result in Table 2.

TABLE II. RESULT OF OVERLAPPING CLUSTERING IN EXAMPLE 2

No. of edges	No. of nodes	No. of Cliques	Specified Minimum Difference Density (DD)
50	500	1322	0.4

Cluster	No. of Nodes	Nodes	DD
1	9	5 6 14 17 24 27 31 39 41	0.41
2	7	8 25 26 34 37 47 50	0.43
3	8	6 10 13 24 28 33 36 43	0.42
4	7	16 23 26 35 37 44 47	0.44
5	7	3 8 20 23 26 39 42	0.41
6	6	4 18 24 32 43 49	0.42
7	7	2 4 9 22 23 27 37	0.41
8	7	6 12 23 26 31 44 46	0.43
8	8	1 11 13 15 30 31 39 43	0.42
10	7	6 16 18 29 32 48 49	0.41
11	6	1 7 24 26 40 43	0.44
Unclustered	4	19 21 38 45	Na
Overlapped Nodes	20		Na

Clustering Quality Metrics	
Conductance	Coverage
0.67	0.418

Figure 10 (a) depicts the result of three overlapped clusters (clusters 1, 8 and 10). In this example, the node '6' overlaps with all three clusters (note that edges adjacent to this node have three colours, joining three clusters). Similarly, Figure 10 (b) depicts the result of five overlapped clusters (clusters 2, 4, 5, 8 and 11). In this example, the node

'26' overlaps with all five clusters (note that edges adjacent to this node have five colours, joining five clusters).

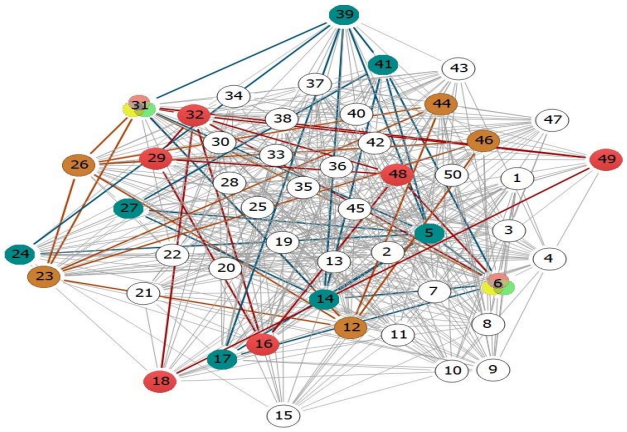

Fig. 10. (a) Three overlapping clusters (Ex. 2)

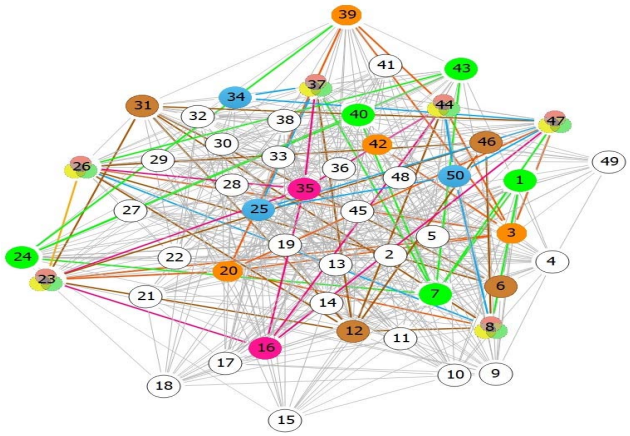

Fig. 10. (b) Five overlapping clusters (Ex. 2)

V. Discussion and Conclusion

The results from previous Sections reveal some interesting findings. They can be summarized into two main facets as follows :

A. *Difference Density*

From our experimentation, the difference density is a major constrains the formation of the overlapping clusters. The higher value of this value leads to more but smaller clusters and more overlapped nodes may occur. Also more number of unclustered nodes is likely to occur is a spare graph, especially if the required difference density is relatively high. Nevertheless, the overall shape of the graph is likely to dictate the formation of the overlapping clusters.

B. *Density of the graph and the spread of degrees among nodes in the graph*

A highly dense graph is likely to implicitly contain more overlapped clusters. Nevertheless, if more nodes with lower degrees exist in such graph, this has an affect of reducing possible overlapped clusters. In situations where few nodes have much lower degrees than the rest, overlapping clustering may be visible from the start, in such cases the use of different density may not be appropriate and suitable results may consists of different cluster sizes and quality.

C. *Quality Metrics Used*

This work adopts the two metrics, Conductance and Coverage. Both take internal and external edges into consideration. Other metrics exist, many measure a quality of a particular cluster instead of the clustering process. The coverage metric has an advantage of easy to understand as it ranges from 0 to 1. Conductance can measure both each cluster and the clustering process.

VI. Conclusion and Future Work

This work is the first attempt in overlapping clusters under a constrain of specified difference density. To date, apart from the work in [9], no similar attempt was reported in literature, therefore comparison with existing techniques will not be meaningful. This work adopts similar approach to [9] but is more complex in nature as clustered nodes and cliques need not be reconsidered once they belong to a partitioning cluster.

The approach in this work allows a user specified constrain in terms of difference density of the cluster (i.e. $\delta int(C). - \delta ext(C)$ value). It also allows further analysis of a cluster or deploy of resource available for each cluster where special attention can be paid to overlapped nodes.

Future work can be explored from different aspects. An obvious candidate is the approach in hierarchical clustering. This suggest identifying the suitable approach (i.e. whether agglomerative or divisive) to the task. Also investigation of this approach can be carried out on publicly available domain graphs [25].

The difference density value adopted in this work is a possible specified constrain. It merits further investigation for its suitability in general. Its description may not be easy to comprehend to those who are not familiar with graph clustering. From this perspective, coverage is a good metric and may be used as a constrain since it is very meaningful (i.e. ratio between all edges in all clusters and total number of edges in the graph).

As stated in Section II, many possible formations of overlapping clustering are possible, therefore, like many graph clustering research an ultimate objective is to have an algorithms which can optimize the overall overlapping clustering process.

Acknowledgment

The authors wish to express their gratitude to The Higher Education Development Program (HEDP), Ministry of Higher Education of Afghanistan and King Mongkut's University of Technology Thonburi, Thailand for the Scholarship of Mr. M. Aslam. Provision of computing facilities at School of Information Technology, KMUTT is gratefully acknowledged. The moral support from Dr. P. Lavangnanada through out this study is much appreciated.

References

[1] Gross, Jonathan L., and Jay Yellen. "Handbook of Graph Theory CRC Press." Boca Raton (2004).

[2] Everitt, B. S., et al. "Cluster Analysis, 5th Editio John Wiley & Sons." West Sussex, UK [Google Scholar] (2011).

[3] Schaeffer, Satu Elisa. "Graph clustering." Computer science review 1.1 (2007): 27-64.

[4] Fortunato, Santo. "Community detection in graphs." Physics reports 486.3-5 (2010): 75-174.

[5] Evans, Tim S. "Clique graphs and overlapping communities." Journal of Statistical Mechanics: Theory and Experiment 2010.12 (2010): P12037.

[6] Tarjan, Robert. "Depth-first search and linear graph algorithms." SIAM journal on computing 1.2 (1972): 146-160.

[7] Xu, Rui, and Donald Wunsch. "Survey of clustering algorithms." IEEE Transactions on neural networks 16.3 (2005): 645-678.

[8] Broder, Andrei, et al. "Graph structure in the web." Computer networks 33.1-6 (2000): 309-320.

[9] Lavangnananda, Kittichai, and Chidchanok Panyarit. "Determination of Different Sizes of Partitioning Clusters in a Highly Connected Graph." 2019 11th International Conference on Knowledge and Smart Technology (KST). IEEE, 2019.

[10] Hierarchical Graph Clustering using Node Pair Sampling, https://arxiv.org/abs/1806.01664, last accessed 2020/08/31

[11] Huijuan, Zhang, Sun Shixuan, and Cai Yichen. "An Efficient Hierarchical Graph Clustering Algorithm Based on Shared Neighbors and Links." International Conference on Knowledge Science, Engineering and Management. Springer, Berlin, Heidelberg, 2013.

[12] Chowdhary, Anupama. "Community Detection: Hierarchical clustering Algorithms." Int. J. Creat. Res. Thoughts 5.4 (2017): 2320-2882.

[13] Fellows, Michael R., et al. "Graph-based data clustering with overlaps." International Computing and Combinatorics Conference. Springer, Berlin, Heidelberg, 2009.

[14] Yang, Jaewon, and Jure Leskovec. "Overlapping community detection at scale: a nonnegative matrix factorization approach." Proceedings of the sixth ACM international conference on Web search and data mining. 2013.

[15] Ester, Martin, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." Kdd. Vol. 96. No. 34. 1996.

[16] Baumes, Jeffrey, et al. "Finding communities by clustering a graph into overlapping subgraphs." IADIS AC 5 (2005): 97-104.

[17] Wang, Xufei, et al. "Discovering overlapping groups in social media." 2010 IEEE international conference on data mining. IEEE, 2010.

[18] Bello-Orgaz, Gema, Héctor D. Menéndez, and David Camacho. "Adaptive k-means algorithm for overlapped graph clustering." International journal of neural systems 22.05 (2012): 1250018.

[19] Gregory, Steve. "An algorithm to find overlapping community structure in networks." European Conference on Principles of Data Mining and Knowledge Discovery. Springer, Berlin, Heidelberg, 2007.

[20] Hüffner, Falk, Christian Komusiewicz, and Manuel Sorge. "Finding highly connected subgraphs." International Conference on Current Trends in Theory and Practice of Informatics. Springer, Berlin, Heidelberg, 2015.

[21] Chung, Fan RK, and Fan Chung Graham. Spectral graph theory. No. 92. American Mathematical Soc., 1997.

[22] Zhao, Peixiang. "gSparsify: Graph motif based sparsification for graph clustering." Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. 2015.

[23] Schank, Thomas, and Dorothea Wagner. "Approximating clustering coefficient and transitivity." Journal of Graph Algorithms and Applications 9.2 (2005): 265-275.

[24] Biswas, Anupam, and Bhaskar Biswas. "Defining quality metrics for graph clustering evaluation." Expert Systems with Applications 71 (2017): 1-17.

[25] Stanford Large Network Dataset Collection (SNAP) [Online]. Available : https://snap.stanford.edu/data/ [Accessed 8th-Nov.-2020].