Correlation coefficient analysis: centrality vs. maximal clique size for complex real-world network graphs

Natarajan Meghanathan

Jackson State University,
Mailbox 18839, 1400 John R. Lynch Street,
Jackson, Mississippi, MS 39217, USA
Email: natarajan.meghanathan@jsums.edu

Abstract: The high-level contribution of this paper is correlation analysis between the centrality values observed for nodes (a computationally lightweight metric) and the maximal clique size (a computationally hard metric) that each node is part of in complex real-world network graphs. The real-world network graphs studied range from regular random network graphs to scale-free network graphs. The maximal clique size for a node is the size of the largest clique (in terms of the number of constituent nodes) the node is part of. We observe the degree-based centrality metrics such as the degree centrality and eigenvector centrality to be relatively better correlated with the maximal clique size compared to the shortest path-based centrality metrics such as the closeness centrality and betweenness centrality. As the real-world networks get increasingly scale-free, we observe the correlation between the centrality value and the maximal clique size to increase.

Keywords: assortativity index; centrality; correlation; maximal clique size; network graph; random network; scale-free network.

Reference to this paper should be made as follows: Meghanathan, N. (2016) 'Correlation coefficient analysis: centrality vs. maximal clique size for complex real-world network graphs', *Int. J. Network Science*, Vol. 1, No. 1, pp.3–27.

Biographical notes: Natarajan Meghanathan has completed his BTech in Chemical Engineering from Anna University, India; MS in Computer Science from Auburn University, Alabama, USA and PhD in Computer Science from The University of Texas at Dallas, USA. He specialises in the areas of wireless ad hoc networks and sensor networks, network science and graph theory, cyber security and computational biology. His research has been funded through the US National Science Foundation, the Army Research Lab, National Aeronautics and Space Administration (NASA) and the Air Force Office of Scientific Research. He is serving as the Editor-in-Chief of three international journals and is an active member in the editorial boards of more than ten journals as well as in the organising and technical committees of several international conferences. Presently, he is working as an Associate Professor of Computer Science at Jackson State University, Mississippi, USA.

1 Introduction

Network science is a fast-growing discipline in academics and industry. It is the science of analysing and visualising complex real-world networks using graph theoretic principles. Several metrics are used to analyse the characteristics of the real-world network graphs; among them 'centrality' is a commonly used metric. The centrality of a node is a measure of the topological importance of the node with respect to the other nodes in the network (Newman, 2010). It is purely a link-statistics based measure and not based on any offline information (such as reputation of the node, cost of the node, etc.). The commonly used centrality metrics are degree centrality (DegC), eigenvector centrality (EVC), closeness centrality (ClC) and betweenness centrality (BWC). DegC of a node is simply the number of immediate neighbours for the node in the network. The EVC of a node is a measure of the degree of the node as well as the degree of its neighbour nodes. We refer to DegC and EVC as degree-based centrality metrics. ClC of a node is the inverse of the sum of the shortest path distances of the node to every other node in the network. BWC of a node is the ratio of the number of shortest paths the node is part of for any source-destination node pair in the network, summed over all possible source-destination pairs that do not involve the particular node. We refer to ClC and BWC as shortest path-based centrality metrics. Computationally efficient polynomial-time algorithms have been proposed in the literature (Strang, 2005; Cormen et al., 2009; Brandes, 2001; Newman, 2010) to determine exact values for each of the above centrality metrics; hence we categorise centrality as a computationally lightweight metric.

A 'clique' is a complete sub graph of a graph (i.e., all the nodes that are part of the sub graph are directly connected to each other). Cliques are used as the basis to identify closely-knit communities in a network as part of studies on homophily and diffusion. Unfortunately, the problem of finding the maximum-sized clique in a graph is an NP-hard problem (Cormen et al., 2009), prompting several exact algorithms and heuristics to be proposed in the literature (Tomita and Seki, 2003; Palla et al., 2005; Fortunato, 2010; Sadi et al., 2010; Pattabiraman et al., 2013). In this paper, we choose a recently proposed exact algorithm (Pattabiraman et al., 2013) to determine the size of the maximum clique for large-scale complex network graphs and extend it to determine the size of the maximal clique that a particular node is part of. We define the maximal clique size for a node as the size of the largest clique (in terms of the number of constituent nodes) the node is part of. Note that the maximal clique for a node need not be the maximum clique for the entire network graph; but, the maximum clique for the entire graph could be the maximal clique for one or more nodes in the network.

Since the maximal clique size problem is a computationally hard problem and exact algorithms run significantly slower on large network graphs, our goal in this paper is to explore whether the maximal clique size correlates well to one of the commonly studied computationally lightweight metrics, viz., centrality of the vertices, for complex real-world network graphs: if we observe a high positive correlation between maximal clique size and one or more centrality metrics, we could then infer the maximal clique size of the vertices to be directly correlated to the centrality values of the corresponding vertices in real-world network graphs. Ours will be the first paper to conduct a correlation study between centrality and maximal clique size for real-world network graphs. To the best of our knowledge, we did not come across such a paper that has done correlation

study between these two metrics (and in general, a computationally hard metric vis-a-vis a computationally lightweight metric) for real-world network graphs.

We hypothesise the degree-based centrality metrics to exhibit a high positive correlation with maximal clique size, compared to that observed with the shortest path-based centrality metrics. From the results obtained for real-world network graphs, we observe our hypothesis to be true: the eigenvector-based centrality metric shows a high positive correlation to the maximal clique size and the BWC metric shows a low correlation to the maximal clique size. The high positive correlation between EVC and maximal clique size indicates that a high degree vertex present in a neighbourhood of high degree vertices is likely to be part of larger cliques; on the other hand, nodes that are part of a maximal sized clique need not always play the central role in facilitating communication between any two nodes in the network; rather it is more likely the nodes whose maximal clique is smaller (i.e., at the best are part of smaller-sized cliques) play a central role in facilitating communication between various nodes and communities in the network (a measure of BWC). We observe the correlation between the centrality metrics and maximal clique size to increase with increase in the variation of node degrees in the network. As we transition from a regular random network (variation in node degree is minimum and all nodes have comparable degrees) to a scale-free network (variation in node degrees is the maximum; a majority of the nodes have low degree, but there are some appreciable number of high-degree nodes), we observe the correlation coefficient between the centrality value and maximal clique size for the nodes to increase. In addition to the above correlation study, we also analyse the distribution of the maximal clique size of the nodes in the real-world network graphs as well as analyse the assortativity index of the vertices in these graphs with respect to both maximal clique size and node degree. We observe the frequency distribution of the maximal clique size of the nodes to resemble a Poisson distribution for five of the six real-world network graphs considered in this study while the remaining network graph exhibits an exponential-style distribution for the maximal clique size of the nodes.

The rest of this paper is organised as follows: Section 2 discusses related work and motivates the need for a correlation coefficient analysis between centrality and maximal clique size. Section 3 reviews the four centrality metrics studied in this paper and briefly discusses the algorithmic approach to determine them on network graphs. Section 4 describes an efficient branch-and-bound technique based exact algorithm proposed in the literature to determine the maximum clique size for massively large complex network graphs and our extensions to the algorithm to determine the size of the maximal clique that an individual node is part of (so that it can be applied for every node in the network). Section 5 describes the six real-world network graphs that are used in this paper and presents an analysis of the degree distribution of the vertices and the distribution of the maximal clique size of the vertices in these graphs. The section also presents an analysis of the assortativity index of the vertices in the real-world network graphs with respect to node degree and maximal clique size. Section 6 presents the results of the correlation studies between centrality and maximal clique size at the node level for each of the real-world network graphs. Section 7 concludes the paper. Throughout the paper, we use the terms ‘node’ and ‘vertex’ as well as ‘link’ and ‘edge’ interchangeably. They mean the same.

2 Related work and motivation

To the best of our knowledge, ours is the first work to focus on correlation coefficient analysis between a computationally hard metric (maximal clique size for the individual vertices) with that of a computationally lightweight metric (centrality values of individual vertices) for complex real-world network graphs. The work available in the literature so far considers these two metrics separately. Recently, Li et al. (2014) conducted a correlation coefficient analysis study among the centrality metrics for real-world network graphs. Centrality metrics have been widely studied for analysis and visualisation of complex networks in several domains, ranging from biological networks to social networks (e.g., Koschutzki and Schreiber, 2008; Opsahl et al., 2010). The research focus with regards to cliques in the context of complex networks is to come up with efficient heuristics to reduce the run-time complexity in determining the maximum size clique for the entire network graph. Though branch-and-bound has been the common theme among these works, the variation is in the approach used to arrive at the bounds and enforce them in the search space. Strategies used for pruning the search space are typically based on node degree (e.g., Pattabiraman et al., 2013), vertex ordering (e.g., Carraghan and Pardalos, 1990) and vertex colouring (e.g., Ostergard, 2002). Recently, Rossi et al. (2014) proposed a parallelised branch and bound approach for determining cliques in real-world networks ranging from 1,000 to 100 million nodes. Nevertheless, none of the research so far has focused on identifying correlation between the maximal clique size for an individual vertex (the size of the largest clique that a particular vertex is part of) with any of the commonly studied metrics (like centrality) for network analysis. Ours is the first step in this direction. With the problem of determining maximum size clique for the entire network graph and maximal size cliques for the individual vertices being NP-hard and computationally time-consuming for complex real-world networks of larger size, it becomes imperative to analyse the correlations of the maximal clique size values of the individual vertices with that of the network metrics that can be easily computed so that meaningful inferences about maximal clique size values can be made.

3 Centrality metrics

This section discusses the four centrality metrics that are studied in this paper. We discuss the algorithm/approach used to determine each of them and also illustrate the same with an example and figures. The highest ranked vertex or set of vertices with respect to each of the centrality metrics is shown shaded in the graphs of these figures. The algorithms to determine the centrality metrics use the adjacency matrix of the network graph as the basis; it is a binary matrix where there is a 1 in the i^{th} row and j^{th} column, if there is an edge from vertex i to vertex j and 0 otherwise.

3.1 Degree centrality

DegC of a vertex is the number of neighbours adjacent to it. The larger the number of neighbours for a vertex, the higher the DegC of the vertex. We determine DegC by simply multiplying the adjacency matrix with a column vector of 1s $[1\ 1\ 1\ 1\ \ldots\ 1\ 1]^T$, where the number of 1s is the number of vertices in the network graph. Since the number of neighbours for a vertex can take only discrete values, it is possible that two or more

vertices could have identical values for DegC. In such cases, the ranking among the vertices has to be either arbitrarily broken or based on some ordering among the vertices (like the vertex ID). Figure 1 illustrates an example to compute the DegC.

Figure 1 Example to illustrate the computation of DegC (see online version for colours)

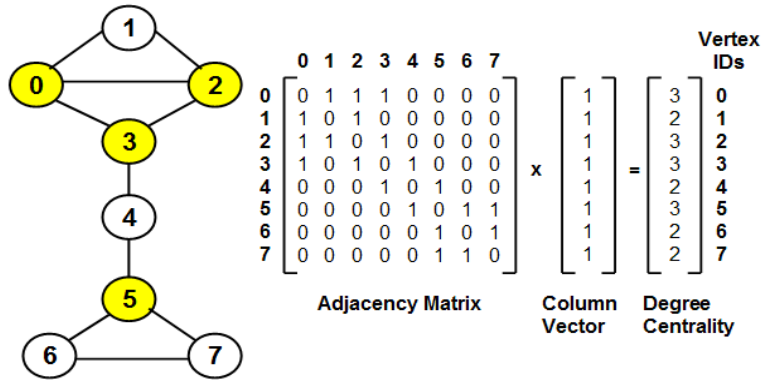
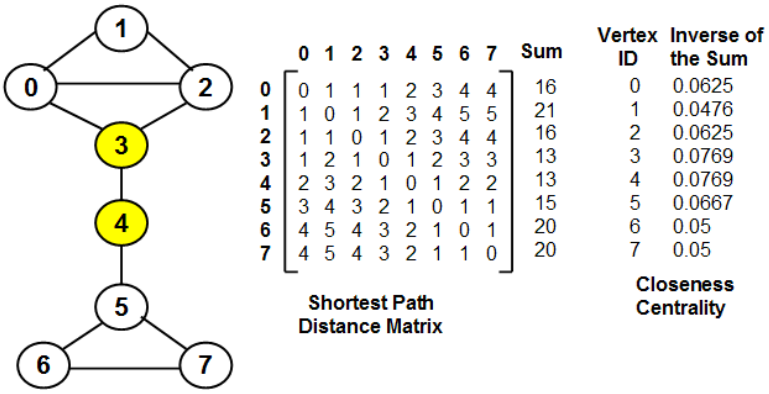


Figure 2 Example to illustrate the computation of ClC (see online version for colours)



3.2 Closeness centrality

The ClC of a vertex is the inverse of the sum of shortest path distances from the vertex to every other vertex. We determine the ClC of a vertex i by running the breadth first search (BFS) algorithm (Cormen et al., 2009) starting from that vertex i; the root of the BFS tree (vertex i) is said to be at level 0, its one-hop neighbours are at level 1, its two-hop neighbours are at level 2 and so on. The length of a shortest path from vertex i to vertex j is the level of vertex j in the BFS tree rooted at vertex i. The sum of the shortest path distances for vertex i is then simply the sum of the level values of all the vertices in the BFS tree rooted at vertex i. Since the sum of the shortest path distances is an integer value, the ClC will take only discrete values; two or more vertices are likely to have identical values for the sum of shortest path distances, leading to ambiguity in the ranking of vertices based on ClC. Figure 2 illustrates an example to determine the ClC of the

vertices in a graph. As can be seen, vertices (3 and 4) that lie in the centre of the network are likely to have smaller values for the sum of shortest path distances, compared to vertices that are in the periphery.

3.3 Eigenvector centrality

EVC of a vertex is a measure of the degree of the vertex as well as the degree of its neighbours. EVC of the vertices are the entries in the principal eigenvector of the adjacency matrix of the graph.

Each vertex has an entry (in the order of the vertex IDs) in the principal eigenvector. The larger the value of the entry for an vertex in the principal eigenvector, the higher is its ranking with respect to EVC. The principal eigenvector is determined by running the power-iteration algorithm (Strang, 2005) on the adjacency matrix of the network graph. The eigenvector X_{i+1} of a network graph at the end of the $(i + 1)^{th}$ iteration is given by: $X_{i+1} = \frac{AX_i}{\| AX_i \|}$, where $\|AX_i\|$ is the normalised value of the product of the adjacency matrix A of a given graph and the tentative eigenvector X_i at the end of iteration i. The initial value of X_i is the transpose of [1, 1, …, 1], a column vector of all 1s, where the number of 1s correspond to the number of vertices in the graph. We continue the iterations until the normalised value $\|AX_{i+1}\|$ converges to that of the normalised value $\|AX_i\|$. The value of the column vector X_i at this juncture is the principal eigenvector of the graph; the entries corresponding to the individual rows in X_i represent the EVC of the vertices of the graph. The converged normalised value of the principal eigenvector is referred to as the *spectral radius*. The EVC values are more likely to be continuous (real numbers); hence, unless two vertices have the same degree as well as connected to the same number of neighbours with identical degree distribution, their EVC values are likely to be different – leading to unambiguous ranking of the vertices with respect to EVC.

Figure 3 Example to illustrate the computation of EVC (see online version for colours)

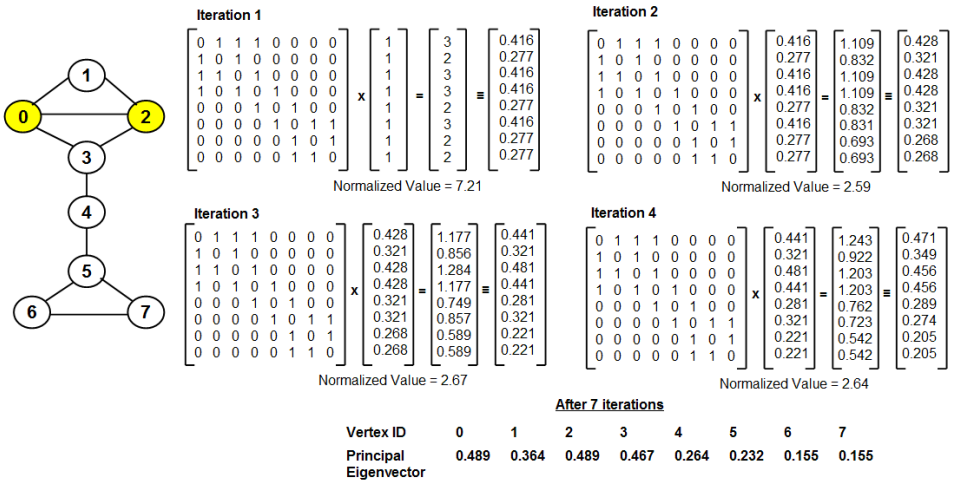


Figure 3 illustrates an example to determine the EVC of the vertices of a network graph. As can be seen in the example, the EVC of a vertex is a function of both its degree as well as the degrees of its neighbours. Vertices 3 and 5 have the same degree (3 neighbours each); however, vertex 3 is connected to two vertices that have a high degree (3) and vertex 5 is connected to three vertices that have a lower degree (2); hence, vertex 3 has a higher EVC than vertex 5.

3.4 Betweenness centrality

BWC is a measure of how significant a node is in facilitating communication between any two nodes in the network. BWC for a node is the ratio of the number of shortest paths a node is part of for any source-destination node pair in the network, summed over all possible source-destination pairs that do not involve the particular node. If the number of shortest paths between two nodes j and k that go through node i as the intermediate node is denoted as $\mathbf{sp}_{jk}(i)$ and the total number of shortest paths between the two nodes j and k is denoted as $\mathbf{sp}_{jk}$, then the BWC for node i is given by: $BWC(i) = \sum_{j \neq k \neq i} \frac{sp_{jk}(i)}{sp_{jk}}$.

The algorithm described below is a BFS-based variation of the computationally efficient algorithm proposed by Brandes (2001).

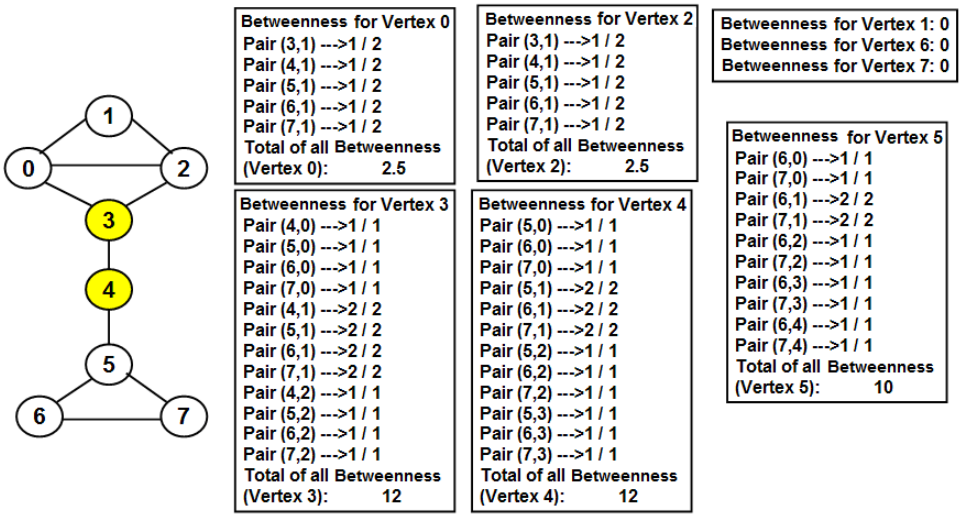
The number of shortest paths from a node j to all other nodes k in an undirected graph can be determined by running the well-known BFS algorithm on the graph, starting from vertex j (which is also considered to be at level 0 for this BFS run). All the vertices that are directly reachable from vertex j are said to be at level 1; the two hop neighbours of vertex j are at level 2 and so on. Though the BFS algorithm primarily determines a shortest path tree rooted at vertex j, the level of a vertex k on this BFS tree (i.e., the minimum number of hops from the root j to vertex k) can be used to determine the number of shortest paths from the root vertex j to the vertex k. The number of shortest paths from the root j (at level 0) to itself is set to be 1. For any other vertex k (at level l, where $l > 0$) on this shortest path BFS tree rooted at j: the number of shortest paths from j to k ($\mathbf{sp}_{jk}$) is the sum of the number of shortest paths from j to each of the neighbours of k (in the original graph) that are at level $l - 1$ on the BFS tree.

The number of shortest paths between two nodes j and k that go through node i (i.e., $\mathbf{sp}_{jk}(i)$) is simply the maximum of the number of shortest paths from vertex j to i and the number of shortest paths from vertex k to i. This can be determined from the BFS trees rooted at vertices j and k using the approach described earlier. However, the above assertion holds true (i.e., $\mathbf{sp}_{jk}(i) > 0$) only if node i lies on at least one shortest path between j and k. We test this by keeping track of the set of predecessors at all levels ($< l$; $l > 0$) for a vertex k (at level l; $l > 0$) in the BFS tree rooted at vertex j and vice-versa. Accordingly, the set of predecessors for a vertex k at level l in a BFS tree rooted at vertex j is the union of all the neighbour vertices of k (in the original graph) at level $l - 1$ in the BFS tree (rooted at j) as well as the union of the sets of predecessors of all these neighbour vertices. For an undirected graph, to test whether vertex i is on one of the shortest paths from vertex j to k, it is enough to test whether node i is one among the predecessors for vertex k on the BFS tree rooted at vertex j.

Figure 4 illustrates an example to compute the BWC of the vertices in the example graph that is also used in Figures 1 to 3. We notice that BWC-based ranking of the vertices is different from the DegC and EVC-based ranking of the vertices. The degree

and EVC-centralities take into consideration only the degree of the vertices and they are positively correlated. However, the BWC centrality takes into consideration the contribution of a vertex in facilitating communication between any two vertices in the network on the shortest path; such vertices are likely to be more central to the network and form the backbone or the core of the network. As can be seen in the example run of Figure 4, vertex 4 that was ranked lower with respect to degree and EVC is ranked the highest (along with vertex 3) with respect to BWC. Also note that in this example graph, no source-destination pair need to go through vertices 1 or 6 or 7 on a shortest path. As a result, the BWC for each of these three vertices (1, 6 and 7) is 0.

Figure 4 Example to illustrate the computation of BWC (see online version for colours)



4 Clique

A clique is a sub graph of a graph in which all the vertices are adjacent to each other. The problems of finding maximum size clique for the entire graph as well as the maximal size cliques for the individual nodes are NP-hard problems (Cormen et al., 2009). Several exact algorithms (that at the worst case incur exponential time for a NP-hard problem) have been proposed to determine maximum size cliques for sparse graphs. Recently, with the surge in interest to analyse large real-world networks from a graph theoretic point of view, researchers have proposed efficient exact algorithms (e.g., Tomita and Seki, 2003; Palla et al., 2005; Fortunato, 2010; Sadi et al., 2010; Pattabiraman et al., 2013) to determine maximum size cliques for large/dense graphs. The common theme (Ostergard, 2002) behind these algorithms is a branch and bound approach of searching through all possible candidate cliques and limiting the search to only viable candidate sets of vertices whose agglomeration has scope of being a clique of size larger than the currently known clique found as part of the search; the variation among these exact algorithms is the pruning strategy (the approach taken to compute the bounds and use them) to limit the search. In this section, we will describe one such branch and bound-technique based exact algorithm that has been recently proposed in the literature (Pattabiraman et al.,

2013) to determine maximum size clique in large network graphs and explain our modification to the algorithm so that it can be used to determine the maximal cliques that each vertex in the graph is part of; the largest among these cliques is the maximum size clique for the entire graph.

Figure 5 outlines the pseudo code of the algorithm (proposed originally by Pattabiraman et al., 2013) to determine the maximum size clique for an entire graph. The algorithm starts with an estimate of 0 for the maximum size clique (variable *max*) in the entire graph; the value for *max* is updated as and when a clique of size larger than the latest value of *max* is found. The procedure MAXCLIQUE proceeds in iterations, with each iteration designed to determine the maximum size clique for the entire graph that could also include vertex v_i (considered in the increasing order of the IDs). In a particular iteration, vertex v_i is considered worthy of exploration for presence in a maximum size clique only if its degree is at least the value of *max* at that time (i.e., only vertices that could be part of a clique of size larger than the currently known maximum size clique are considered – a pruning strategy). For each such vertex v_i, a candidate set U of neighbour vertices v_j (whose degree is at least the latest value for *max*) is constructed and passed to the sub routine CLIQUE to find a clique among these vertices; the initial size of the clique is 1 – accounting for v_i.

Figure 5 Exact algorithm to determine the maximum size clique for an entire graph

```
                                    Subroutine CLIQUE(G = (V, E), U, size)
                                      // size is the size of clique found so far
                                        if U = φ then
Procedure MAXCLIQUE (G = (V, E))          if size > max then
  max ← 0                                   max ← size
  for i : 1 to |V| do                     return
    if degree(v_i) ≥ max then           while |U| > 0 do
      U ← φ                               if size + |U| ≤ max then
      for each v_j ∈ Neighbour(v_i)         return
      do
        if degree(v_j) ≥ max then       select any vertex u from U
          U ← U ∪ {v_j}                 U ← U \{u}
      CLIQUE(G, U, 1)                   N'(u) := {w | w ∈ Neighbour(u) ∧
                                        degree(u) ≥ max}
                                        Clique(G, U ∩ N'(u), size + 1)
```

Source: Adapted from Pattabiraman et al. (2013)

The sub routine CLIQUE called with vertex v_i as the first constituent vertex of the largest possible clique involving v_i, expands with one vertex at a time through a combination of iterations and recursions; the sub routine runs as long as the size of the set U is greater than zero or if the current value of *max* is less than the sum of the sizes of the set U and the current clique found so far (a pruning strategy). In each such iteration, a vertex u (that is also a neighbour of the starting vertex v_i and the other vertices in the clique determined so far) is randomly removed from the set U and the neighbours of u that are also present in U (and hence are neighbours of the starting vertex v_i and the other vertices that are part

of the clique found so far) are only further considered to be candidates that could be part of the clique, and a recursive call to the CLIQUE sub routine is made with the value of variable *size* (the size of the largest clique found so far involving vertex v_i) incremented by 1 – accounting for u as the latest entrant in the clique determined so far. Each recursive call to CLIQUE is accompanied by an iteration where a vertex u (that is also a neighbour of the vertices already part of the clique) is removed from the set U passed to the sub routine and only the neighbours of u that are also neighbours of the vertices already in the clique are considered. During any such recursive call, if the size of the set U passed to the sub routine CLIQUE reaches zero, the algorithm terminates the sequence of recursions and updates the value of *max* if the size of the clique determined so far involving vertex v_i is larger than the current value of *max*. During the sequence of returns from the recursive calls, it is possible that a new sequence of recursions and iterations is triggered due to the presence of a neighbour u of v_i that has scope for being in a clique (involving v_i) of size larger than the clique found so far for the entire graph. The algorithm explores all such possible cliques involving vertex v_i that have scope for exceeding the currently known maximum size clique for the entire graph.

Figure 6 Exact algorithm to determine the maximal clique size for each vertex in a graph

```
Procedure MAXIMALCLIQUE (G = (V, E))
    for i : 1 to |V| do
        maximalCliqueSize[v_i] ← 0
        U ← φ
        for each v_j ∈ Neighbour(v_i) do
            U ← U ∪ {v_j}
        CLIQUE(G, v_i, U, 1)

Subroutine CLIQUE(G = (V, E), v_i, U, size) // size is the size of clique found so far for vertex v_i
    if U = φ then
        if size > maximalCliqueSize[v_i] then
            maximalCliqueSize[v_i] ← size
        return
    while |U| > 0 do
        if size + |U| ≤ maximalCliqueSize[v_i] then
                return
        select any vertex u from U
        U ← U \{u}
        N'(u) := {w | w ∈ Neighbour(u) ∧ degree(u) ≥ maximalCliqueSize[v_i]}
        Clique(G, v_i, U ∩ N'(u), size + 1)
```

Source: Adapted from Pattabiraman et al. (2013)

At the end, the algorithm returns the maximum size clique for the entire graph that also happens to be the maximal size clique involving some vertex v_i such that there is no other vertex v_j ($i > j$) that is also part of the clique. Since the algorithm proceeds with vertices in the increasing order of their IDs, if the maximum size clique for the entire graph involves at least one vertex v_i with a smaller ID, the presence of the maximum size clique is detected much earlier and the subsequent iterations (with vertices whose IDs are greater than v_i, but could be part of only cliques of size smaller or equal to the maximum size clique of the entire graph involving v_i) are merely pruned, contributing to the time-efficiency of the algorithm. Hence, the labelling of the vertices with their IDs plays a significant role in the run-time complexity of the algorithm; the algorithm is capable of quickly determining the maximum size clique if the latter comprises of at least one vertex with a smaller ID.

Figure 6 illustrates our modifications (to determine the size of the maximal clique that each vertex is part of) to the pseudo code of the algorithm presented in Figure 5. The trade-off is an increase in the run-time of the algorithm: we cannot just prune our search based on the vertex IDs; we have to explore the neighbourhood of each of the vertices to determine the maximal size clique that each vertex is part of. Since to start with, the maximal size clique known for vertex v_i is 0, there is no need to filter the neighbours of v_i in procedure MAXIMALCLIQUE based on the degree of the neighbours; all neighbours of v_i are included in the set U and passed onto the sub routine CLIQUE. However, we could retain all of the pruning steps in sub routine CLIQUE (called to find the maximal size clique for each of the vertices v_i) and recursive calls to the same: there is no need to explore the neighbours of vertex u whose degree is less than that of the currently known maximal clique size for vertex v_i.

5 Real-world network graphs and their analysis

In this section, we describe the network graphs analysed and illustrate the degree distribution and the distribution of the maximal clique size of the vertices in the network graphs. We do so to understand the topological structure of the real-world network graphs as well as to elucidate the impact of the degree and maximal clique size distribution of the vertices on the correlation between the centrality values and the maximal clique size observed for the vertices. The network graphs analysed are briefly described as follows:

1 *Zachary's Karate Club* (Zachary, 1977): Social network of friendships (78 edges) between 34 members of a karate club at a US university in the 1970s.

2 *Dolphins' Social Network* (Lusseau et al., 2003): An undirected social network of frequent associations (159 edges) between 62 dolphins in a community living off Doubtful Sound, New Zealand.

3 *US Politics Books Network* (Krebs, 2008): Nodes represent a total of 105 books about US politics sold by the online bookseller Amazon.com. A total of 441 edges represent frequent co-purchasing of books by the same buyers, as indicated by the 'customers who bought this book also bought these other books' feature on Amazon.

4 *Word Adjacencies Network* (Newman, 2006): This is a word co-appearance network representing adjacencies of common adjective and noun in the novel 'David Copperfield' by Charles Dickens. A total of 112 nodes represent the most commonly occurring adjectives and nouns in the book. A total of 425 edges connect any pair of words that occur in adjacent position in the text of the book.

5 *US College Football Network* (Girvan and Newman, 2002): Network represents the teams that played in the Fall 2000 season of the American Football games and their previous rivalry – nodes (115 nodes) are college teams and there is an edge (613 edges) between two nodes if and only if the corresponding teams have competed against each other earlier

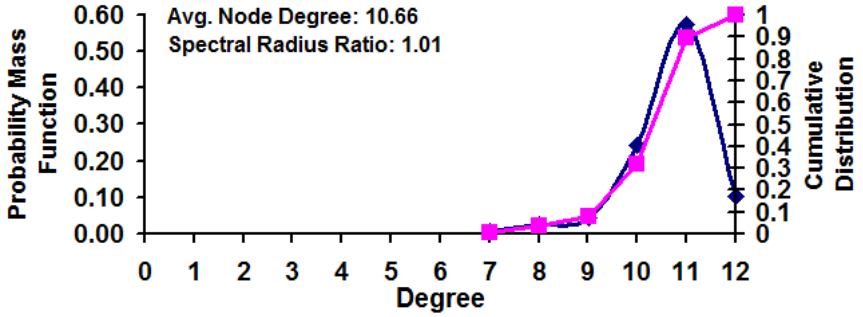
6 *US Airports* 1997 *Network*: A network of 332 airports in the USA (as of 1997) wherein the vertices are the airports and two airports are connected with an edge (a total of 2,126 edges) if there is at least one direct flight between them in both the directions.

Data for networks (1) to (5) can be obtained from http://www-personal.umich.edu/~mejn/netdata/; data for net (6) is at http://vlado.fmf.unilj.si/pub/networks/pajek/data/gphs.htm.

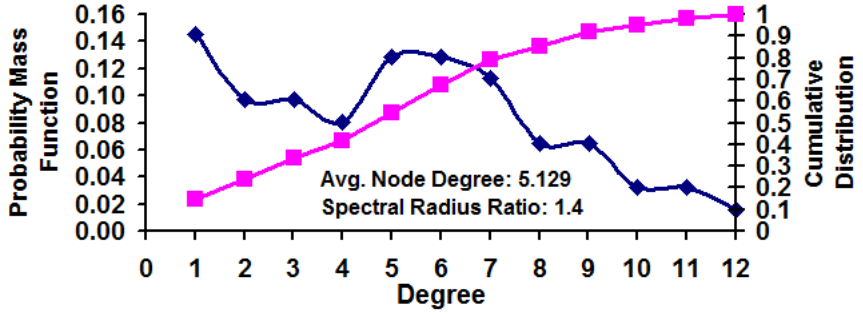
5.1 Degree distribution of the real-world network graphs

Figure 7 presents the degree distribution of the vertices in the six network graphs in the form of both the probability mass function (the fraction of the vertices with a particular degree) and the cumulative distribution function (the sum of the fractions of the vertices with degrees less than or equal to a certain value). We also compute the average node degree and the spectral radius degree ratio (ratio of the spectral radius and the average node degree); the spectral radius (bounded below by the average node degree and bounded above by the maximum node degree) is the largest eigenvalue of the adjacency matrix of the network graph, obtained as a result of computing the EVC of the network graphs. The spectral radius degree ratio is a measure of the variation in the node degree with respect to the average node degree; the closer the ratio is to 1, the smaller the variations in the node degree and the degrees of the vertices are closer to the average node degree (characteristic of random graph networks). The farther the ratio from 1, the larger the variations in degree of the nodes (characteristic of scale-free networks). Figure 7 presents the degree distribution of the network graphs in the increasing order of their spectral radius ratio for node degree (1.01 to 3.23). The US College Football network exhibits minimal variations in the degree of its vertices (each team has more or less played against an equal number of other teams). The US Airports Network exhibits maximum variation in the degree of its vertices (there are some hub airports from which there are flights to several other airports; whereas there are several airports with only fewer connections to other airports). In between these two extremes of networks, we have the other four network graphs, all of which have a spectral radius ratio for node degree around 1.4–1.7, indicating a moderate variation in the node degree.

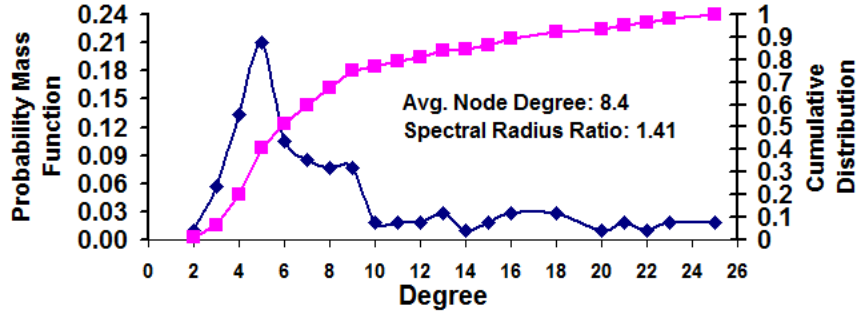
Figure 7 Distribution of node degrees in the real-world network graphs (probability mass function and cumulative distribution), (a) US College Football Network (115 nodes, 613 edges) (b) Dolphins' Social Network (62 nodes, 159 edges) (c) Politics Books Network (105 nodes, 441 edges) (d) Karate Club Network (34 nodes, 78 edges) (e) Word Adjacencies Network (112 nodes, 425 edges) (f) US Airports'97 Network (332 nodes, 2,126 edges) (see online version for colours)



(a)

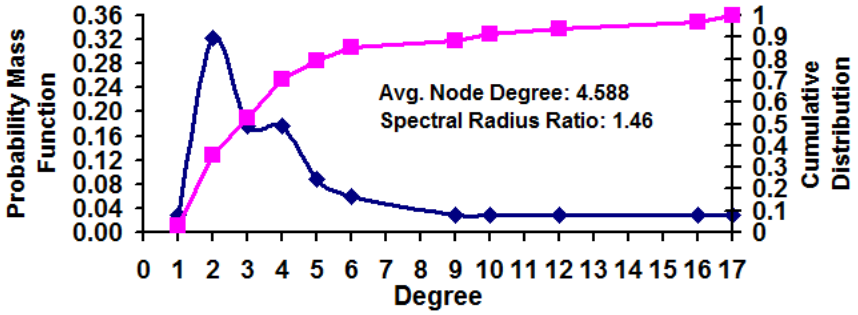


(b)

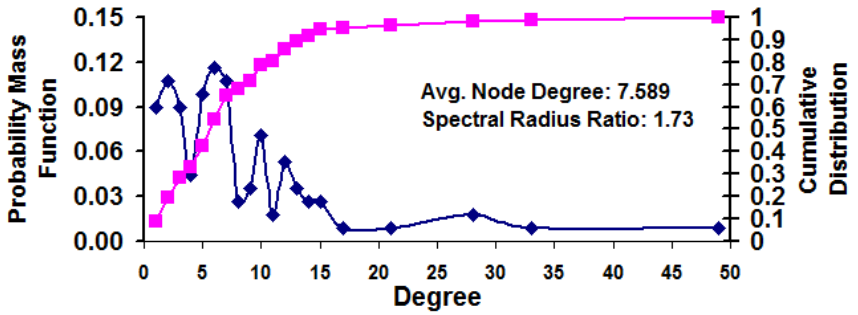


(c)

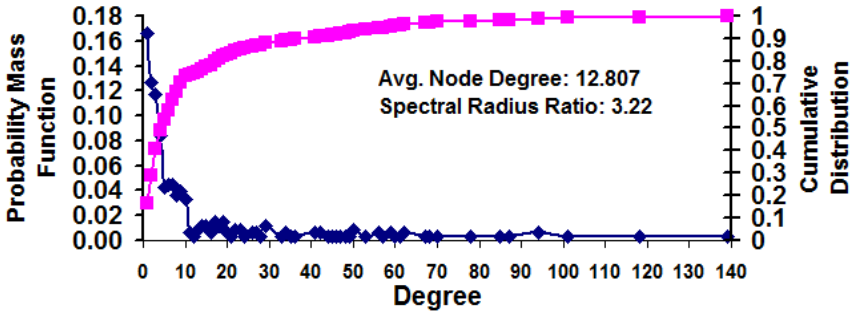
Figure 7 Distribution of node degrees in the real-world network graphs (probability mass function and cumulative distribution), (a) US College Football Network (115 nodes, 613 edges) (b) Dolphins' Social Network (62 nodes, 159 edges) (c) Politics Books Network (105 nodes, 441 edges) (d) Karate Club Network (34 nodes, 78 edges) (e) Word Adjacencies Network (112 nodes, 425 edges) (f) US Airports'97 Network (332 nodes, 2,126 edges) (continued) (see online version for colours)



(d)

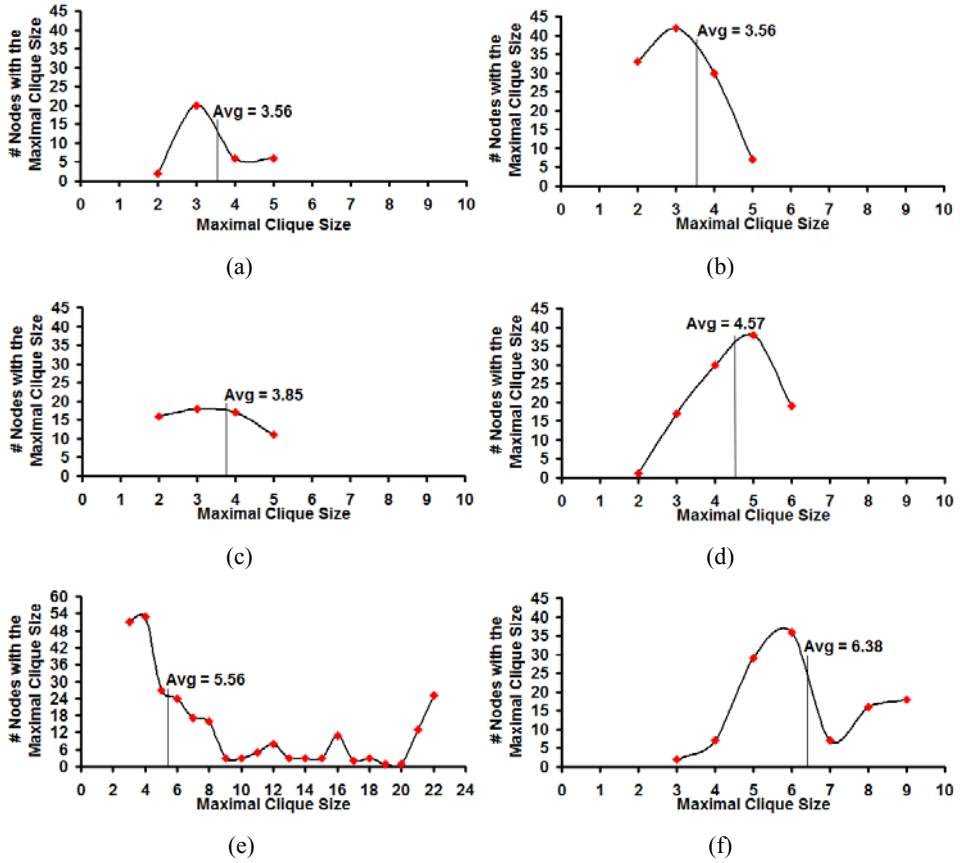


(e)



(f)

Figure 8 Distribution of maximal clique size of the vertices in the real-world network graphs, (a) karate club network (34 nodes, 78 edges) (b) Word Adjacencies Network (112 nodes, 425 edges) (c) dolphins' social network (62 nodes, 159 edges) (d) US Politics Books Network (105 nodes, 441 edges) (e) US Airports'97 Network (332 nodes, 2,126 edges) (f) US College Football Network (115 nodes, 613 edges) (see online version for colours)



5.2 Maximal clique size distribution of the real-world network graphs

Figure 8 presents the distribution of the maximal clique size of the vertices for the six real-world network graphs, in the increasing order of the average value for the maximal clique size of the vertices. An interesting observation is that five of the six real-world network graphs exhibit a Poisson-style distribution for the maximal clique size and the average value of the maximal clique size for the nodes is very close to the maximum value. The only real-world network that does not exhibit a Poisson-style distribution for the maximal clique size is the US Airports Network whose distribution of the maximal clique size appears to be more of a scale-free pattern with a long tail (wherein the average maximal clique size is 5.56, but there exists a significant number of nodes whose maximal clique size values are 21 and 22). We can also notice that the average value of the maximal clique size of the nodes is not proportional to the number of nodes in the

network nor to the spectral radius ratio for node degree. This is evident from three of the six real-world networks with comparable number of nodes (Word Adjacency Network – 112 nodes, US Politics Books Network – 112 nodes and the US Football Network – 105 nodes) incurring significantly different average values for the maximal clique size (3.56, 4.57 and 6.38, respectively). Similarly, though the spectral radius ratio for node degree increases with increase in the scale-free nature of the networks, we do not observe any such pattern of increase or decrease for the maximal clique size; for example: the US Football Network, Word Adjacency Network and the US Airports Network have spectral radius ratio for node degree values of 1.01, 1.73 and 3.22 respectively; whereas, their average maximal clique size values are 6.38, 3.56 and 5.56 respectively (no pattern of increase or decrease with increase in the spectral radius ratio for node degree).

Table 1 Assortativity index for maximal clique size and node degree

Network index	*Network name*	*Spectral radius ratio for node degree*	*Assortativity index for maximal clique size*	*Assortativity index for node degree*
(vi)	US Airports 1997 Network	3.22	0.17	–0.21
(iv)	Word Adjacencies Network	1.73	0.20	–0.09
(i)	Zachary's Karate Network	1.46	0.13	–0.48
(iii)	US Politics Books Network	1.41	0.20	–0.02
(ii)	Dolphins' Social Network	1.40	0.23	–0.04
(v)	US College Football Network	1.01	0.59	0.19

5.3 Assortativity index: maximal clique size and node degree

The assortativity index for a network graph with respect to a particular node-related metric is a measure of the association of nodes with similar values for the metric (Newman, 2010). For example, the assortativity index of a graph with respect to node degree is a measure of the association of higher degree nodes with other high degree nodes as well as the association of nodes of lower degree nodes with other lower degree nodes. In this section, we analyse the assortativity index of the six real-world network graphs with respect to the maximal clique size and node degree, and examine the nature of association between nodes having higher values for each of these two metrics. If m is the node-related metric of interest, then the assortativity index of the network graph with respect to m is evaluated as the correlation coefficient of the values (with respect to metric m) for the end nodes of the edges in the graph. Consider a network graph of n nodes and set of undirected (bi-directional) edges E; let $m[1]$, $m[2]$, …, $m[n]$ be the values for nodes 1, 2, …, n with respect to metric m and $\overline{m}$ be the average value of the metric, the assortativity index with respect to metric m is given by equation (1).

$$AssortativityIndex(m) = \frac{\sum_{(i,j)\in E} (m[i]-\bar{m})*(m[j]-\bar{m})}{\sqrt{\sum_{(i,j)\in E} (m[i]-\bar{m})^2}\sqrt{\sum_{(i,j)\in E} (m[j]-\bar{m})^2}} \quad (1)$$

Positive values for the assortativity index with respect to a metric indicates that the network exhibits assortativity with respect to the metric (nodes with higher values for the metric are more likely to be connected to nodes with higher values and vice-versa); negative values for the assortativity index indicates the network exhibits disassortativity (nodes with lower values for the metric are more likely to be connected to nodes with higher values for the metric and vice-versa); assortativity index values close to 0 indicates the network is more neutral with respect to the metric (i.e., no correlation between the values for the end nodes).

Table 1 lists the assortativity index values for the maximal clique size and degree of the vertices for the six real-world network graphs, along with their spectral radius ratio for node degree. We observe the assortativity index (with respect to the maximal clique size) for all the six network graphs to be positive and the assortativity index value for the maximal clique size increases with increase in the level of randomness in the network, indicating that the association of nodes of a particular maximal clique size with other nodes that are also of the same maximal clique size is more by chance. On the other hand, we observe the assortativity index (with respect to the node degree) for five of the six network graphs (i.e., all network graphs, except the US Football Network that exhibits the characteristic of random graphs) to be negative and the assortativity index values for the node degree gets more negative with increase in the scale-free nature of the network, indicating high degree nodes are more likely to be associated with low degree nodes (especially with increase in the spectral radius ratio for node degree).

6 Correlation coefficient analysis: centrality vs. maximal clique size

In this section, we present the results of correlation coefficient analysis conducted between the centrality values observed for the vertices vis-a-vis the maximal size clique that each vertex is part of. The analysis has been conducted on the six real-world network graphs (mentioned in Section 5) with respect to the four centrality metrics (degree-based DegC and EVC as well as the shortest-path based BWC and ClC metrics) and the maximal clique size measured for the vertices in these graphs. We implemented the algorithms to determine each of the four centrality metrics (see Section 3 for a description of these metrics and the algorithms/approach taken to measure each of these metrics) and the exact algorithm to determine the maximal clique size for each of the vertices in a graph (see Figure 6). The visualisation figures presented in the paper were obtained by porting the network data as well as the centrality/maximal clique size results to Gephi (Cherven, 2013) and making appropriate changes to the settings in the latter. The layout algorithm chosen in Gephi for visualisation is the Fruchterman Reingold algorithm (Fruchterman and Reingold, 1991) that presents the network in a circular format (like a globe).

Table 2 presents a correlation coefficient analysis of the four centrality metrics and the maximal clique size observed for the vertices in each of the six real-world network graphs studied in this paper. Values of correlation coefficient greater than or equal to 0.8 (high correlation) have been highlighted in yellow; values below 0.5 (low correlation) are highlighted in light blue; and values between 0.5 and 0.8 (moderate correlation) are not highlighted in any colour. If $\bar{X}$ and $\bar{Y}$ are the average values of the two metrics (say X and Y) observed for the vertices (IDs 1 to n, where n is the number of vertices) in the network, the formula used to compute the correlation coefficient between two metrics X and Y is given in equation (2), as follows:

$$CorrCoeff(X,Y) = \frac{\sum_{ID=1}^{n}(X[ID]-\bar{X})*(Y[ID]-\bar{Y})}{\sqrt{\sum_{ID=1}^{N}(X[ID]-\bar{X})^2}\sqrt{\sum_{ID=1}^{N}(Y[ID]-\bar{Y})^2}} \quad (2)$$

Table 2 Correlation between centrality metrics and maximal clique size for the nodes (see online version for colours)

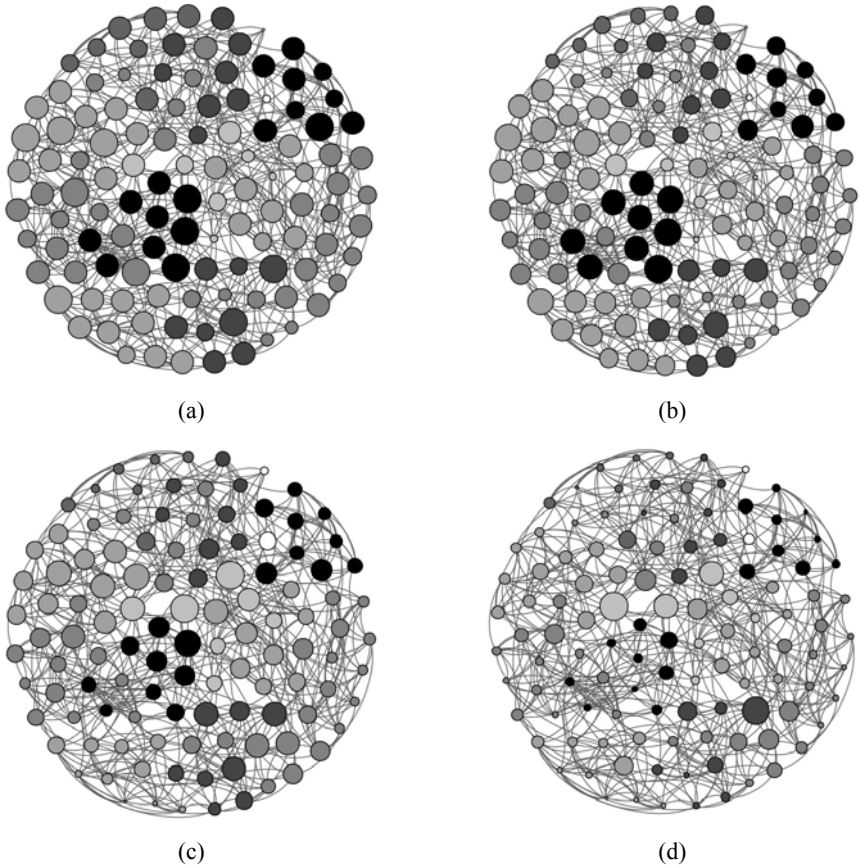
Network index	*Network name (increasing order of spectral radius ratio)*	*Degree vs. clique*	*Eigenvector vs. clique*	*Closeness vs. clique*	*Betweenness vs. clique*
(5)	US College Football Network	0.315	0.348	–0.028	–0.168
(2)	Dolphins' Social Network	0.776	0.563	0.418	0.277
(3)	US Politics Books Network	0.701	0.747	0.321	0.367
(1)	Zachary's Karate Network	0.641	0.767	0.615	0.458
(4)	Word Adjacencies Network	0.706	0.815	0.835	0.478
(6)	US Airports 1997 Network	0.868	0.953	0.843	0.404

As we can see in Table 2, in general, the correlation between the centrality metrics and the maximal clique size increases as the spectral radius ratio for node degree increases. This implies, the more scale-free a real-world network is, the higher the correlation between the centrality value and the maximal clique size observed for a node. With several of the real-world networks being mostly scale-free, we expect these networks to exhibit a similar correlation to that observed in this paper.

Figures 9 to 14 depict the correlation observed for the four centrality metrics with that of the maximal clique size for the vertices in the real-world network graphs. In these figures, the node size is a measure of the node centrality (the larger a node is, the larger is its centrality value); the node colour is a measure of the maximal size clique the vertex is part of (the darker a node is, the larger is the size of the maximal clique for the node). Among the two classes of centrality metrics, we observe the degree-based centrality metrics (DegC and EVC) to be very positively and highly correlated with the maximal clique size observed for the nodes. Between the two degree-based centrality metrics, the

EVC metric shows higher positive correlations to the maximal clique size. This could be attributed to the EVC of a node being a measure of both the degree of the node as well as the degrees of its neighbours. That is, a node with high degree as well as located in a neighbourhood of high degree vertices is more likely to be part of a maximal clique of larger size. In addition, as the networks get increasingly scale-free, nodes with high degree are more likely connected to other similar nodes with high degree (to facilitate an average path length that is almost independent of network size: characteristic of scale-free networks) contributing to a positive correlation between degree-based centrality metrics and maximal clique size.

Figure 9 US College Football Network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC



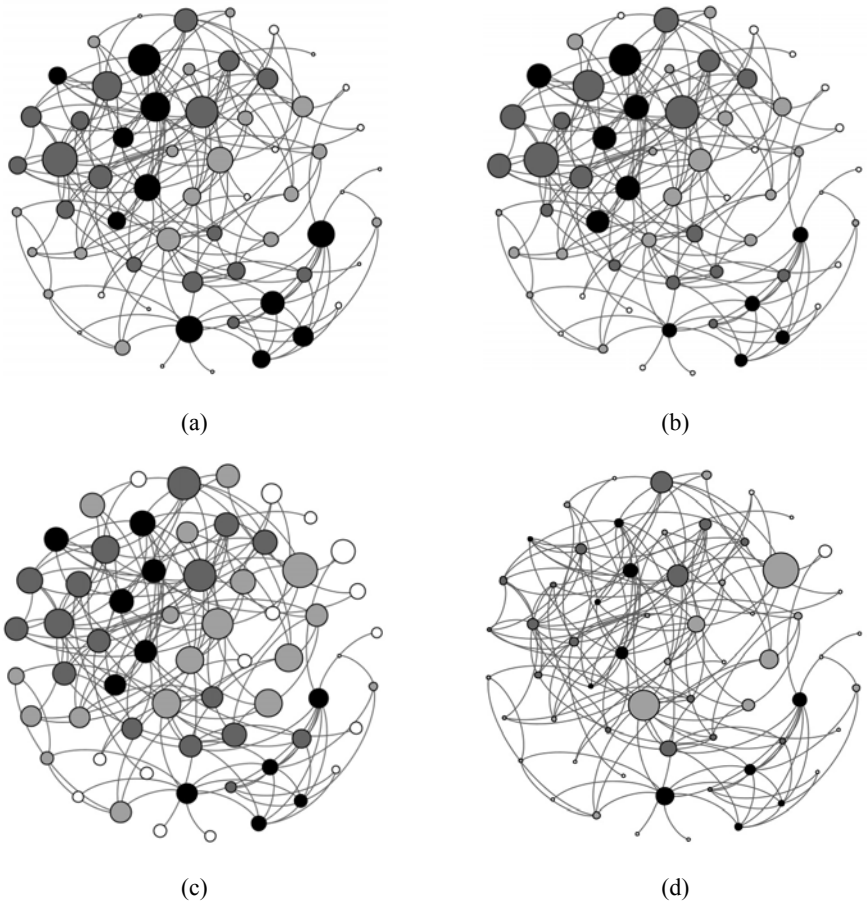
(a) (b)

(c) (d)

With respect to the two shortest-path based centrality metrics, the BWC metric is observed to exhibit a low correlation with maximal clique size for all the six real-world network graphs; the correlation coefficient increases as the network becomes increasingly scale-free. In networks with minimal variation in node degree (like the US College Football network that is more closer to a random network), nodes that facilitate shortest-path communication between several node pairs in the network are not part of a larger size clique; on the other hand, nodes that are part of larger size cliques in such

random networks exhibit a relatively lower BWC. Since the degrees of the vertices in random networks are quite comparable to the average node degree, there is no clear ranking of the vertices based on the degree-based centrality metrics and maximal size cliques that they are part of. Also, if at all a vertex ends up being a larger sized clique in random network graphs, it does not facilitate shortest path communication between the majority of the vertices (contributing to a negative/zero correlation or at best a low correlation with BWC). As the network becomes increasingly scale-free, the hubs that facilitate shortest-path communication between any two nodes in the network exhibit higher betweenness and closeness centralities as well as form a clique with other high-degree hubs – exhibiting the ultra small-world property; the average path length is ln(ln N), where N is the number of nodes in the network (Newman, 2010). The correlation of the ClC values and the maximal clique size values observed for the vertices in real-world network graphs is significantly higher (i.e., positive correlation) for networks that are increasingly scale-free.

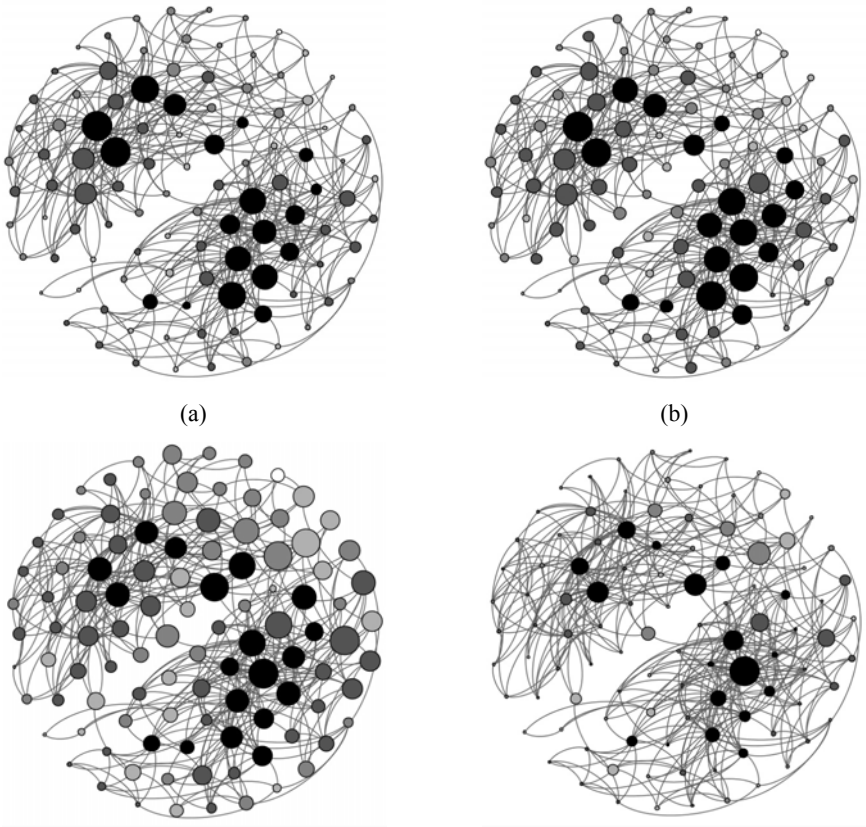
Figure 10 Dolphins' social network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC



(a) (b)

(c) (d)

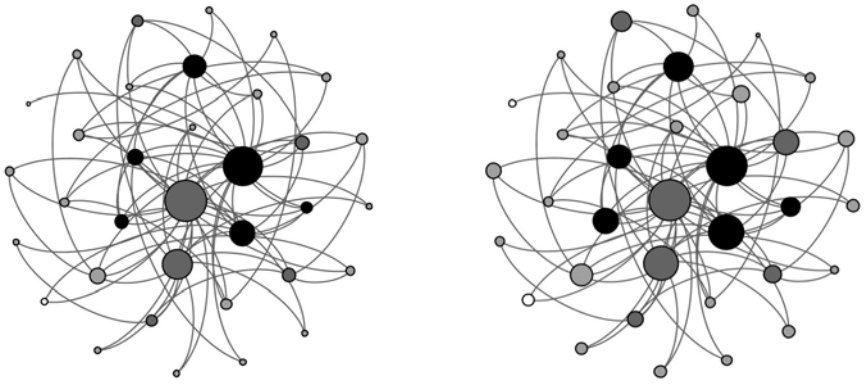
Figure 11 US Politics Books Network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC



(a) (b)

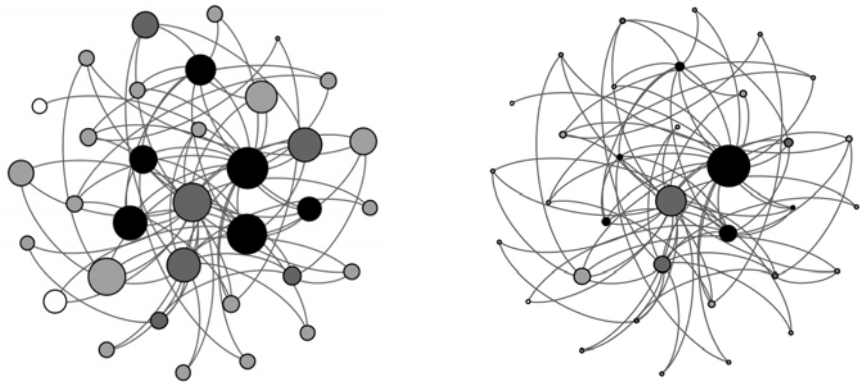
(c) (d)

Figure 12 Zachary's karate club network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC



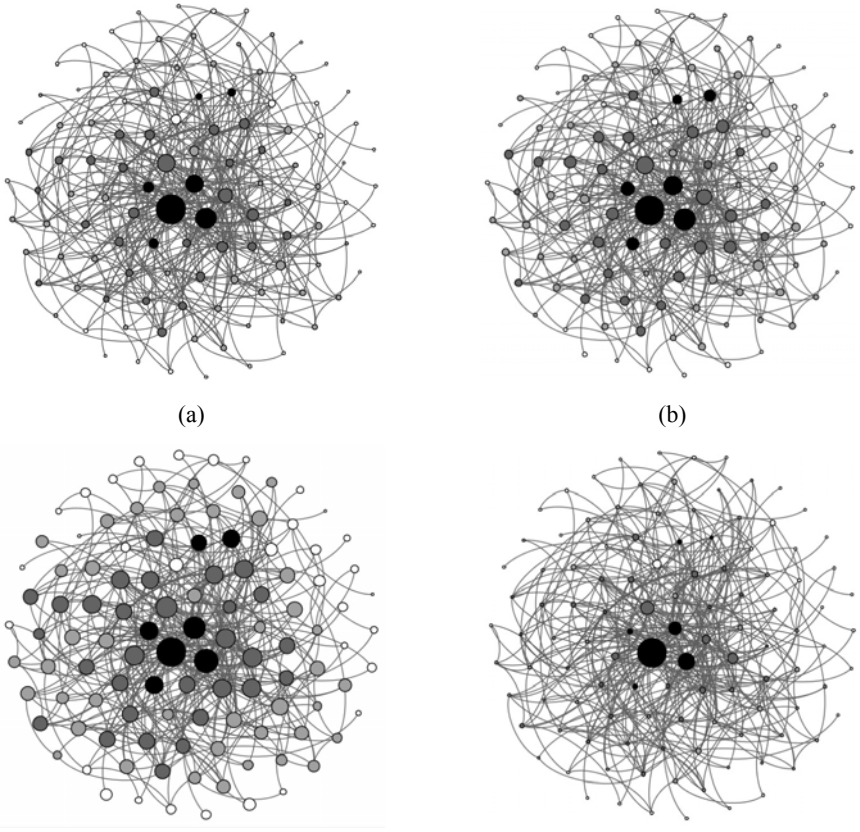
(a) (b)

Figure 12 Zachary's karate club network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC (continued)



(c) (d)

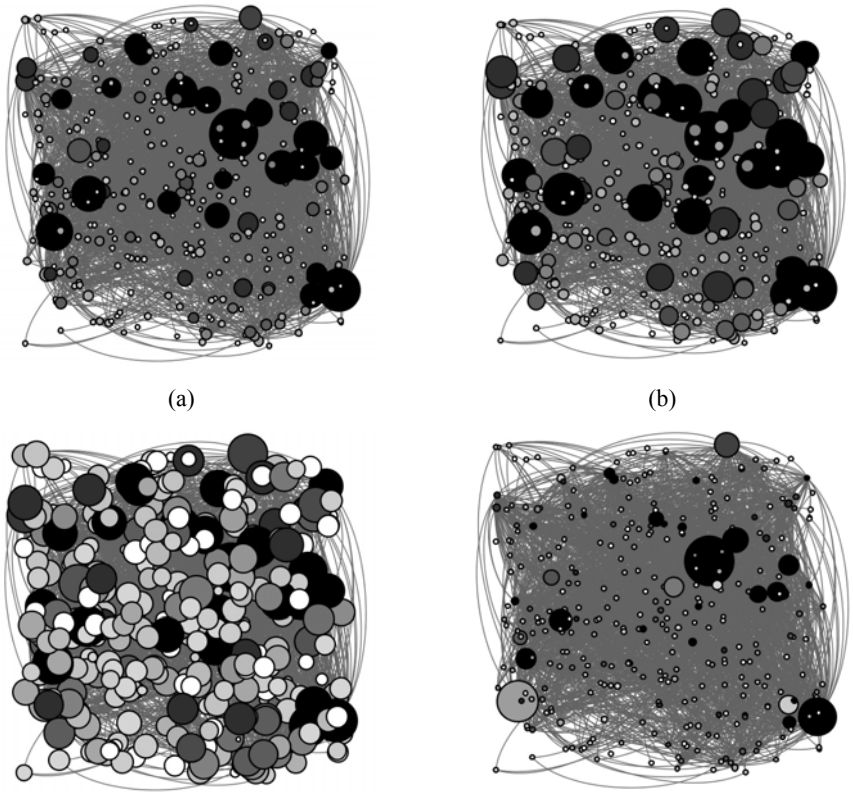
Figure 13 Word Adjacencies Network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC



(a) (b)

(c) (d)

Figure 14 US Airports (1997) network: correlation of maximal clique size with centrality metrics, (a) DegC (b) EVC (c) ClC (d) BWC



(a) (b)

(c) (d)

Overall, the degree-based centrality metrics exhibit a relatively better correlation with the maximal clique size compared to that of the shortest-path based centrality metrics (especially in networks with low-moderate variation in node degree). For real-world networks that exhibit moderate-high variation in node degree, the shortest-path based centrality metrics (especially ClC) fast catch up with that of the degree-based centrality metrics and exhibit higher levels of positive correlation with maximal clique size. We anticipate that as the networks become increasingly scale-free, the hubs (that facilitate shortest-path communication between any two nodes) are more likely to form the maximum clique for the entire network graph – contributing to higher levels of positive correlation between any of the four centrality metrics and maximal clique size.

7 Conclusions

The correlation coefficient analysis studies between the four centrality metrics (degree, eigenvector, betweenness and closeness centralities) and the maximal clique size for the vertices in the real-world network graphs illustrate several significant findings that have been so far not reported in the literature:

1 the degree-based centrality metrics (especially the EVC) exhibit a significantly high positive correlation to the maximal clique size as the networks get increasingly scale-free

2 the BWC of the vertices exhibits a low correlation with that of the maximal size cliques the vertices can be part of

3 in real-world networks that are close to random network graphs, the centrality metrics exhibit a low correlation to maximal clique size (especially in the case of shortest-path based closeness and BWC metrics)

4 for all the four centrality metrics, the extent of positive correlation with maximal clique size increases as the real-world networks become increasingly scale-free.

With the problem of determining maximal clique sizes for individual vertices being computationally time consuming, our approach taken in this paper to study the correlation between maximal clique sizes to centrality can be the first step in identifying positive correlation between cliques/clique size in real-world network graphs to one or more network metrics (like centrality) that can be quickly determined and henceforth, appropriate inferences can be made about the maximal size cliques of the individual vertices. We observe the degree-based centrality metrics (especially the EVC) to show promising positive correlations to that of maximal clique sizes of the individual vertices, especially as the networks get increasingly scale-free; this observation could form the basis of future research for centrality-clique analysis for complex real-world networks.

References

Brandes, U. (2001) 'A faster algorithm for betweenness centrality', *Journal of Mathematical Sociology*, Vol. 25, No. 2, pp.163–177.

Carraghan, R. and Pardalos, P. (1990) 'An exact algorithm for the maximum clique problem', *Operations Research Letters*, Vol. 9, No. 6, pp.375–382.

Cherven, K. (2013) *Network Graph Analysis and Visualization with Gephi*, 1st ed., Packt Publishing, Birmingham, UK.

Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C. (2009) *Introduction to Algorithms*, 3rd ed., MIT Press, Cambridge, MA, USA.

Fortunato, S. (2010) 'Community detection in graphs', *Physics Reports*, Vol. 486, Nos. 3–5, pp.75–174.

Fruchterman, T.M.J. and Reingold, E.M. (1991) 'Graph drawing by force-directed placement', *Software: Practice and Experience*, Vol. 21, No. 11, pp.1129–1164.

Girvan, M. and Newman, M. (2002) 'Community structure in social and biological networks', *Journal of the National Academy of Sciences of the United States of America*, Vol. 19, No. 12, pp.7821–7826.

Koschutzki, D. and Schreiber, F. (2008) 'Centrality analysis methods for biological networks and their application to gene regulatory networks', *Gene Regulation and Systems Biology*, Vol. 2, No. 1, pp.193–201.

Krebs, V. (2008) [online] http://www.orgnet.com/divided.html (accessed 27 May 2015).

Li, C., Li, Q., Van Mieghem, P., Stanley, H.E. and Wang, H. (2014) 'Correlation between centrality metrics and their application to the opinion model', *Physics and Society*, arXiv: 1409.6033.

Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E. and Dawson, S.M. (2003) 'The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations', *Behavioral Ecology and Sociobiology*, Vol. 54, No. 4, pp.396–405.

Newman, M. (2006) 'Finding community structure in networks using the eigenvectors of matrices', *Physics Review*, Vol. E 74, p.036104.

Newman, M. (2010) *Networks: An Introduction*, 1st ed., Oxford University Press, Oxford, UK.

Opsahl, T., Agneessens, F. and Skvoretz, J. (2010) 'Node centrality in weighted networks: generalizing degree and shortest paths', *Social Networks*, Vol. 32, No. 3, pp.245–251.

Ostergard, P.R.J. (2002) 'A fast algorithm for the maximum clique problem', *Discrete Applied Mathematics*, Vol. 120, Nos. 1–3, pp.197–207.

Palla, G., Derenyi, I., Farkas, I. and Vicsek, T. (2005) 'Uncovering the overlapping community structure of complex networks in nature and society', *Nature*, Vol. 435, No. 1, pp.814–818.

Pattabiraman, B., Patwary, M.A., Gebremedhin, A.H., Liao, W-K. and Choudhary, A. (2013) 'Fast problems for the maximum clique problem on massive sparse graphs', *Proceedings of the 10th International Workshop on Algorithms and Models for the Web Graph: Lecture Notes in Computer Science*, Vol. 8305, pp.156–169.

Rossi, R.A., Gleich, D.F. and Patwary, M.A. (2014) 'Fast maximum clique algorithms for large graphs', *Proceedings of the 23rd International Conference on World Wide Web Companion*, pp.365–366.

Sadi, S., Oguducu, S. and Uyar, A.S. (2010) 'An efficient community detection method using parallel clique-finding ants,' *Proceedings of IEEE Congress on Evolutionary Computation*, pp.1–7.

Strang, G. (2005) *Linear Algebra and its Applications*, 4th ed., Cengage Learning, Boston, MA, USA.

Tomita, E. and Seki, T. (2003) 'An efficient branch-and-bound algorithm for finding a maximum clique', *Proceedings of the 4th International Conference on Discrete Mathematics and Theoretical Computer Science*, pp.278–289.

Zachary, W.W. (1977) 'An information flow model for conflict and fission in small groups', *Journal of Anthropological Research*, Vol. 33, No. 4, pp.452–473.