COMPARATIVE PERFORMANCE ANALYSIS OF RNSC AND MCL ALGORITHMS ON POWER-LAW DISTRIBUTION

Mousumi Dhara[1] and K. K. Shukla[2]

[1]Department of Computer Engineering, Indian Institute of Technology, Banaras Hindu University, Varanasi City
mousumi.dhara@gmail.com

[2]Department of Computer Engineering, Indian Institute of Technology, Banaras Hindu University, Varanasi City
kkshukla.cse@itbhu.ac.in

ABSTRACT

Cluster analysis of graph related problems is an important issue now-a-day. Different types of graph clustering techniques are appeared in the field but most of them are vulnerable in terms of effectiveness and fragmentation of output in case of real-world applications in diverse systems. In this paper, we will provide a comparative behavioural analysis of RNSC (Restricted Neighbourhood Search Clustering) and MCL (Markov Clustering) algorithms on Power-Law Distribution graphs. RNSC is a graph clustering technique using stochastic local search. RNSC algorithm tries to achieve optimal cost clustering by assigning some cost functions to the set of clusterings of a graph. This algorithm was implemented by A. D. King only for undirected and unweighted random graphs. Another popular graph clustering algorithm MCL is based on stochastic flow simulation model for weighted graphs. There are plentiful applications of power-law or scale-free graphs in nature and society. Scale-free topology is stochastic i.e. nodes are connected in a random manner. Complex network topologies like World Wide Web, the web of human sexual contacts, or the chemical network of a cell etc., are basically following power-law distribution to represent different real-life systems. This paper uses real large-scale power-law distribution graphs to conduct the performance analysis of RNSC behaviour compared with Markov clustering (MCL) algorithm. Extensive experimental results on several synthetic and real power-law distribution datasets reveal the effectiveness of our approach to comparative performance measure of these algorithms on the basis of cost of clustering, cluster size, modularity index of clustering results and normalized mutual information (NMI).

KEYWORDS

RNSC, MCL, Cost of Clustering, Cluster size, Modularity Index of Clustering Results, NMI

1. INTRODUCTION

The main intention of clustering is to achieve some meaningful information by partition a dataset into clusters in terms of its intrinsic structure, without resorting to any a priori knowledge such as the number of clusters, the distribution of the data elements, etc. This clustering process is basically used to sort out some problems incurred into complex systems in nature and society. The important thing of clustering is that it can relate with many applications, and a number of different algorithms and techniques have emerged over the years. Clustering graph in complex network systems is an essential problem with many applications in a number of disciplines. Graph clustering algorithms emphasis on clustering the nodes of a graph [1], [2]. It can expect from a graph clustering scenario that it contains a collection of sub graphs (nearly completely connected) and a small fraction of edges are existed between them for inter

DOI : 10.5121/acij.2012.3503

connection. For a weighted graph, the edge weight should be considered here for creating sub graphs and small weight edges are taking part between them [3].

Graph clustering is a powerful tool and has been studied and applied in many research areas, which include image segmentation [4,5], machine learning, data mining [6], bioinformatics [7,8], etc. Spectral methods have been achieved effectiveness in solving a number of graph clustering objectives, including ratio cut [9] and normalized cut [10] and has been convenient in many areas such as circuit layout [9] and image segmentation [10]. Recently, spectral clustering is getting immense popularity because of the convention of eigenvectors applied in various machine learning tasks [11]. In the recent past, various other graph clustering algorithms came into the field like restricted neighbourhood search clustering (RNSC) [12], Markov clustering (MCL) [13], super paramagnetic clustering (SPC), Genetic Algorithm, Molecular Complex Detection (MCODE), Local Clique Merging Algorithm (LCMA), etc.

RNSC, which is a cost based clustering method and performs local search iteratively to obtain optimum clustering in an efficient way. RNSC is a stochastic technique which uses restricted neighbourhood search concept. It also acts like a metaheuristic technique like tabu search, described in [14] and also can be used in various search space schematics. It is also known as Variable neighbourhood search [15]. A restriction is imposed in the neighbourhood for the current clustering while doing iterative local search. The main goal of this algorithm is to find the best cost clusterings (lower cost) from the set of clusterings of a graph by assigning some cost functions (Naive cost function and scaled cost function). The memory requirement for RNSC is O (n^2). The complexity of a move in the naive cost function is O (n), which is the size of the restricted neighbourhood of a move M.

MCL is an efficient clustering method in weighted graphs, based on the prototype of stochastic flow simulation technique. In this technique, clusters (a natural grouping of densely flow-connected vertices) are obtained by using two operators: flow expansion and inflation. MCL technique performs well for sparse graphs

Recently, complex graphs or complex networks are most popular in nature and society. It can clarify various complex systems such as the cell, a network of substrates connected by chemical reactions [16], the society, a network of individuals linked by various social links [17], the Internet, a network of routers connected by various physical connections [18], the World Wide Web, etc. The probability P (k) that a node in the network is connected to k other nodes (k=0, 1, 2,.., n) is called the degree distribution or connectivity distribution. This is a very important characteristic of a network. Power-law or scale-free graph was first introduced by Barabasi and Albert (1999) [19]. Complex network topology like WWW, the actor collaboration network and the citation network, etc., are Scale-free network, which is described by them. Scale-free network usually follows the power law degree distribution; where γ is the degree exponent.

$$\mathrm{P}[\kappa] \sim k^{-\gamma} \tag{1}$$

In this work, the performance of RNSC and MCL is tested on both real and synthetic benchmark undirected large-scale scale-free graph. Widespread experimental results on several real and synthetic datasets demonstrate the behaviour of both the algorithms. The comparative assessment of both the algorithms is measured in terms of cost of clustering, cluster size, modularity index of clustering results and NMI value.

2. GRAPH CLUSTERING ALGORITHMS AND POWER-LAW GRAPH

Here we discuss about the graph clustering algorithms mentioned above and the random power-law graph datasets, used in the performance analysis of these algorithms.

2.1. RNSC (Restricted Neighbourhood Search Clustering*)*

RNSC is a local search meta-heuristic technique which is used to minimize the cost of clustering in the solution space. According to Stijn van Dongen, the vertex-wise performance criteria for clustering of unweighted graphs as the sum of the coverage measure taken on each vertex. In RNSC, a simple integer-valued cost function (called the naive cost function) is used as a pre-processor to produce initial clustering results on a graph and after that to evaluate the low-cost clustering result, a more expressive (but less efficient) real-valued cost function (called the scaled cost function) is applied. The scaled function tries to optimize the output from naive function and reach to the global optimal solution.

For a clustering C on a graph G (V, E) in which |V| = n, the coverage measure for Naïve cost function is expressed as in eq. (2).

$$Cov(G,C,v)=1-\frac{\neq^{1}_{out}(G,C,v)+\neq^{0}_{in}(G,C,v)}{n-1} \qquad (2)$$

Where $\neq^{1}_{out}(G,C,v)$ and $\neq^{0}_{in}(G,C,v)$ are denoted respectively as a number of cross edges incident to *v* and number of vertices in C_v that are not adjacent to *v* and for good clustering, these noted parameters should be small. Naive cost function is expressed as follows.

$$C_n(G,C)=\frac{1}{2}\sum_{v\in V}(\neq^{1}_{out}(G,C,v)+\neq^{0}_{in}(G,C,v)) \quad . \qquad (3)$$

The more expressive scaled coverage measure is in the following expression where, *N (v)* is the open neighbourhood of *v*.

$$Cov(G,C,v)=1-\frac{\neq^{1}_{out}(G,C,v)+\neq^{0}_{in}(G,C,v)}{N(v)C_v} \qquad (4)$$

The scaled cost function is expressed as in eq. (5).

$$C_s(G,C)=\frac{n-1}{3}\sum_{v\in V}\frac{1}{|N(v)\cup C_v|}(\neq^{1}_{out}(G,C,v)+\neq^{0}_{in}(G,C,v)) \qquad (5)$$

The pseudo code of the Naive cost scheme and Scaled cost Scheme are presented here.

//Naive Cost scheme:-

```
Begin
      Exper =1;
      $N_E$=Total no of Experiments;
            If (Exper$\geq N_E$)
                  {
      Then
      Obtain final clustering;
                  }
      Else
                  {
      Initial clustering = $C_0$; //Cluster in C with label 0.
      Bestcost =  ∞ (user input);
      $T_n$= Naive stopping Tolerance;
            If (Bestcost has improved in the last $T_n$ moves)
                  {
            Then
            Make a non-tabu near-optimal move;
            Initial clustering = New clustering;
            If (New clustering is the best clustering so far)
            Then
            Store it as best clustering $C_n$;
      Bestcost ++;
            Update tabulist and other datastructures;
            Else
            Update tabulist and other datastructures;
                  }
      Else
      Run scaled cost scheme;
                  }
      Exper ++;
```

// Scaled Cost Scheme:-

```
Exper =1;
$N_E$= Total no of Experiments;
If (Exper$\geq N_E$)
            {
Then
      Obtain final clustering;
            }
Else
            {
Input the Naive cost clustering $C_n$;
NumMoves = 0;
DivCount = 0;
Bestcost =$\infty$;
$T_S$ = Scaled stopping Tolerance;
$F_D^{'}$=Destructive    diversification frequency;
If (Bestcost has improved in the last $T_S$ moves)
Then
      Output final Clustering
Else
            {
If (DivCount$\geq F_D^{'}$)
            {
Then
      Destroy a random cluster;
DivCount = 0;
Obtain New clustering;
If (The New clustering is the best clustering so far)
            {
      Then
      Store it as best clustering $C_S$;
      Bestcost ++;
      Update tabulist and other data structures;
            }
      Else
      Update tabulist and other data structures;
            }
Else
Make a non-tabu near-optimal move.
NumMoves ++;
DivCount ++;
Obtain New Clustering;
      If (The New clustering is the best clustering so far)
      Then
      Store it as best clustering $C_S$;
      Bestcost ++;
      Update tabulist and other data structures;
      Else
      Update tabulist and other data structures;
            }
            }
Exper ++;
```

2.2. MCL (Markov clustering)

The Markov clustering, proposed by Stijn van Dongen, this delivers a very fast clustering method and also provides a natural clustering in weighted graphs [20]. This algorithm is based on the prototype of stochastic flow simulation technique. Two operators, flow expansion and inflation are used to create a natural grouping of densely flow-connected vertices, clusters. These two operators are constructed from the input graph and they are used to transform the probability of the random walk in the Markov chain like way to another. Actually, the inflation is used for strengthening the flow where it is strong and also weakening the flow where it is already weak and the flow expansion is used for propagating the flow within the graph. MCL is fast for sparse graphs. MCL Algorithm is explained step by step below.

Step1: Input weighted directed or undirected graph;
Step2: Create the adjacency matrix of the graph;
Step3: Add self-loop to each vertex;
Step4: Normalize the matrix $R^{k\times l}$;
Step5: Expand the matrix with e^th^ power i.e. $(R_{kl})^e$
Step6: Inflate the matrix by taking inflation of the resulting matrix with parameter r;
Step7: Repeat step 5 and 6 until a steady state is achieved;
Step8: Interpret resulting matrix to discover clusters.
The inflation operator is denoted as Γ_r with power coefficient r, a real nonnegative number. The matrix is denoted as M∈ $R^{k\times l}$, M≥0.The matrix resulting from rescaling each of the columns of M with power coefficient r is denoted as $\Gamma_r M$ i.e.

$$(\Gamma_r M)_{pq} = \frac{(M_{pq})^r}{\sum_{i=1}^{k}(M_{iq})^r} \tag{6}$$

2.3. Power-law Graph

It follows power law distribution, as shown it before in eq. (1). The exponent γ is scattered between 2.1 and 3. The power-law tailed degree distribution is remarkably different from the Poisson distribution. Scale-free networks are inhomogeneous, leading over time to some vertices that are highly connected, a “rich-get-richer” phenomenon that can be easily detected in real networks as shown in figure 8 and figure 11.

3. Few Parametric Concepts

The evaluation of clustering results, produced by algorithms is performed on the basis of some metrics. Some of the metrics are defined here such as modularity index, NMI value and cluster size. Graph size is a basis, depend on which all the computation are performed to obtain the characteristics of both the algorithm.

3.1 Modularity Index

A topology-based modularity metric, originally proposed by Newman and Girvan [21], is used in this investigation to check the performance. This is a square symmetric matrix of clusters where each element d_{ij} represents the fraction of edges that link nodes between clusters i and j and each d_{ii} represents the fraction of edges linking nodes within cluster i. The modularity measure is given by eq. (1).

$$M = \sum_i (d_{ii} - (\sum_j d_{ij})^2) \tag{7}$$

3.2 Normalized Mutual Information (NMI)

It is the measure of the quality of clusters, which is the mutual information shared between clusterings. This is originally proposed by Alexander Strehl and Joydeep Ghosh [22]. Assume, there are set of groupings of clusterings as $\{\lambda^{(q)} | q \in \{1,..,r\}\}$ which is denoted by ^. Let $n_h^{(a)}$ be the number of objects in cluster c_h according to $\lambda^{(a)}$ and $n_l^{(b)}$ be the number of objects in cluster c_l according to $\lambda^{(b)}$. Let $n_{h,l}$ represents the number of objects that are in c_h according to $\lambda^{(a)}$ and in cluster c_l according to $\lambda^{(b)}$. The symbol $\phi^{(NMI)}$ is denoted as the estimation of NMI.

$$\phi^{(NMI)}(\lambda^{(a)}\lambda^{(b)}) = \frac{\sum_{h=1}^{k^{(a)}} \sum_{l=1}^{k^{(b)}} n_{h,l} \log(\frac{n.n_{h,l}}{n_h^{(a)} n_l^{(b)}})}{\sqrt{(\sum_{l=1}^{k^{(b)}} n_l^{(b)} \log(\frac{n_l^{(b)}}{n}))(\sum_{h=1}^{k^{(a)}} n_h^{(a)} \log(\frac{n_h^{(a)}}{n}))}} \tag{8}$$

3.3 Cluster Size

Cluster size can determine the quality of clusters produced in clustering by any algorithm. It is also computed as the number of clusters, produced from the clustering results.

3.4 Graph size

It is obtained by computing the total number of nodes of the input graph. It is a basic parameter used in testing the behaviour of algorithms with different approach.

4. Experimental Results and Discussions

The efficiency and robustness of the RNSC and MCL algorithm are to be tested on few benchmark power-law graphs. To carry out the experiments, it needs real and synthetic data sets as input of the algorithm. The performance of the algorithms will be verified by comparing the clustering results of both the algorithms.

All the experiments are carried out with the following initial configuration for RNSC and MCL. For RNSC, the following parameters are set like as d (diversification Length) = 10; D (shuffling Frequency) = 40; t (tabu-length) = 250 and e (number of experiments) = 1000 and in case of MCL, the inflation (I) value is 4; reweight loops c= 0. 25; pre-inflation value p= 0. 8 and preset resource scheme= 5.

4.1 Evaluation on Real-World Network Datasets

All the evaluations of the performance behaviour of RNSC and MCL are carried out by using some of the real-world datasets like scale-free networks and using computer-generated benchmark synthetic scale-free datasets. Here in this section, all the real scale-free graphs, taken for the experiments, are shown in table 1 with the detailed parametric knowledge.

Table 1. Real power-law Network Data sets

Real Networks	**Graph Size**	**Average Degree <k>**	**Degree exponent (γ_{out})**	**Degree exponent (γ_{in})**
Electronic circuits [23]	329	3.17	2.5	2.5
Protein, S. Cerev [24]	985	1.83	2.5	2.5
Software [25]	1376	6.39	2.5	2.5
Protein, S. cerev. [26]	1870	2.39	2.4	2.4
Internet, router [27]	3,888	2.57	2.48	2.48
Internet, domain [27]	4,389	3.76	2.2	2.2
Prot. Dom. (PromDom) [28]	5995	2.33	2.5	2.5

4.1.1 Cost of Clustering vs Increasing Graph Size for RNSC and MCL

This table comprises of the cost of clustering results, produced by RNSC and MCL. The computation of the cost is performed on real scale-free network with increasing graph size.

Table 2. Cost of Clustering with increasing Graph Size of Real Scale-free network

Real Scale-free Networks	**Cost of Clustering (RNSC)**	**Cost of Clustering (MCL)**
Electronic circuits	20809.51	35552.36
Protein, S. Cerev	197488.1	322559.9
Software	452564.8	629417.7
Protein, S. cerev.	781375.4	1163968
Internet, router	3233392	5032580
Internet, domain	4497642	6411685
Prot. Dom. (PromDom)	7738787	9652835

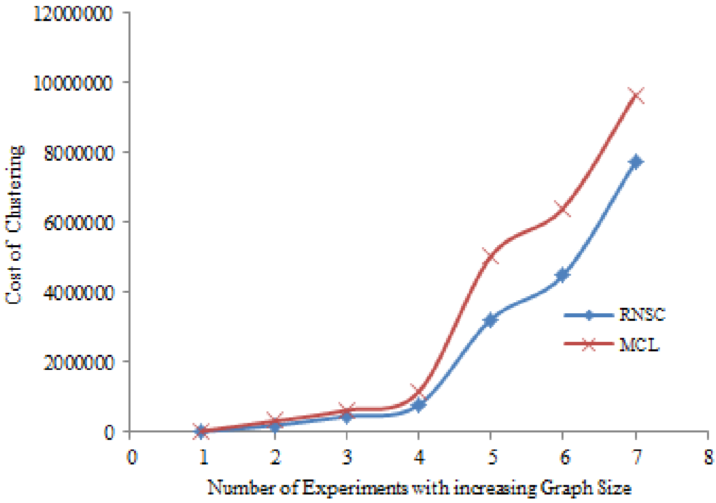

Figure 1. Cost of clustering on real scale-free graphs

It is observed from figure 1 that the cost of clustering is high in case of MCL compared to RNSC with increasing graph size. The cost is increasing gradually for all the test dataset in case of both algorithms. But, RNSC is behaving less costly compared to MCL. It can be concluded that RNSC is producing clusters with lower cost compared to MCL.

4.1.2 Modularity of Clustering Results vs Increasing Graph Size for RNSC and MCL

Table 3 gives the facts about the entire computed modularity index of clustering results, produced by RNSC and MCL. The evaluation of modularity is done on real scale-free network with increasing graph size.

Table 3. Modularity of Clustering with increasing Graph Size of Real Scale-free network

Real Scale-free Networks	**Modularity Index (RNSC)**	**Modularity Index (MCL)**
Electronic circuits	-107.417	-14.6429
Protein, S. Cerev	-354.502	-46.9323
Software	-632.844	-57.5208
Protein, S. cerev.	-891.266	-117.11
Internet, router	-1652.29	-243.728
Internet, domain	-2343.96	-291.768
Prot. Dom. (PromDom)	-2523.57	-390.877

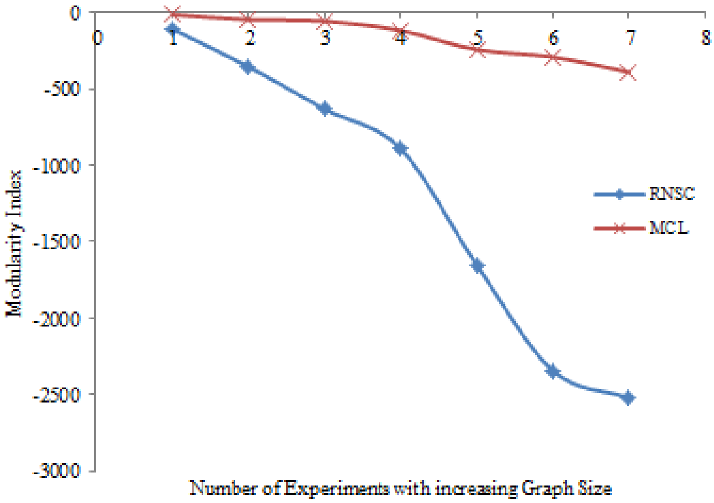

Figure 2. Computation of Modularity index on real scale-free graph

Modularity Index is an important measurement technique to check the performance or accuracy of the clustering results of different graph clustering methods. It is also used to compute the strength of the clusters, produced during clustering. Figure 2 shows that the modularity index of clustering results is decreasing in case of both the algorithms with increasing graph size. The modularity index of RNSC's clustering results is gradually lowering compared to MCL's modularity index. It can be stated by observing the figure that MCL gives clusters with better modularity compared to RNSC's clusters.

4.1.3 Cluster Size vs Increasing Graph Size for RNSC and MCL

Table 4 represents the cluster size values which are produced during clustering by RNSC and MCL algorithms. The computation of cluster size is done on real scale-free network with increasing graph size.

Table 4. Cluster size with increasing Graph Size of Real Scale-free network

Real Scale-free Networks	Cluster Size (RNSC)	Cluster Size (MCL)
Electronic circuits	187	54
Protein, S. Cerev	602	183
Software	805	397
Protein, S. cerev.	1311	253
Internet, router	2589	700
Internet, domain	3084	572
Prot. Dom. (PromDom)	4003	1005

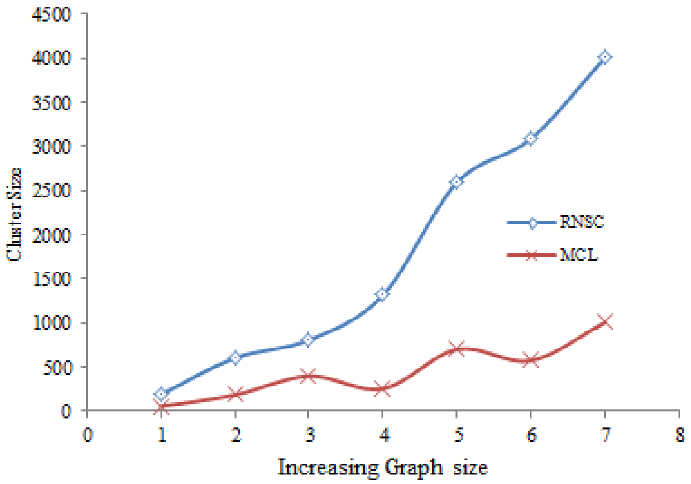

Figure 3. Cluster size computation on real scale-free graph

It is clearly observed from figure 3 that the cluster size evaluation is performed correctly by RNSC and MCL algorithms. From the figure, it can assume that RNSC is producing more number of clusters compared to MCL's clusters. It can state that RNSC is exploring more number of clusters compared to MCL's exploration. In this case, RNSC may possibly be more accurate than MCL.

4.2 Evaluation on Synthetic Graph

Synthetic benchmark scale-free graphs with increasing graph size are used for the performance evaluation of these graph clustering algorithms.

4.2.1 Cost of Clustering vs Increasing Graph Size for RNSC and MCL

It is observed from figure 4 that the cost of clustering evaluation curve of RNSC and MCL is increasing gradually with increasing of graph size. But, RNSC is giving less cost compared to MCL for all the test graphs.

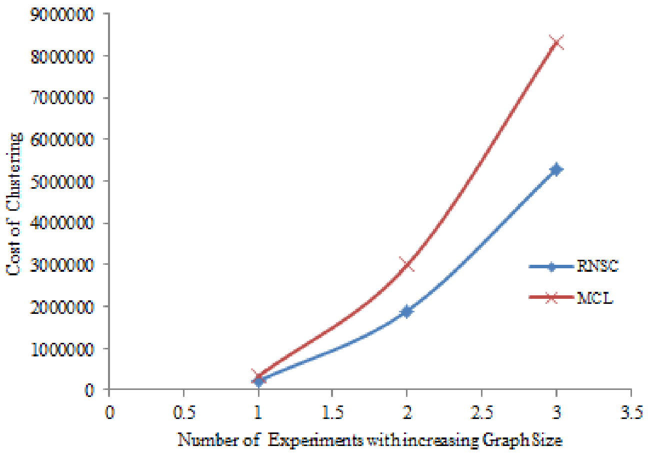

Figure 4. Cost of clustering on synthetic scale-free graph

It can be concluded that RNSC is producing less costly clusters compared to MCL for the synthetic graphs also.

4.2.2 Cluster Size vs Increasing Graph Size for RNSC and MCL

Figure 5 shows that the cluster size is evaluated for RNSC and MCL for all the test graphs and RNSC is producing more number of clusters compared to MCL's clusters for all the test cases. RNSC is exploring the network more compared to MCL.

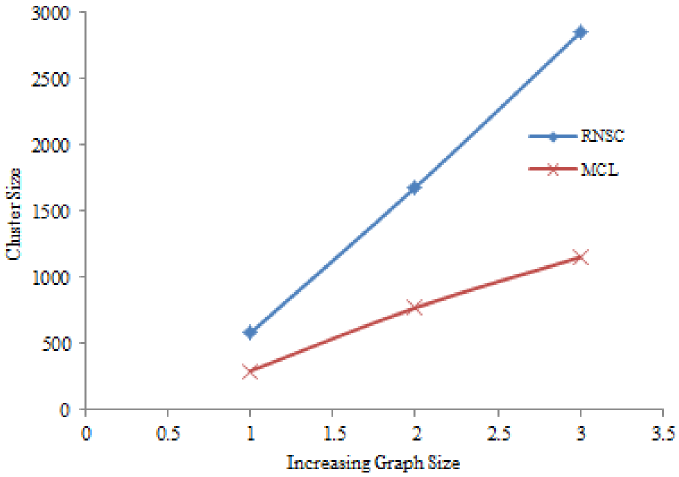

Figure 5. Evaluation of cluster size on scale-free graph

It can be concluded that RNSC is giving more number of meaningful clusters compared to MCL's clusters.

4.2.3 Modularity of Clustering Results vs Increasing Graph Size for RNSC and MCL

Modularity Index is an important approach to check the performance or correctness of the clustering results of different graph clustering methods.

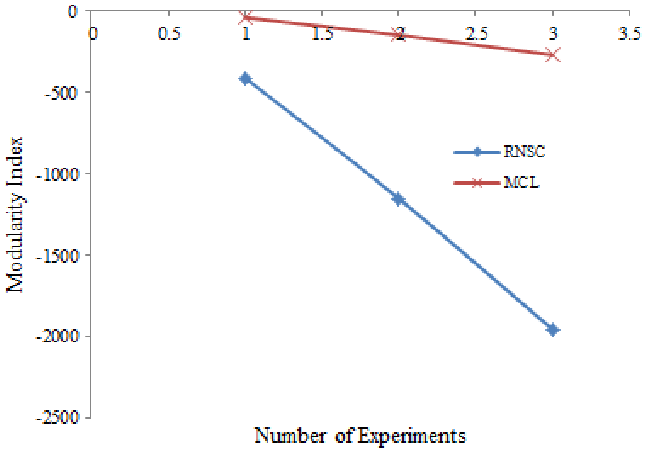

Figure 6. Modularity computation on scale-free graph

It is observed from figure 6 that RNSC's modularity is decreasing gradually whereas MCL is performing better in that case. Modularity of clustering results evaluated by MCL is decreasing with increasing of graph size but it is better compared to RNSC's modularity of evaluated clustering results.

4.3 NMI Value vs Number of Experiments for RNSC and MCL on real scale-free network

The NMI value plays an important role in checking the optimal nature of clusterings of different clustering methods. It evaluates the algorithm's behaviour in information passing through different clustering results.

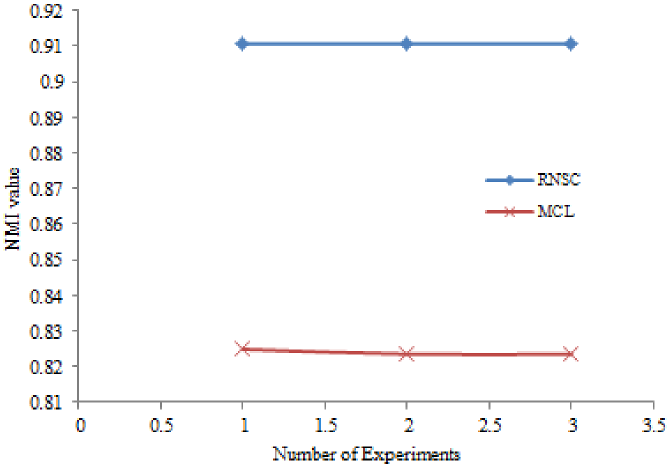

Figure 7. NMI value computation on real power-law graph (Prot. Dom.)

Figure 7 shows that the NMI value is high in case of RNSC compare to MCL. So the quality of the clusters of RNSC is better compared to MCL. After 300, 500, 700 runs with using real large-scale scale-free graph (Prot. Dom. [15]), NMI value is obtained in case of RNSC and in case of MCL; experiments are performed by changing the inflation value as I= {2.5, 3.5, 4.5}. The mutual information sharing between clusterings is more effective for RNSC whereas MCL can't provide good quality clusters due to the less NMI value compare to RNSC. For all the three experiments, the NMI value of RNSC's clustering is stable and in a much high position compared to the MCL's NMI value of clustering results on real scale-free networks. MCL is not giving accuracy in producing optimal clusters compared to RNSC. It can be concluded that RNSC is producing meaningful clusters compared to MCL's produced clusters. So, RNSC is more optimal than MCL.

4.4 Visualization of Real power-law graph and Clustering results of RNSC and MCL on real power-law graph

Figure 8 shows the visualization of real scale-free graph Protein, S. Cerev. It is observed from figure 8 that it is a complex model with 985 nodes and huge interactions exist between the nodes. Figure 11 shows the visualization of real scale-free graph Protein, S. cerev. It is also a complex model of 1870 nodes with huge interactions and it maintains the scale-free nature i.e. power-law distribution. It is observed from figure 9 and figure 10 that the clusters, evaluated by RNSC are more accurate and clearly visible compared to MCL's cluster evaluation on Protein, S. Cerev graph respectively. Figure 12 and figure 13 show the same results in visualizing the clusters, produced by RNSC and MCL on Protein, S. cerev. RNSC always produces meaningful clusters compared to MCL. For both the test graphs, RNSC is performing better in producing clusters compared to MCL. It is clear from visualizations that RNSC is more optimal compared to MCL. Figure 14 and figure 15 show the size distribution of clustering results, produced by RNSC and MCL respectively on Protein, S. Cerev graph. The modularity marking after clustering process is basically used to shrink the cluster size, computed by these methods, following some similarity measures i.e. depend on various properties of a complex network. The figures show that RNSC is responding better in shrinking cluster size compared to MCL's response to modularity shrinking size. It can be concluded that the shrinking size is done for RNSC better compared to MCL. RNSC is more appropriate compared to MCL.

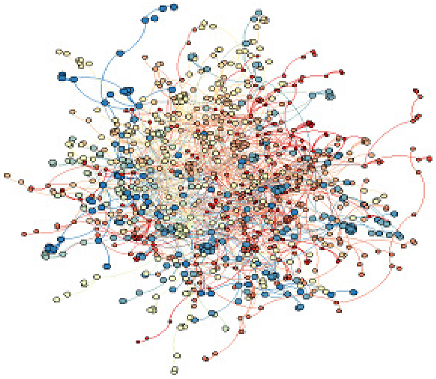

Figure 8.Visual representation of Protein, S. Cerev [24]

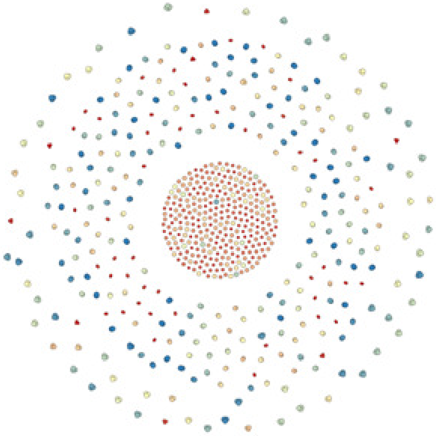

Figure 9. Visual representation of RNSC's clustering on Protein, S. Cerev [24]

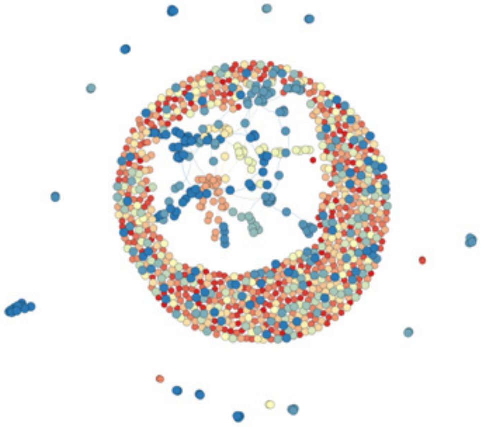

Figure 10. Visual representation of MCL's clustering on Protein, S. Cerev [24]

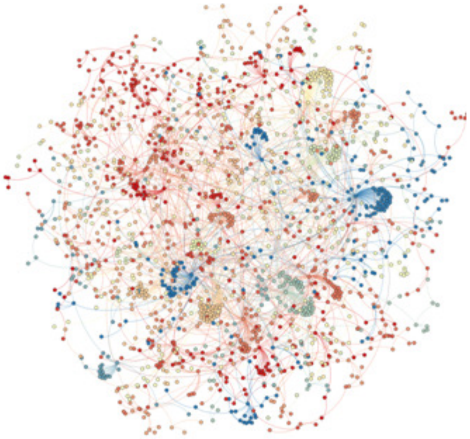

Figure 11. Visual representation of Protein, S. cerev. [26]

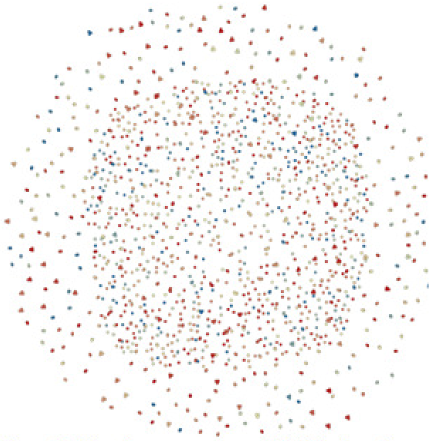

Figure 12. Visual representation of RNSC's clustering on Protein, S. cerev. [26]

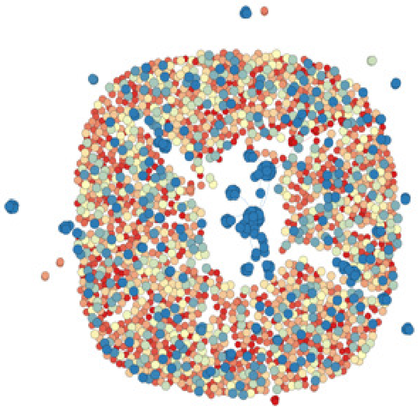

Figure 13. Visual representation of RNSC's clustering on Protein, S. cerev. [26]

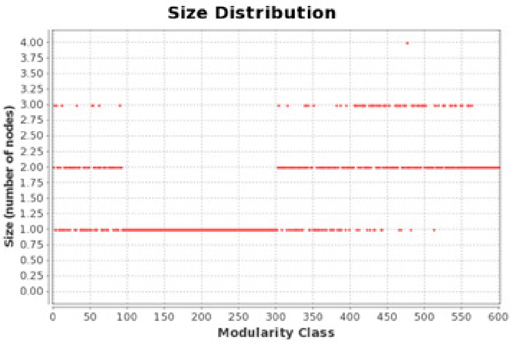

Figure 14. Modularity controlled clusters marking in RNSC's clustering on Protein, S. Cerev [24]

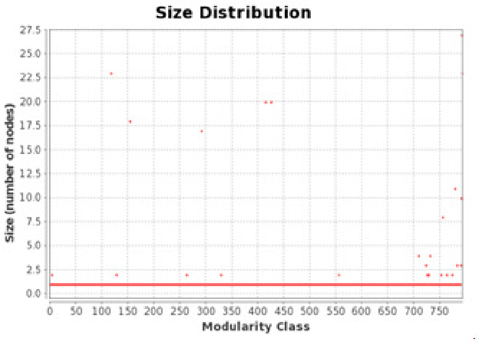

Figure 15. Modularity controlled clusters marking in MCL's clustering on Protein, S. Cerev [24]

5. CONCLUSIONS

This paper presents a comparison between RNSC and MCL algorithm on real and synthetic benchmark scale-free graphs in terms of cost of clustering, modularity index of clustering results, cluster size and quality of clusters on the basis of NMI value. Robustness and optimality of evaluated clustering results of RNSC and MCL algorithms are computed based on above mentioned parameters. The result shows that RNSC is more optimal than MCL. RNSC is getting better NMI value compared to MCL using real scale-free graphs. The quality of the clusters found in RNSC is better compared to MCL. It is clearly observed from the cluster size figure for both test datasets that RNSC can find more number of clusters compared to MCL. So RNSC is more significant compared to MCL. The modularity curve is showing better results in case of MCL's clustering compared to RNSC's clustering for both the test datasets. The cost curve shows that RNSC is producing lower-cost clustering results compared to MCL. It can be concluded that for both the case of real and synthetic benchmark scale-free graphs, RNSC is performing well compared to MCL in producing quality clusters with lowering cost. From the visualization, one's attention can be attracted certainly on RNSC's clustering results compared to MCL's clustering on real scale-free graph.The time complexity of RNSC is O(n^3).The time complexity of MCL is O(n.k^2) where n is the number of nodes and k is the number of resources allocated per node. RNSC can be further extended by implementing it for weighted and directed graph where the weight can be added to the cost functions (naive and scaled cost), which will change and will give better results. Also, it can be further extended by a parallel move method which will give better results in the case of run-time or average cost. MCL can be further extended to produce good quality clusters.

6. ACKNOWLEDGEMENTS

The first author would like to thankfully acknowledge the research scholarship awarded by Banaras Hindu University, Varanasi.

7. REFERENCES

[1] Donath , W.E. and Hoffman, A.J., (1973) “Lower Bounds for the Partitioning of Graphs”, IBM J. Research and Development, vol. 17, pp. 422-425.

[2] Hall, K.M., (1970) “An R-Dimensional Quadratic Placement Algorithm”, Management Science, vol. 11, no. 3, pp. 219-229.

[3] Schaeffer, Sauta Elisa, (2007) “Survey Graph clustering”, Elsevier Computer Science Review, vol. I, pp. 27-64.

[4] Cai, W., Chen, S. and Zhang, D., (2007) “Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation”, Pattern Recognition 40,825–838.

[5] Wu, Z. and Leahy, R., (1993) “An optimal graph theoretic approach to data clustering: theory and its application to image segmentation”, IEEE Transactions on Pattern Analysis and Machine Intelligence 15, 1101–1113.

[6] Han, J. and Kamber, M., (2006) “Data Mining: Concepts and Techniques”, Morgan-Kaufman, San Francisco.

[7] Yu, Z., Wong, H.S. and Wang, H., (2007) “Graph-based consensus clustering for class discovery from gene expression data”, Bioinformatics 23, 2888–2896.

[8] Bandyopadhyay, S., Mukhopadhyay, A. and Maulik, U., (2007) “An improved algorithm for clustering gene expression data”, Bioinformatics 23, 2859–2865.

[9] Chan, P., Schlag, M., and Zien, J., (1994) “Spectral k-Way Ratio Cut Partitioning”, IEEE Trans. CAD-Integrated Circuits and Systems, vol. 13, pp. 1088-1096.

[10] Shi, J. and Malik, J., (2000) “Normalized Cuts and Image Segmentation”, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 888-905.

[11] Ng, A.Y., Jordan, M., and Weiss, Y., (2001) “On Spectral Clustering: Analysis and an Algorithm”, Proc. 14th Advances in Neural Information Processing Systems (NIPS ’01).

[12] King, Andrew Douglas, (2004) “Graph Clustering with Restricted Neighbourhood Search”, M.S Thesis, University of Toronto.

[13] Dongen, S. M. van, (2002) "Graph Clustering by Flow Simulation", PhD thesis, University of Utrecht.

[14] Glover, F., (1989) "Tabu search, part I. ORSA Journal on Computing", 1(3):190-206, summer.

[15] Mladenovi´c, N. and Hansen, P., (1997) "Variable neighbourhood search, *Computers and Operations Research"*, 24(11):1097–1100.

[16] Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N. and Barabasi, A.-L., (2000) "the large-scale organization of metabolic networks", Nature 407, 651 654.

[17] Amaral, L.A.N., Scala, A., Barthelemy, M. and Stanley, H.E., (2000) "Classes of small-world networks", Proc. Natl. Acad. Sci. USA 97, 11149-11152.

[18] Faloutsos, M., Faloutsos, P. and Faloutsos C., (1999) "on power-law relationships of the internet topology", Proceedings of the ACM SIGCOMM, Comput. Commun. Rev.29, 251–262.

[19] Barabasi, A.-L., Albert, R. and Jeong, H., (1999) "Mean-Keld theory for scale-free random networks", Physica A 272, 173–187.

[20] Dongen, S. M. van, (2000) "A cluster algorithm for graphs", Technical Report INS-R0010, Centrum voor Wiskunde en Informatica.

[21] Newman, MEJ and Girvan, M., (2004) "Finding and evaluating community structure in networks", Physical Review E, 69, 026113– 026127.

[22] Strehl, A. and Ghosh,J., (2002) "Cluster ensembles - a knowledge reuse framework for combining partitionings", AAAI, 93–98.

[23] Cancho, R. F. i, Janssen, C. and Sole, R. V., (2001) "Topology of technology graphs: small world patterns in electronic circuits", *Phys. Rev. E*, vol. 64, 046119.

[24] Wagner, A., (2001) "the Yeast Protein Interaction Network Evolves Rapidly and Contains Few Redundant Duplicate Genes", *Mol. Biol. Evol.*, **18**, 1283-1292.

[25] Valverde, S., Ferrer-Cancho, R. and Sole, R. V., (2002) "Scale-Free Networks from optimal design", arXiv: cond-mat/0204344.

[26] Jeong, H., Mason, S., Barabási, A.-L., and Oltvai, Z. N., (2001) "Lethality and centrality in protein networks", Nature, 411, 41-42.

[27] Faloutsos, M., Faloutsos, P., and Faloutsos, C., (1999) "on power-law relationships of the Internet topology", *Comput. Commun. Rev.*, 29, 251-262.

[28] Wuchty, S., (2001) "Scale-Free Behavior in Protein Domain Networks", *Mol. Biol. Evol.*, 18, 1699-1702.

Authors

Mousumi Dhara: She completed her M. Tech in computer Science and Engineering from NIT Durgapur. She is pursuing her Ph.D. since 2010 in the Department of Computer Engineering, IIT (BHU), Varanasi.

K .K. Shukla: He is professor of Department of Computer Engineering, Indian Institute of Technology, Banaras Hindu University, Varanasi.