Mathematics of Cellular Networks

First Assignment

Prof. Dr. Volkhard Helms
Tutor: Maryam Nazarieh

April 22, 2014

Exercise 1.1: Closeness Centrality (25 points)

Consider an undirected tree of n vertices. A particular edge in the tree joins vertices 1 and 2 and divides the tree into two disjoint regions of n_1 and n_2 vertices as sketched below:
show that closeness centralities C_1 and C_2 of the two vertices, defined according equation $C_i =$

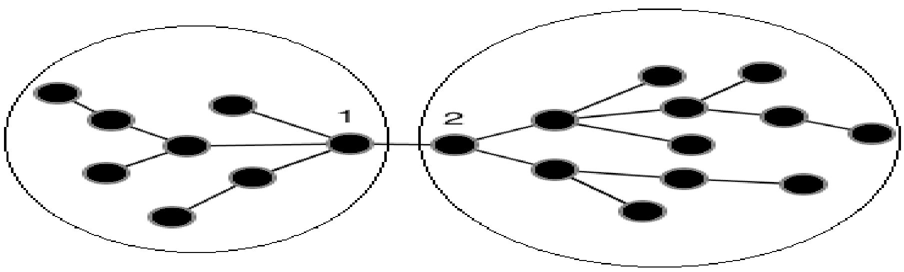

Figure 1: Partitionaning based on closeness centrality

$\frac{1}{l_i} = \frac{n}{\sum_j d_{ij}}$, are related by $\frac{1}{C_1} + \frac{n_1}{n} = \frac{1}{C_2} + \frac{n_2}{n}$

Exercise 1.2: Comparison of Centrality Measures (50 points)

Write a program to apply the three centrality measures listed below to report a table with the 5 vertices with the highest centrality in the toy network shown in Figure 2. (35 points)

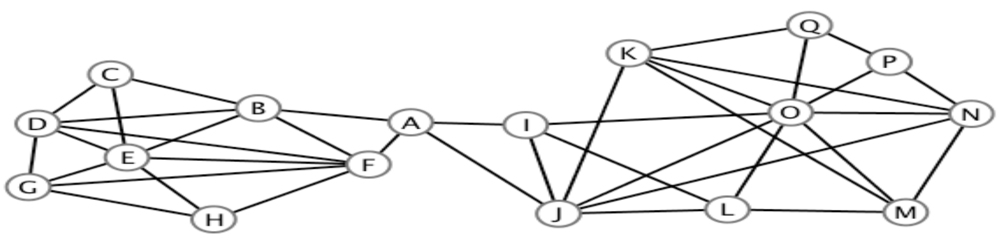

Figure 2: Comparison of different centrality measures

- degree centrality:
- eigenvector centrality:

- closeness centrality:

Discuss the difference of the results and explain this by the properties of the individual centrality measures. (15 points)

Exercise 1.3: Partitionening based on Edge Betweenness (25 points)

Betweenness centrality has been used in a variety of fields to infer important information about the network. One of its applications is to find disjoint or overlapping communites by removing high betweenness edges. The Girvan-Newman algorithm extends the definition of vertex betweenness by defining edge betweenness. The edge betweenness of an edge is equal to sum over all pairs of vertices (u,w), of the fraction of shortest paths between u and w that passes through e. $BC(e) = \sum_{\substack{u,w\in V \\ u\neq w}} \frac{\sigma_{uw}(e)}{\sigma_{uw}}$ Apply the mentioned algorithm to answer the following questions for the figure shown below: How many steps are required to have three disjoint components? (You may solve

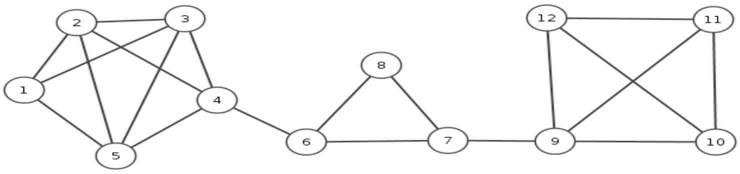

Figure 3: Partitioning based on edge betweenness

this problem either with a short program or by hand.)

(a) find the edge betweenness for all the edges in the network. List them in a table. (10 points)

(b) remove the edge with the highest betweennness value from the network.

(c) recalculate the edge betweenness values for all the edges in the remaining network.

(d) return to step 2 until the graph has no edges. (15 points)

Submission:
Send a single PDF attachment with your name(s), your solution(s) and a printout of your program(s) for exercises 1.2 and 1.3 via mail to nazarieh@mpi-inf.mpg.de until May 6, 2014 10:00 AM.