Information Society
Brassersplein 2
2612 CT Delft
P.O. Box 5050
2600 GB Delft
The Netherlands

www.tno.nl

T +31 88 866 70 00
F +31 88 866 70 57
infodesk@tno.nl

TNO report 35526

Robustness of networks with respect to node removal

Date	30 August 2011
Author(s)	Rashnie Chhatta
Reviewer	Rob Kooij
Number of pages	33 (incl. appendices)
Number of appendices	2
Project name	Robustness of networks with respect to node removal

Contents

Summary

The robustness of complex networks, like metro networks, is a really important issue to our daily life. With robustness we indicate the extent to which a network can deal with challenges imposed on the network, such as failures or attacks of nodes and/or links of the network. In this case the stations of the metro networks are the nodes of the complex network and the rail tracks are the links. The robustness of these networks will be related to the expected relative size of the largest connected component as a function of node failure probability of the network for the specific case of random nodes removal, which is presented in plots. The plots show that the smaller metro networks are more robust than the larger ones, at least according to our definition of robustness. The smallest network is the network of Rome with 5 nodes and the largest one is London with 83 nodes. Also the numbers of iterations which are needed to calculate the largest connected component and the impact of two networks, with same number of nodes and various numbers of links, on the expected relative size have been discussed in this report.

1 Signature

Delft, 30 august 2011

Head of department Author

2 Introduction

Our daily activities rely increasingly on complex networks. The power grid, the Internet, and transportation networks are all examples of complex networks. Because these networks are so important to our daily life, the robustness of complex networks is a very important issue. With robustness we indicate the extent to which a network can deal with challenges imposed on the network, such as failures or attacks of nodes and/or links of the network. A network is modeled as a finite, undirected graph $G=(V,E)$ in which each node $v \in V$ represents a site of the network, and each link $e \in E$ represents a bidirectional link between two sites. Nodes are prone to failure, each node fails independently with the same failure probability q:

$$q = 1 - p \tag{1}$$

where p denotes the availability of each node in the network. In this report the focus will be on a specific class of transportation networks, namely the metro networks of 33 cities in the world. The data of these networks comes from a prior study conducted at the University of Toronto [1]. The robustness of these networks will be related to the expected relative size of the largest connected component:

$$E_G[q] = \sum_{k=0}^{n} LCC_k(G).(1-q)^{n-k}.q^k \tag{2}$$

where $LCC_k(G)$ denotes the expected relative size of largest connected component of graph G after k random node removals. To be more precise

$$LCC_k = \frac{E[n_k]}{n} \tag{3}$$

where $E[n_k]$ denotes the expected size of the largest connected component, after random removal of k nodes, see [2]. The nodes in the graph will be removed until no nodes remain. The largest connected component of a graph is the largest subgraph such for any two of its nodes, say i and j, node j can be reached from node i using a path formed by edges internal to the subgraph [3].The purpose of this project is to determine the expected relative size of the largest connected component as a function of the failure probability of the network for the specific case of random nodes removal.

3 Expected value of the largest connected component

To calculate the *LCC* (eq. 3) of the metro networks, the nodes are removed from these networks until no nodes remain. This process is repeated for several times and the results are averaged in order to get consistent data and afterwards the *LCC* is used to calculate $E_{G1}[q]$ (eq. 2). The algorithms were written in MATLAB (appendix B) to compute the average largest connected component and the expected value of the largest connected component for every metro network. To guarantee exact results, first this process is done analytically and compared with the simulations.

3.1 Number of iterations and the confidence interval

To decide how many iterations are needed to get consistent results, the metro network Rome (Figure 3) is used, which has 5 nodes and 4 links (star network): *n=5* and *m=4*.
The analytical results are obtained from the next formula:

$$E_{G1}(q)=\frac{5}{5}(1-q)^5+\frac{\left(\frac{(4\times4)+(1\times1)}{5}\right)}{5}(1-q)^4 q+\frac{\left(\frac{(6\times3)+(4\times1)}{10}\right)}{5}(1-q)^3 q^2+\frac{\left(\frac{(4\times2)+(6\times1)}{10}\right)}{5}(1-q)^2 q^3+\frac{1}{5}(1-q)q^4+\frac{0}{5}q^5$$

This results in:

$$E_{G1}(q)=1(1-q)^5+0.68(1-q)^4 q+0.44(1-q)^3 q^2+0.28(1-q)^2 q^3+0.2(1-q)q^4$$

The numbers in red are the relative size of the largest connected components of the graph after random node removal. These results are also shown in Table 1, column denoted by A (Analytical results). Also the results after 100, 1000, 10000 and 100000 iterations are presented in the table. To decide how many iterations are sufficient, the confidence interval for each of the cases ($LCC_0(G_1)$ ... $LCC_5(G_1)$) is calculated.

	A	**100**	**1,000**	**10,000**	**100,000**
$LCC_0(G_1)$	1	1	1	1	1
$LCC_1(G_1)$	0.68	0.686	0.6764	0.6824	0.6788
$LCC_2(G_1)$	0.52	0.448	0.4444	0.4424	0.4385
$LCC_3(G_1)$	0.28	0.282	0.2816	0.2807	0.2795
$LCC_4(G_1)$	0.2	0.2	0.2	0.2	0.2
$LCC_5(G_1)$	0	0	0	0	0

Table 1: Analytic and simulation results for Rome

A confidence interval (CI) is a particular kind of interval estimate of a population parameter and is used to indicate the reliability of an estimate [6]. With a probability of 0.95 the confidence interval is:

$$\overline{X} \pm 1.96 \frac{\sigma}{\sqrt{N}} \tag{4}$$

where the calculation to get 1.96 has been explained in [6] and X is equal to LCC_k (eq. 3). $\sigma/\sqrt{N}$ is the standard error and σ is the standard deviation which can be calculated by:

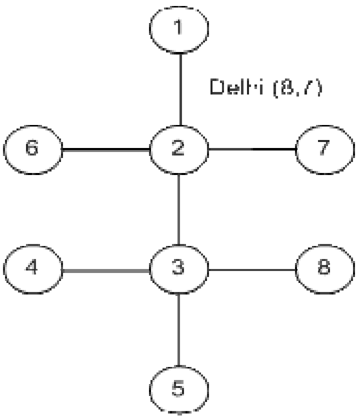
$$\sigma = \sqrt{\frac{\left(X - \overline{X}\right)^2}{N}} \tag{5}$$

where N denotes the number of samples values. With these equations the confidence intervals for the metro network Rome is calculated to decide how many iterations are needed for the simulations. The results are shown in Table 2. For the cases with one possibility, after removing a node, the confidence intervals are not calculated as there is only one possibility.

k	σ	$\overline{X} - 1.96\frac{\sigma}{\sqrt{N}}$	$\overline{X} = LCC$ **A**	$\overline{X} + 1.96\frac{\sigma}{\sqrt{N}}$	$\overline{X} = LCC$ **S**
0	-	-	1	-	1
1	0.1073	0.47	0.68	0.89	0.6824
2	0.03	0.4593	0.52	0.5807	0.4424
3	0.03	0.2193	0.28	0.3407	0.2807
4	-	-	0.2	-	0.2
5	-	-	0	-	0

Table 2: Confidence intervals for LCC_k for Rome, N = 10,000

Table 2 shows that, at least for the Rome network, running 10,000 simulations gives data consistent with the analytical results. In fact, Table 1 indicates that even running just 100 iterations leads to accurate results. We also tested the simulations on two other metro networks for which the analytic results are tractable: Delhi with 8 nodes and 7 links, and Cairo, with 6 nodes and 5 links. The graphs of these networks are given in Figures 1 and 2, respectively. Also in these cases the analytic results are within the confidence intervals. The results for Cairo and Delhi are given in Tables 3 and 4, respectively. Again A is denotes for the analytic results and while S stands for the simulation results (with N = 10,000).



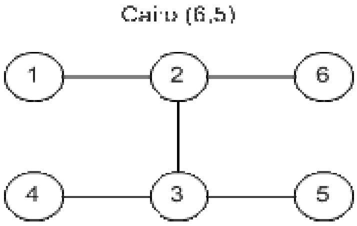


Figure 1: Metro network of Delhi

Figure 2: Metro network of Cairo

Cairo (6,5): Analytic

$$
\begin{aligned}
E_G(q) &= \frac{6}{6}(1-q)^6 + \frac{\left(\frac{(4\times5)+(2\times3)}{6}\right)}{6}(1-q)^5 q \\
&+ \frac{\left(\frac{(6\times4)+(4\times3)+(4\times2)+(1\times1)}{15}\right)}{6}(1-q)^4 q^2 \\
&+ \frac{\left(\frac{(5\times3)+(8\times2)+(7\times1)}{20}\right)}{6}(1-q)^3 q^3 + \frac{\left(\frac{(5\times2)+(10\times1)}{15}\right)}{6}(1-q)^2 q^4 \\
&+ \frac{1}{6}(1-q)q^5 + \frac{0}{6}q^6
\end{aligned}
$$

Delhi (8,7): Analytic

$$
\begin{aligned}
E_G(q) &= \frac{8}{8}(1-q)^8 + \frac{\left(\frac{(2\times4)+(6\times8)}{8}\right)}{8}(1-q)^7 q \\
&+ \frac{\left(\frac{(1\times1)+(6\times3)+(6\times4)+(15\times6)}{28}\right)}{8}(1-q)^6 q^2 \\
&+ \frac{\left(\frac{(6\times1)+(6\times2)+(18\times3)+(6\times4)+(20\times5)}{56}\right)}{8}(1-q)^5 q^3 \\
&+ \frac{\left(\frac{(17\times1)+(18\times2)+(18\times3)+(17\times4)}{70}\right)}{8}(1-q)^4 q^4 \\
&+ \frac{\left(\frac{(26\times1)+(18\times2)+(12\times3)}{56}\right)}{8}(1-q)^3 q^5 + \frac{\left(\frac{(21\times1)+(7\times2)}{28}\right)}{8}(1-q)^2 q^6 \\
&+ \frac{1}{8}(1-q)q^7 + \frac{0}{8}q^8
\end{aligned}
$$

k	**σ**	$\overline{X}-1.96\frac{\sigma}{\sqrt{N}}$	$\overline{X}=LCC$ **A**	$\overline{X}+1.96\frac{\sigma}{\sqrt{N}}$	$\overline{X}=LCC$ **S**
0	-	-	1	-	1
1	0.1571	0.5965	0.7222	0.8479	0.72
2	0.1610	0.4185	0.50	0.5815	0.4988
3	0.1724	0.2412	0.3167	0.3922	0.334
4	0.0786	0.1825	0.2222	0.2619	0.223
5	-	-	0.1667	-	0.1667
6	-	-	0	-	0

Table 3: Confidence intervals for LCC_k for Cairo, N = 10,000

k	**σ**	$\overline{X}-1.96\frac{\sigma}{\sqrt{N}}$	$\overline{X}=LCC$ **A**	$\overline{X}+1.96\frac{\sigma}{\sqrt{N}}$	$\overline{X}=LCC$ **S**
0	-	-	1	-	1
1	0.1624	0.66875	0.78125	0.89375	0.7822
2	0.1818	0.52645	0.59375	0.66105	0.5919
3	0.1404	0.4007	0.4375	0.4743	0.4344
4	0.1380	0.2801	0.3125	0.3449	0.3177
5	0.0981	0.19305	0.21875	0.24445	0.2189
6	0.05	0.13628	0.15628	0.17628	0.1568
7	-	-	0.125	-	0.125
8	-	-	0	-	0

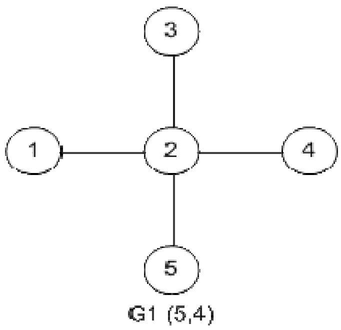
Table 4: Confidence intervals for LCC_k for Delhi, N = 10,000

q	$E_{Cairo}(q)$ **A**	$E_{Cairo}(q)$ **S**	$E_{Delhi}(q)$ **A**	$E_{Delhi}(q)$ **S**
0	1	1	1	1
0.1	0.5776	0.5777	0.4713	0.4713
0.2	0.3192	0.3194	0.2082	0.2082
0.3	0.169	0.1692	0.0861	0.08612
0.4	0.0869	0.08725	0.034	0.03406
0.5	0.0457	0.04605	0.0142	0.01419
0.6	0.0273	0.02756	0.008	0.007979
0.7	0.0201	0.02024	0.0071	0.00712
0.8	0.0166	0.01669	0.0078	0.007775
0.9	0.0116	0.01159	0.007	0.006966
1	0	0	0	0

Table 5: The expected relative size of LCC for Cairo and Delhi.

Table 5 shows analytical and simulation results for the expected relative size of LCC for Cairo and Delhi, with *q* in the interval [0,1].It is clear that the analytical as well the simulation results are almost the same. Therefore we will use 10,000 iterations for the rest of the simulations of the metro networks in the remainder of this report.

3.2 The impact of 2 networks, with same number of nodes and various numbers of links, on $E_G[q]$.



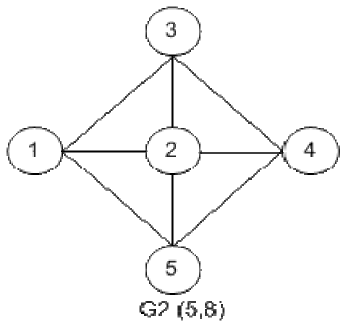


Figure 3: Rome network

Figure 4: Network with more links than Rome

In this example we look at the differences in *LCC* and $E_G[q]$ for two network topologies G_1 (Rome) and G_2, with equal number of nodes but different number of links, e.g. $n_1=n_2=5$, $m_1=4$ and $m_2=8$. The graphs are given in Figure 3 and 4 respectively. In the previous subsection, the results for G_1 are already calculated; results for G_2 are given in Table 6:

S:

$$E_{G1}(q)=1(1-q)^5+0.80(1-q)^4q+0.60(1-q)^3q^2+0.3604(1-q)^2q^3+0.2(1-q)q^4$$

q	$E_{G1}(q)$	$E_{G2}(q)$
0	1	1
0.1	0.6387	0.6477
0.2	0.3942	0.4076
0.3	0.2356	0.2501
0.4	0.1378	0.1513
0.5	0.08127	0.09249
0.6	0.05085	0.05915
0.7	0.03513	0.04042
0.8	0.02554	0.02817
0.9	0.01558	0.01631
1	0	0

Table 6: Expected relative size of LCC for Rome and network in Figure 4

The differences in *LCC* for both of the networks are clear and it is obvious that the larger the number of links the larger the largest connected component of the network. But the differences for $E_G[q]$ are not that much as it is shown in Figure 5, because after a node removal also the number of links connected with that specific node will also be removed.

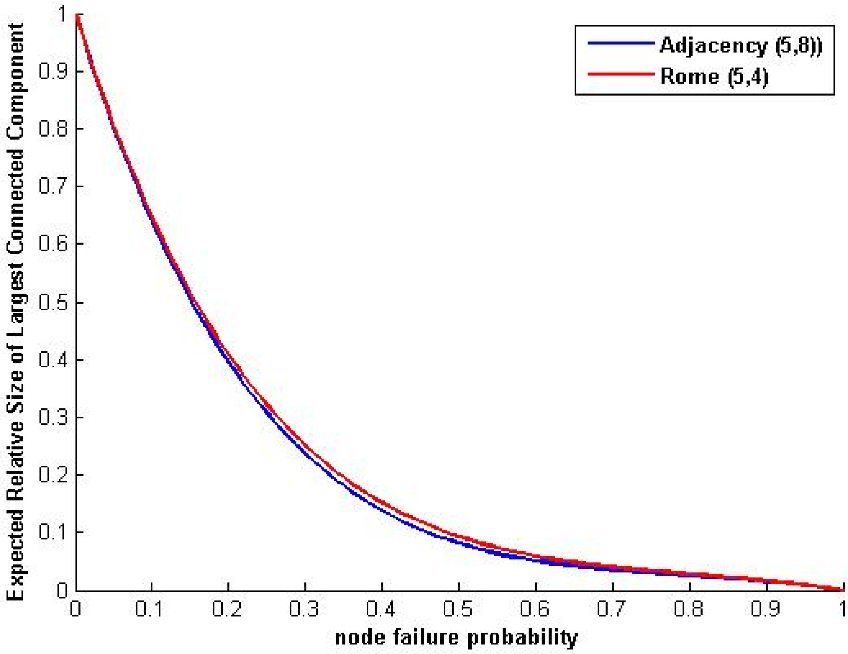


Figure 5: Expected relative size of LCC for Rome and network in Figure 4

4 Results

The resulting plots can be found in Appendix A, they show the expected relative size of the largest connected component as a function of the failure probability of the nodes. From the plots can be concluded that the network Rome, with 5 nodes and 4 links, is the most robust network followed by Cairo, Marseille and Delhi. New York, Paris and London perform the worst. To give a better overview how worst the largest networks perform, the maximum q value for those networks have been reduced from 1 to 0.3 and 0.1. Networks with nodes larger than 20 has a q between 0 and 0.3, while networks with nodes larger than 50 has q value between 0 and 0.1. This means that the smaller metro networks are more robust than the larger ones, at least according to our definition of robustness. As there are several methods which can be related to the robustness of networks, for further research the robustness of the metro networks or any other network can be related to the expected relative size of LCC, but than in case of random link removal instead of random node removal. Also the case of degree based node removal can be analyzed.

5 References

[1] S. Derrible and K. Kennedy, "*The Complexity and robustness of metro networks*", Physica A: Statistical Mechanics and its Applications Volume 389, Issue 17, 1 September 2010, Pages 3678-3691.

[2] W. Ghamry and K. Elsayed, "*Network Design Methods for Mitigation of Intentional Attacks in Scale-free Networks*", Cairo University, Augustus 2010.

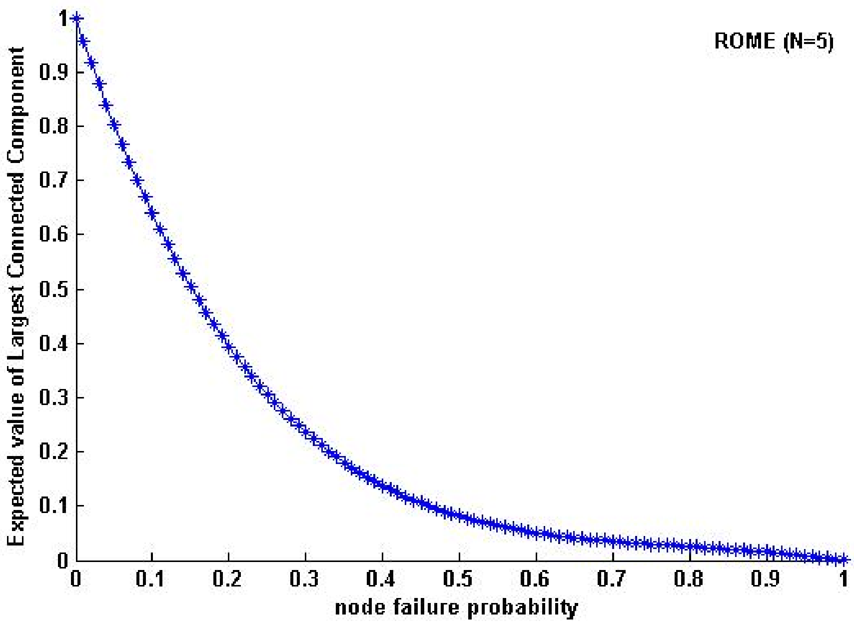
[3] J. Domingo-Ferrer and U. González-Nicolás, "*Decapitation of Networks with and without Weights and Direction: The Economics of Iterated Attack and Defense*", Universitat Rovira i Virgili, Tarragona, Spain, 2009.

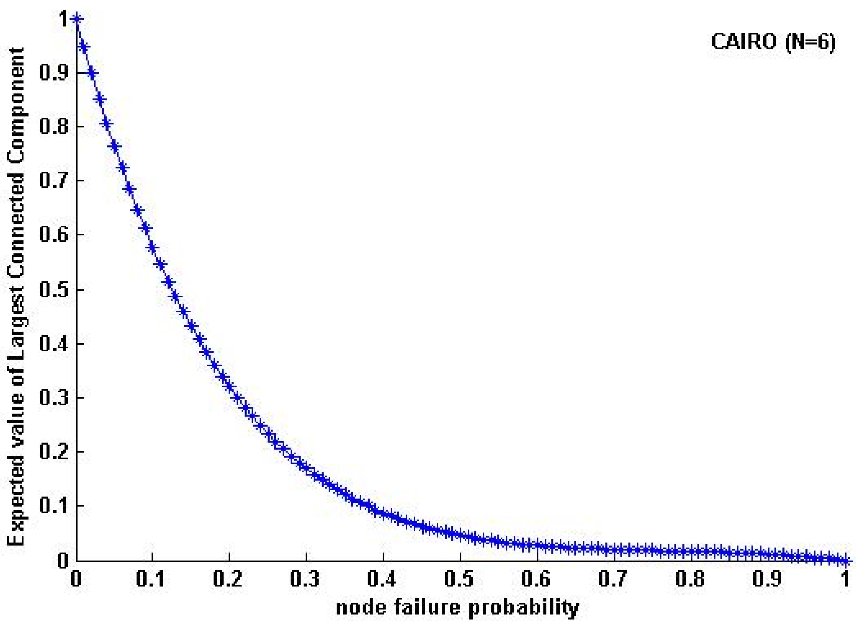
[4] F. Knorn, "*Ranking and importance in complex networks*", Otto–von–Guericke–Universität Magdeburg, Duitsland, October 2003.

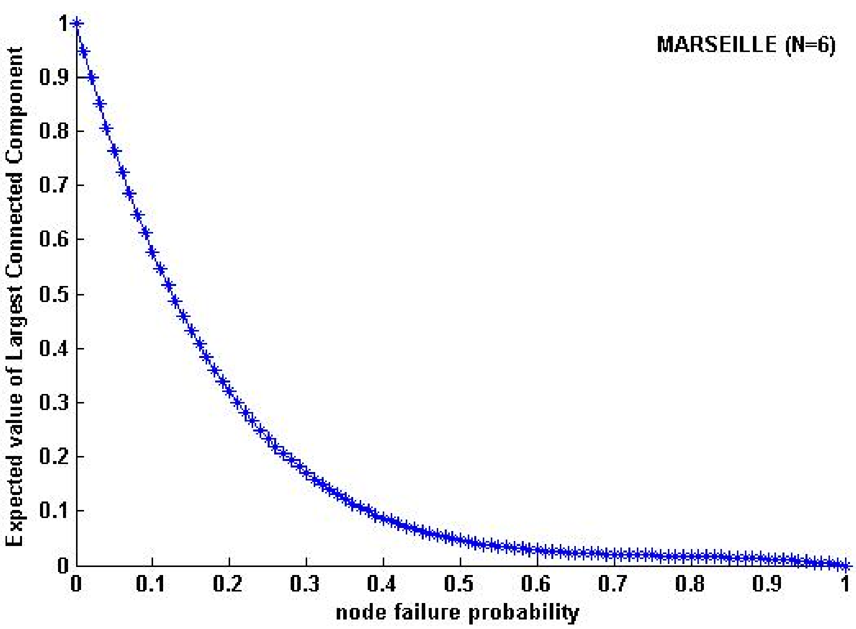
[5] Y. Koç, K. Kotobi and K Lyngbæk, "*Robustness of Metro Networks*", assignment for ET4374 Complex Networks, Technische Universiteit van Delft, Nederland, April 2011.

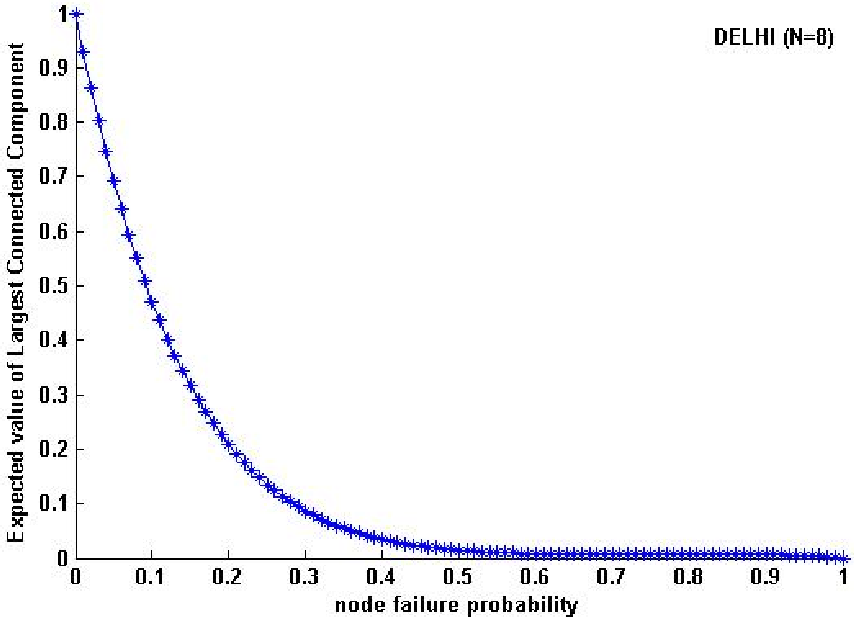
[6] http://en.wikipedia.org/wiki/Confidence_interval, June 2011.

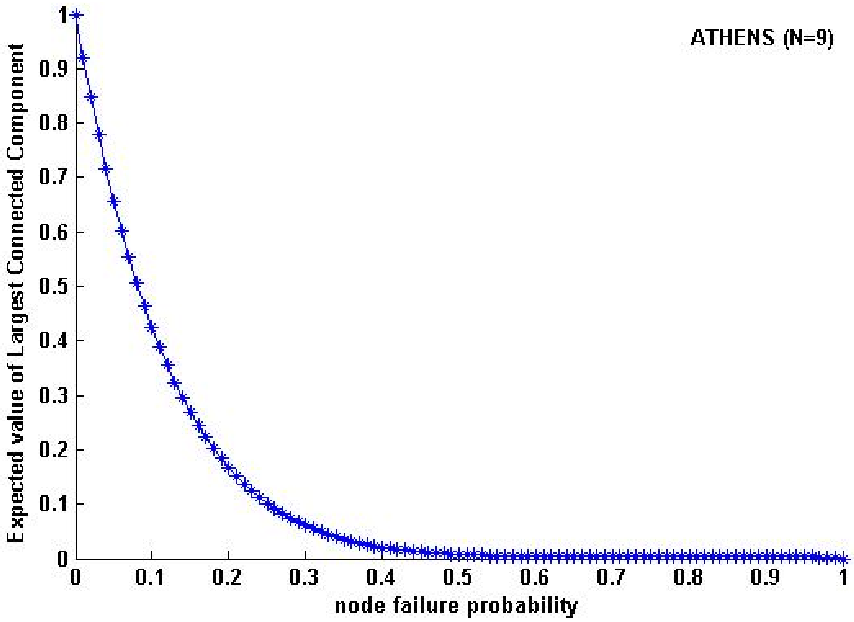
Appendix A: Plots for all metro networks

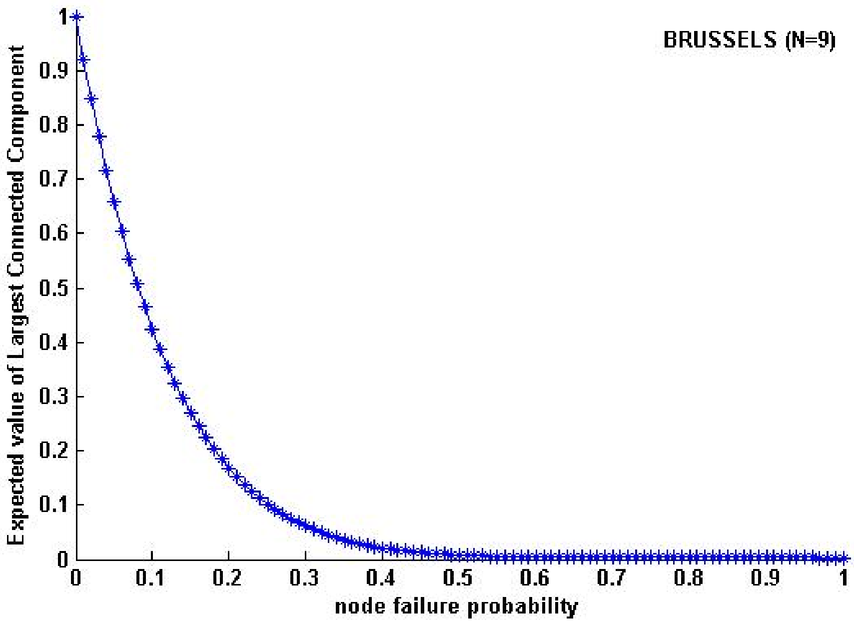


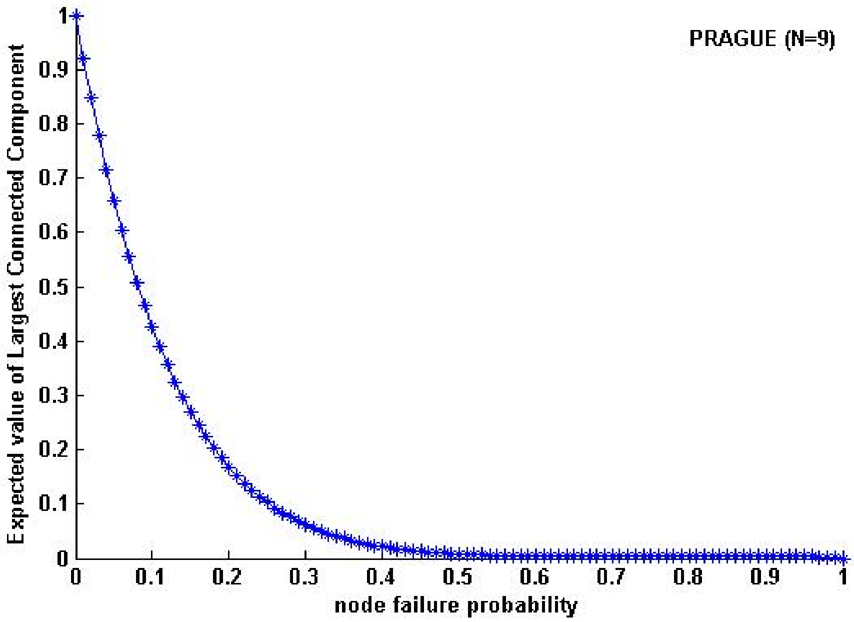


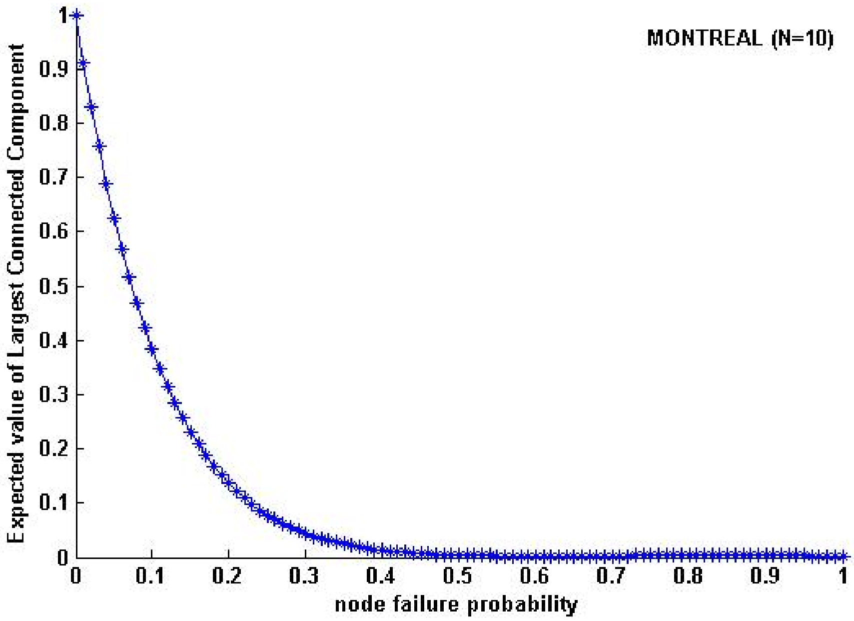


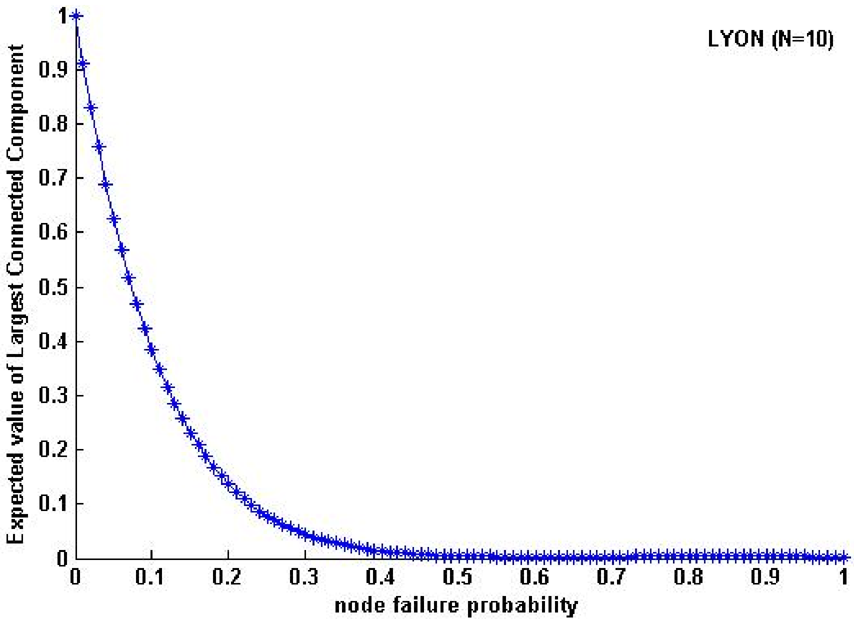

DELHI (N=8)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

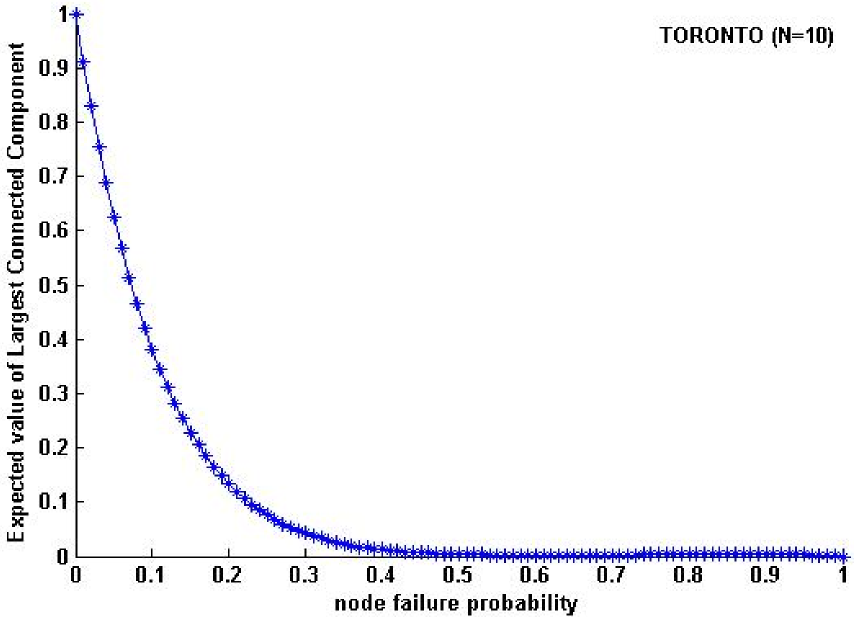

ATHENS (N=9)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

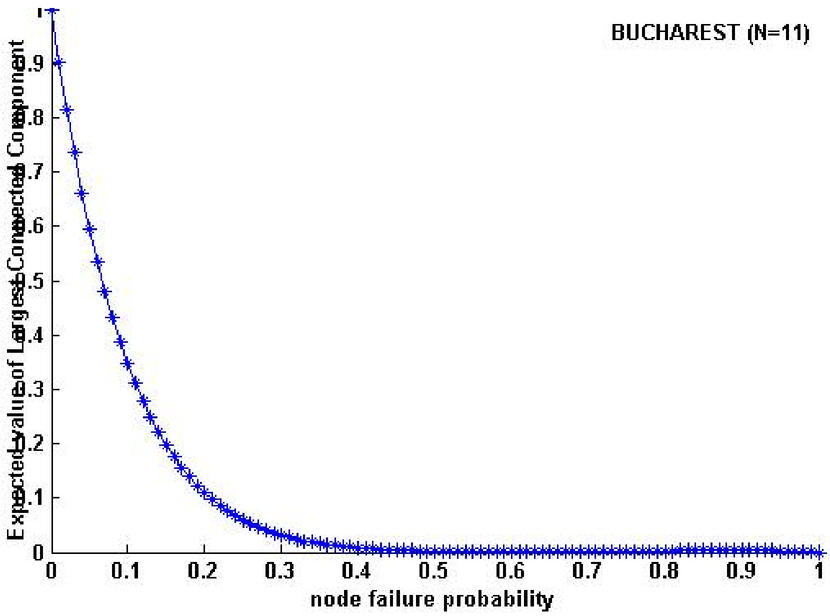

BRUSSELS (N=9)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

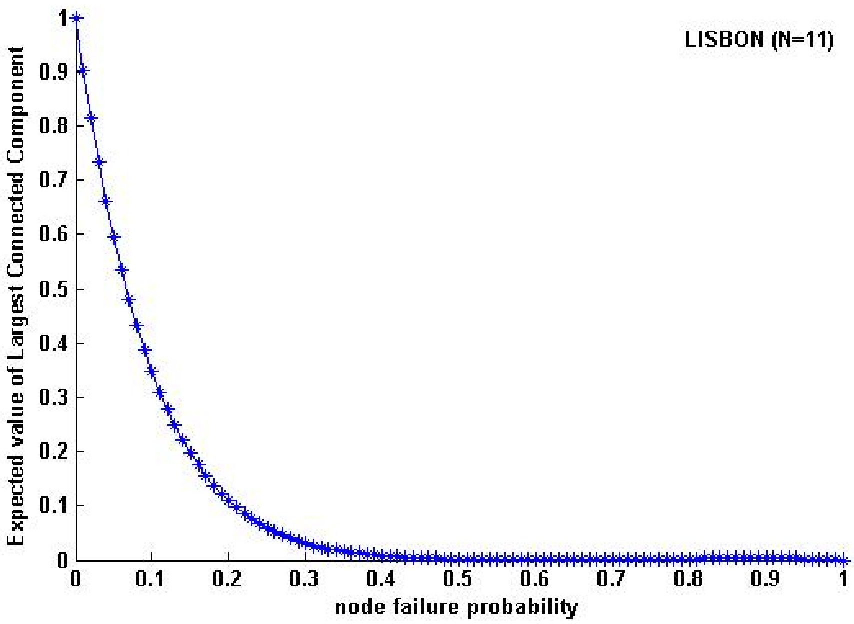

PRAGUE (N=9)
Expected value of Largest Connected Component
node failure probability
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

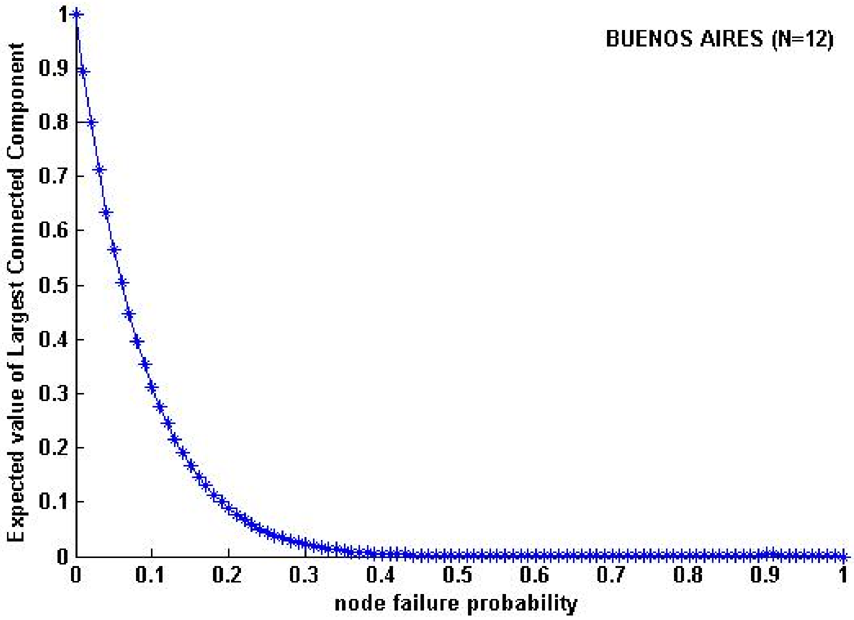

MONTREAL (N=10)
Expected value of Largest Connected Component
node failure probability
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

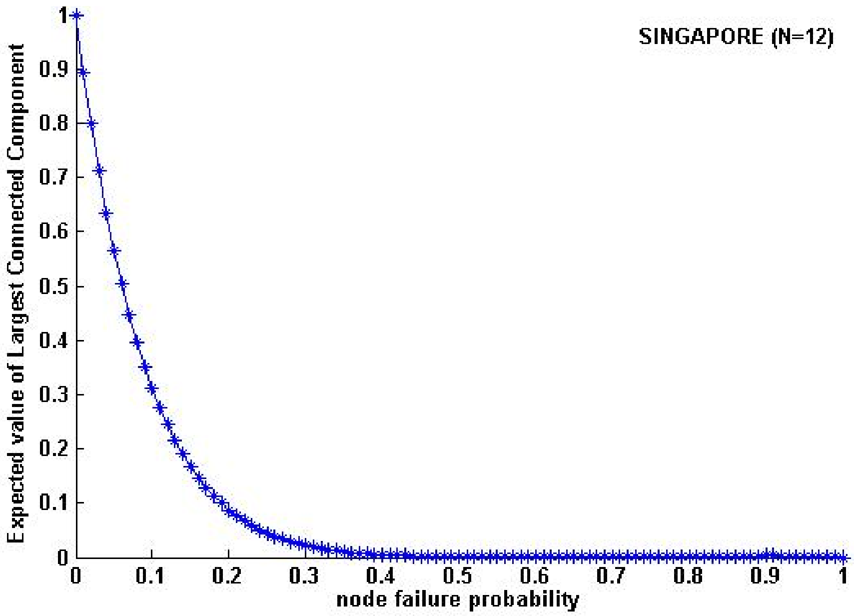

LYON (N=10)
Expected value of Largest Connected Component
node failure probability
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

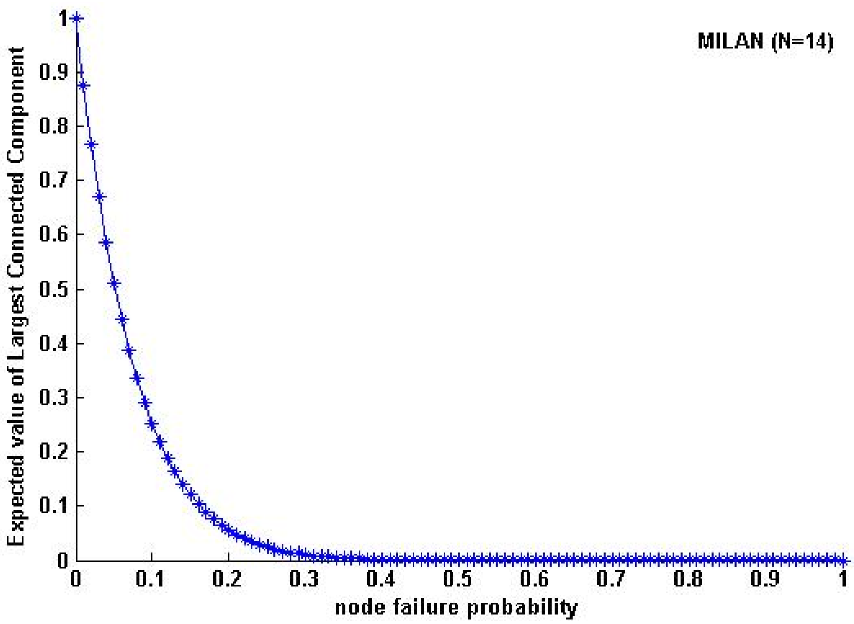

TORONTO (N=10)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

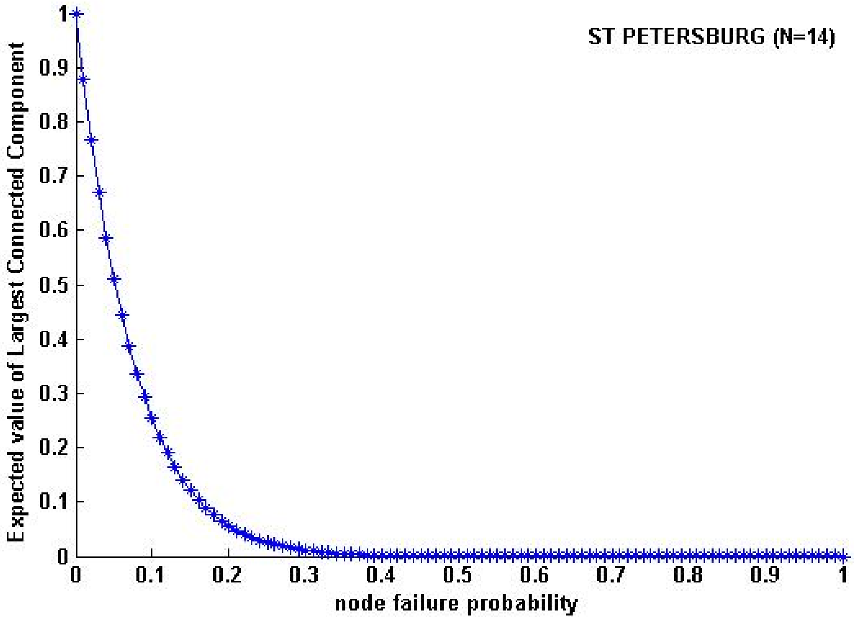

BUCHAREST (N=11)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

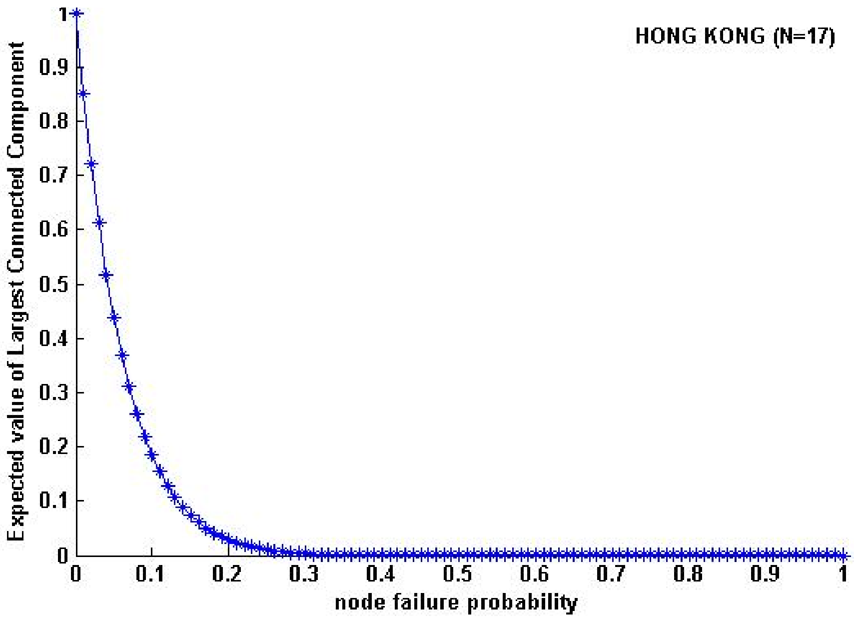

LISBON (N=11)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

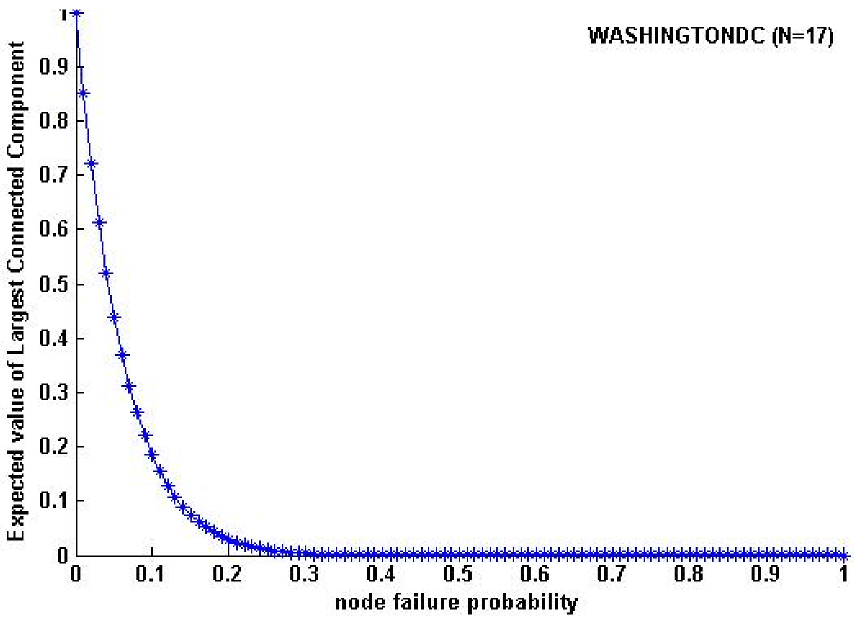

BUENOS AIRES (N=12)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

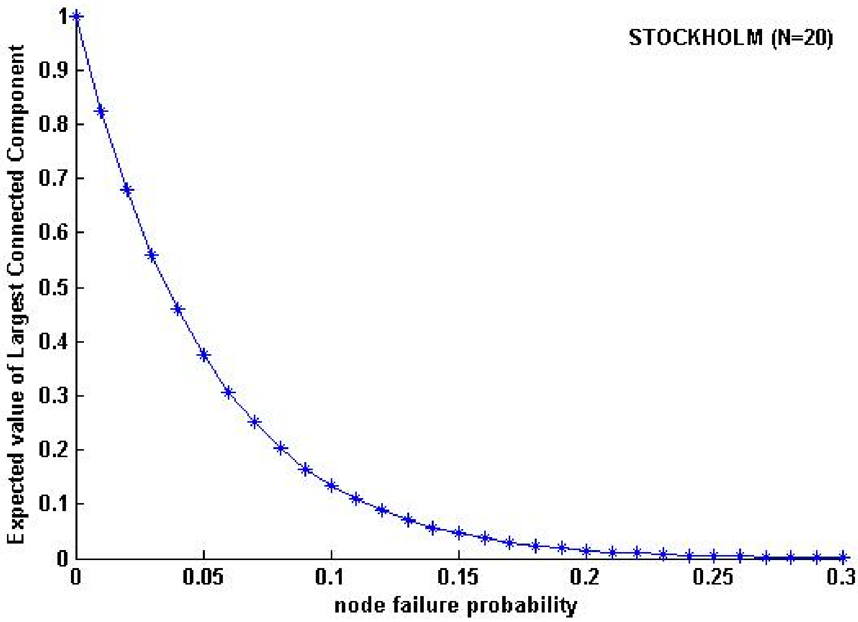

SINGAPORE (N=12)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

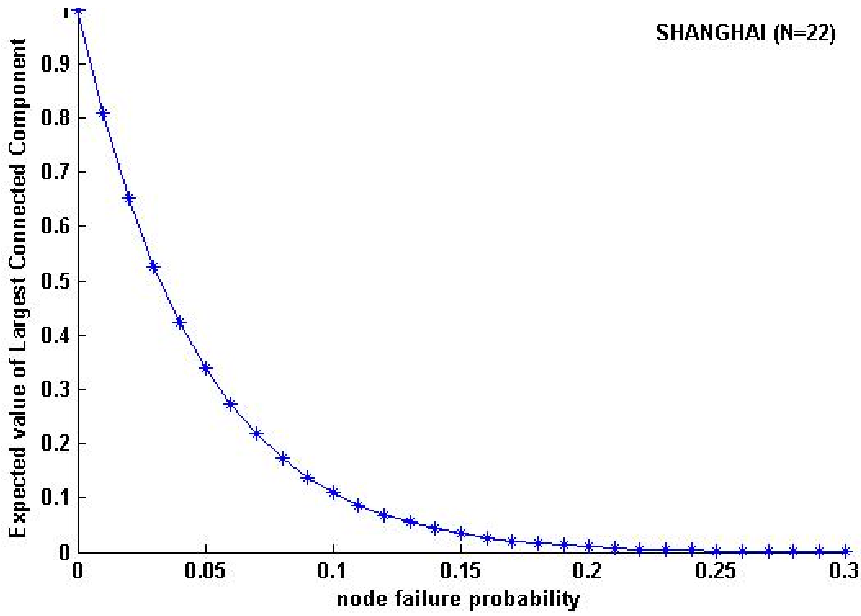

MILAN (N=14)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

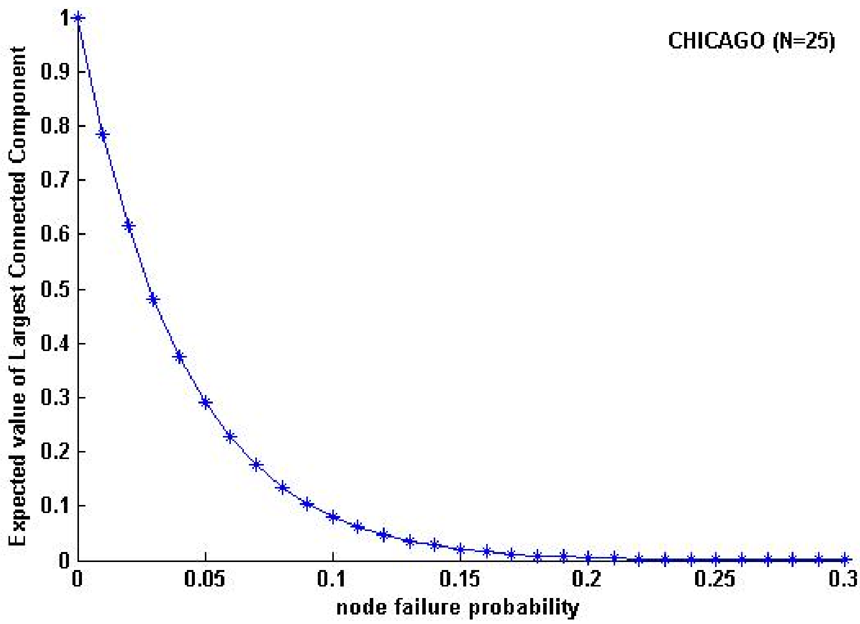

ST PETERSBURG (N=14)
Expected value of Largest Connected Component
node failure probability

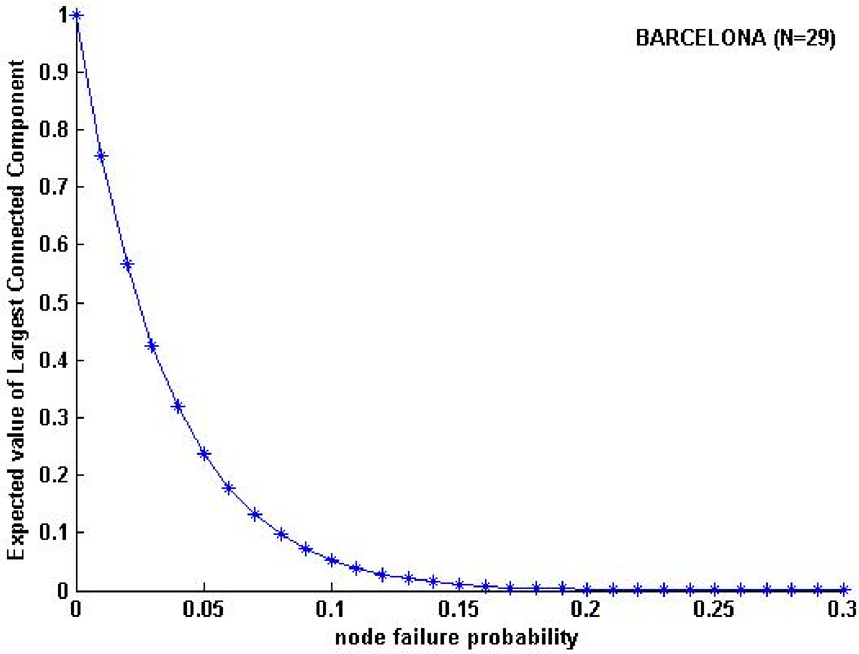

HONG KONG (N=17)
Expected value of Largest Connected Component
node failure probability

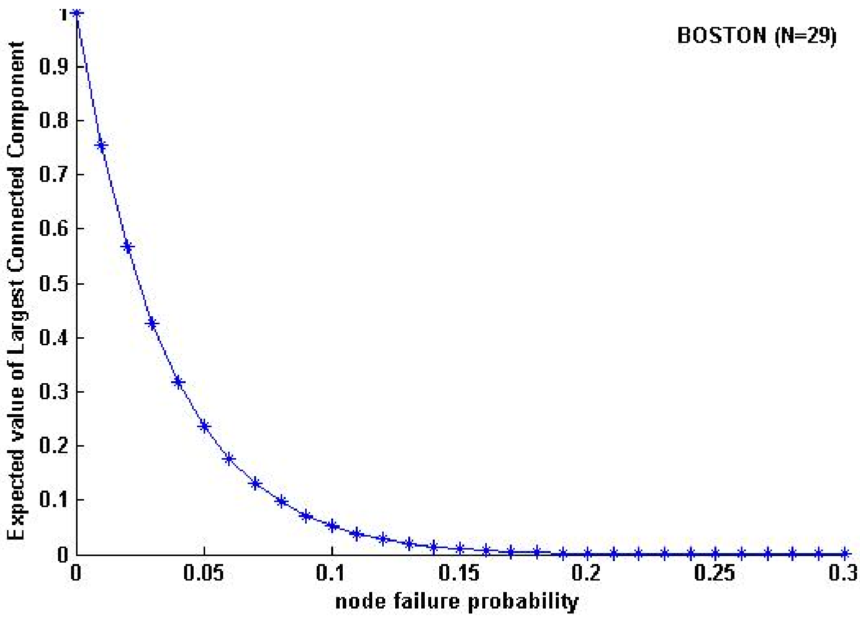

WASHINGTONDC (N=17)
Expected value of Largest Connected Component
node failure probability

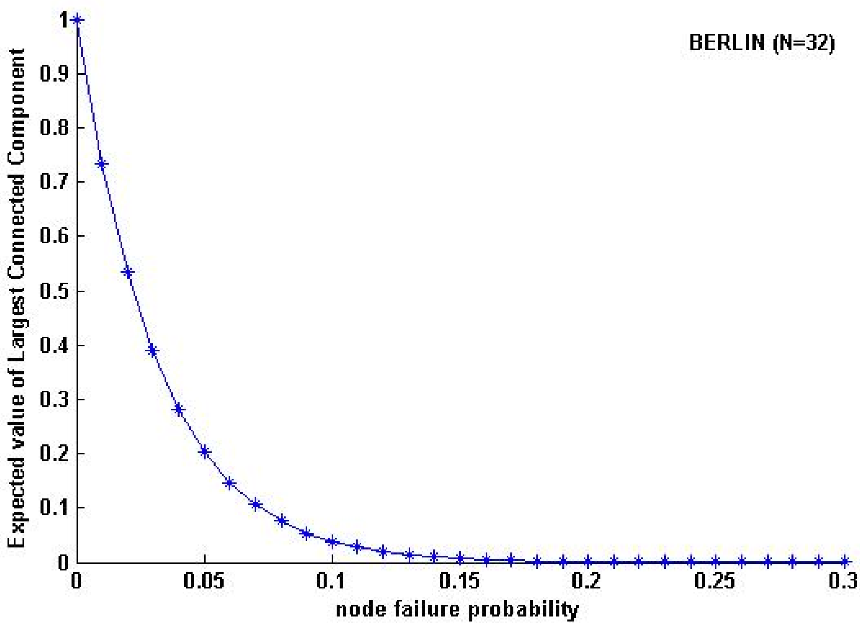

STOCKHOLM (N=20)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.05
0.15
0.25

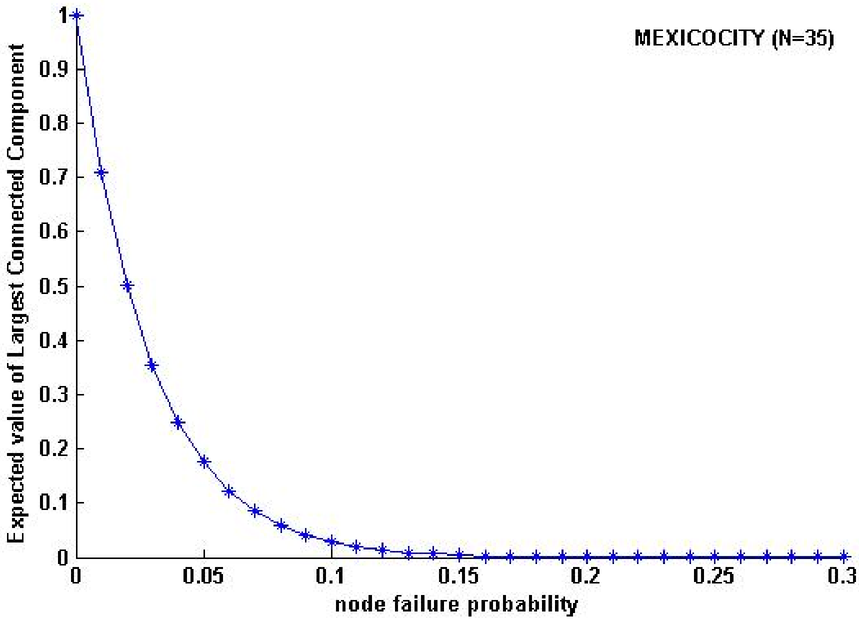

SHANGHAI (N=22)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0.05
0.15
0.25

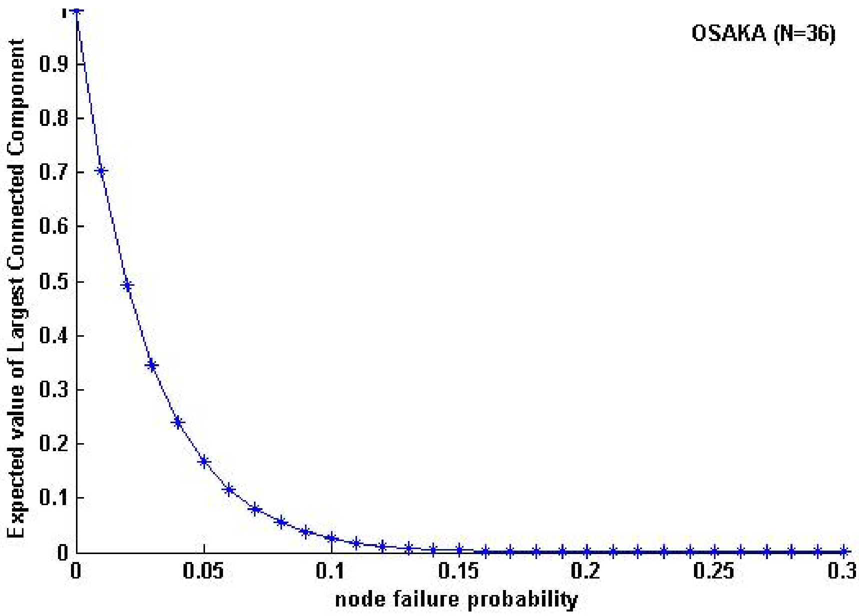

CHICAGO (N=25)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.05
0.15
0.25

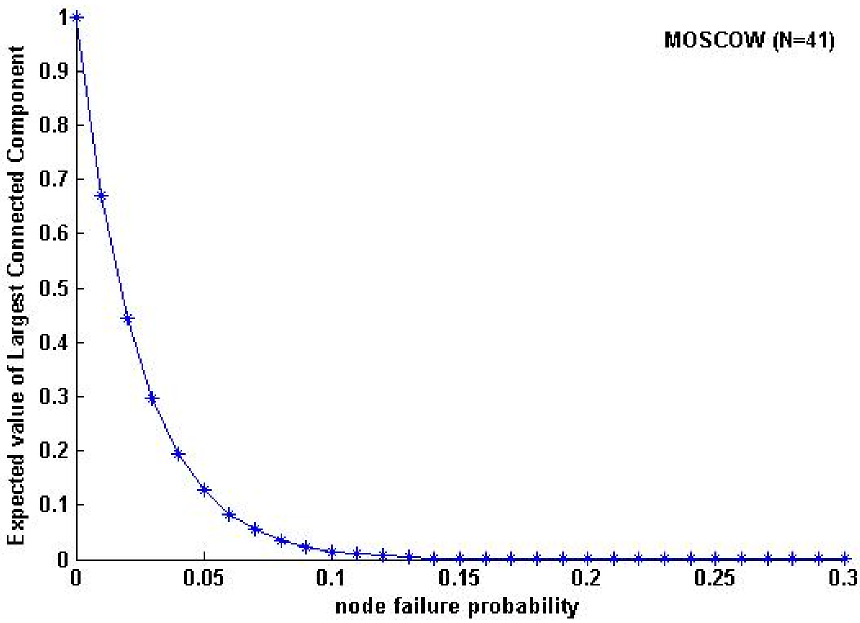

BARCELONA (N=29)
Expected value of Largest Connected Component
node failure probability
0 0.05 0.1 0.15 0.2 0.25 0.3
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

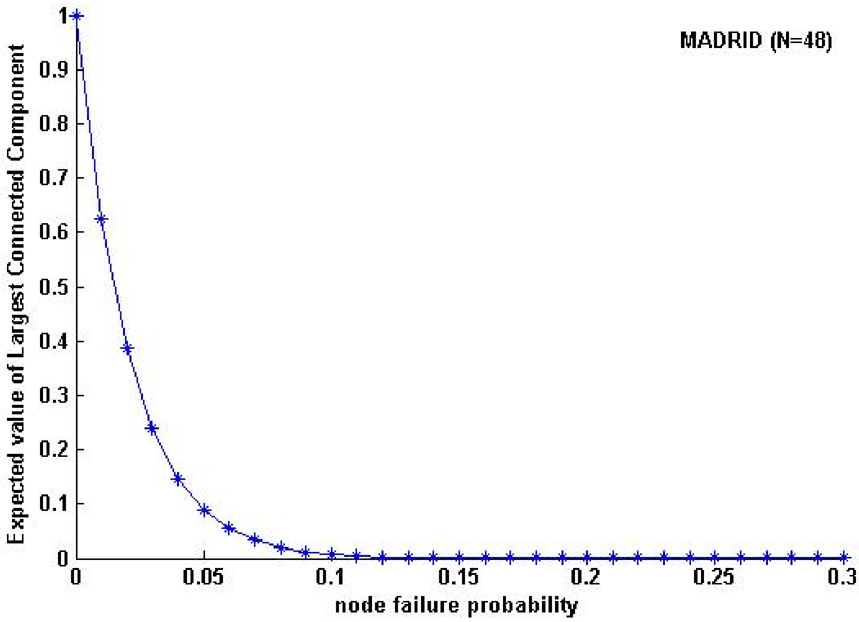

BOSTON (N=29)
Expected value of Largest Connected Component
node failure probability
0 0.05 0.1 0.15 0.2 0.25 0.3
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

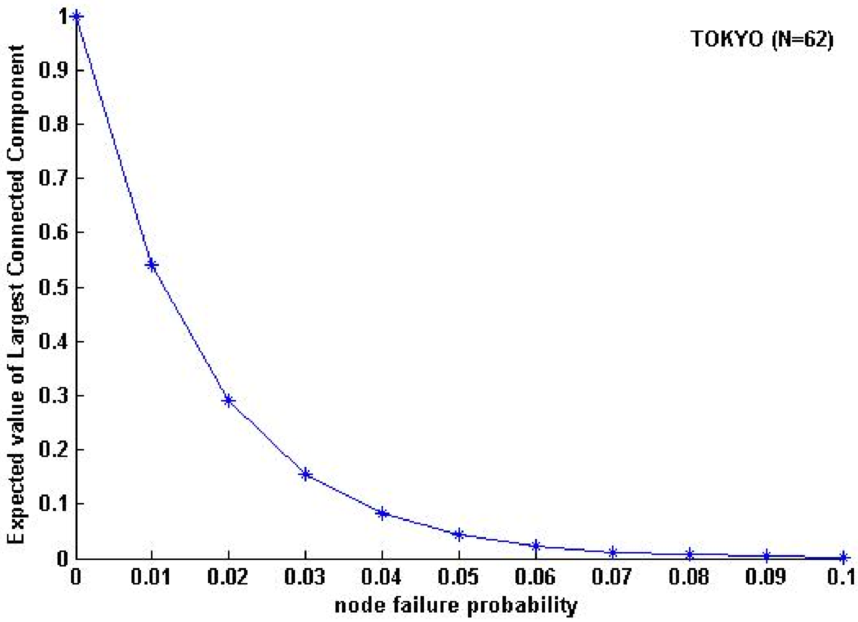

BERLIN (N=32)
Expected value of Largest Connected Component
node failure probability
0 0.05 0.1 0.15 0.2 0.25 0.3
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

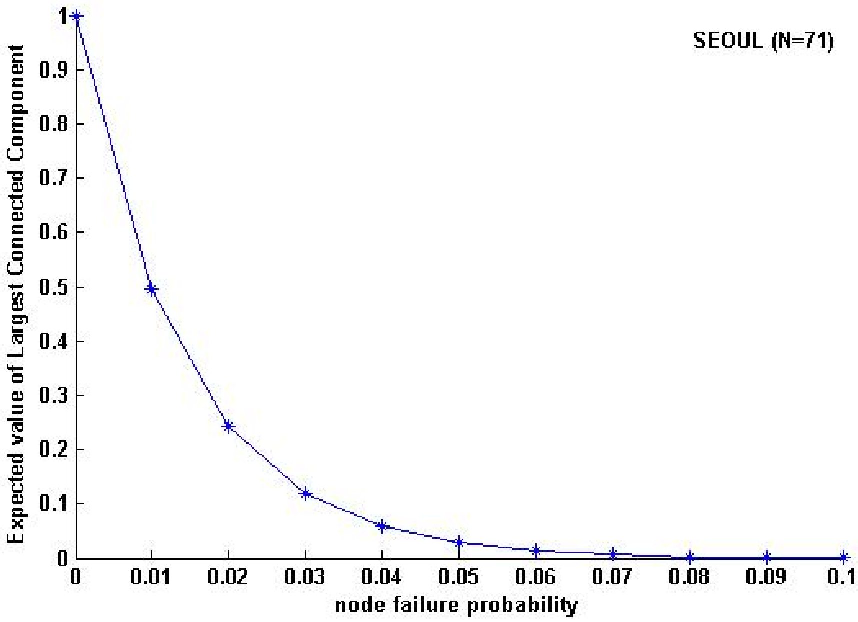

MEXICOCITY (N=35)
Expected value of Largest Connected Component
node failure probability

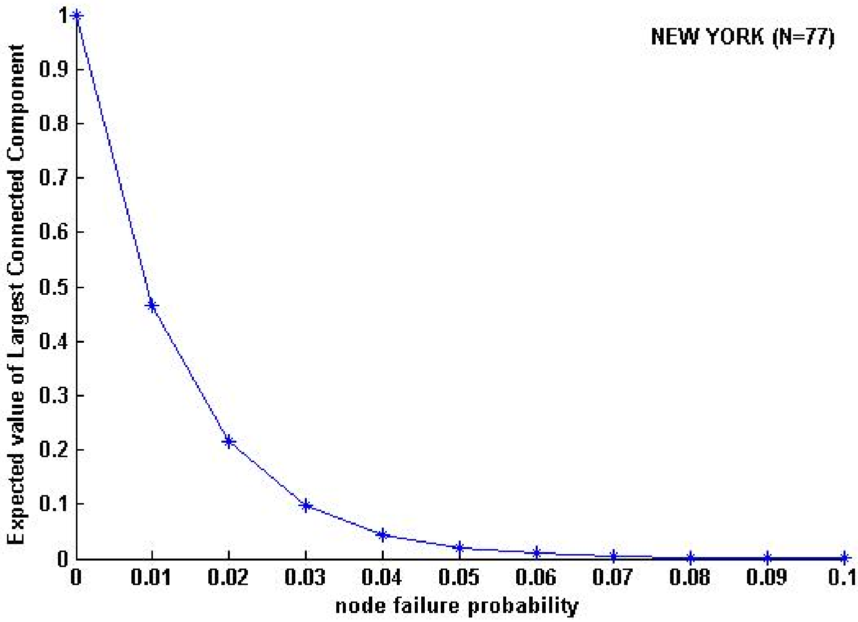

OSAKA (N=36)
Expected value of Largest Connected Component
node failure probability

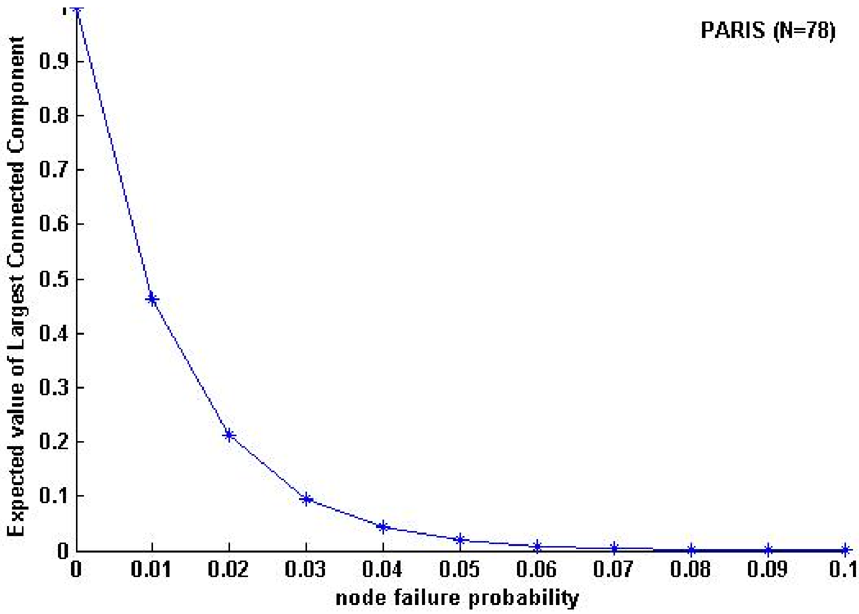

MOSCOW (N=41)
Expected value of Largest Connected Component
node failure probability

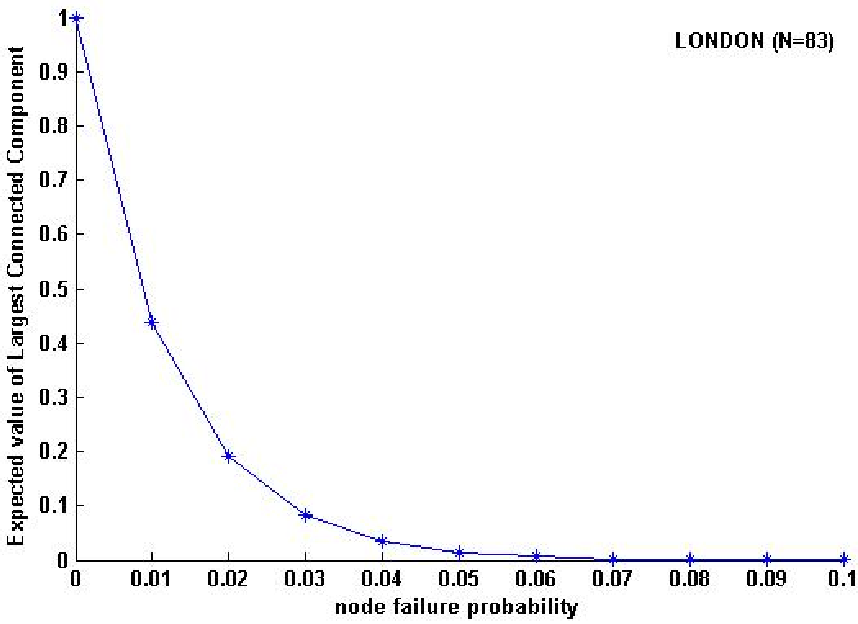

MADRID (N=48)
Expected value of Largest Connected Component
node failure probability


TOKYO (N=62)
Expected value of Largest Connected Component
node failure probability


SEOUL (N=71)
Expected value of Largest Connected Component
node failure probability


NEW YORK (N=77)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09


PARIS (N=78)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09


LONDON (N=83)
Expected value of Largest Connected Component
node failure probability
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09

Appendix B: MATLAB files

The algorithm to compute the expected value of the largest connected component as a function of failure probability is implemented by using MATLAB. In this part, the most important m-files will be introduced.

pg.m

pg is used to generate the polynomial of a network. It requires G, the graph of the metro network, as input and outputs the polynomial. The MATLAB code is:

```
function [Poly]=pq(G)
% Constructing P vector.This gives q^i*p^(n-i) in the form of col vector,
% with q = (1-p)
% Input:     G=Adjacency matrix

syms p q;
n = length(G);

L = [];
for I = 0:n
    temp = (q)^i;
    L = [L;temp];
end
Q = diag(L);

P = [];
for I = 0:n
    temp = (1-q)^(n-i);
    P = [P;temp];
end
K=diag(P);
Poly=Q*K;
```

maxCC.m

maxCC is used to generate the maximum dimension of a network after random node removal. It requires G, the graph of the metro network, as input and outputs the maximum dimension. The MATLAB code is:

```
function [maxDim] = maxCC(G)

maxDim = 0;
visited = zeros(1,length(G));

for i = 1:length(G)
   if (visited(i) == 1)
      continue;
   end
   [dim visited] = check_col(G,i,zeros(1,length(G)));
   if (dim > maxDim)
      maxDim = dim;
   end
```

```
end
```

largestcc.m
largest_cc is used to generate the largest connected component of a network after random node removal. It requires G, the graph of the metro network, as input and outputs the largest connected component. The MATLAB code is given below:

```
function [LCC] = largestcc(G)
%This function generates the largest connected component of a metro network
%Input: G = metro network

lenG = length(G);
alldata = zeros(1,lenG+1);

for i = 1:10000     % 10,000 iterations

    H = G;        % reset graph
    nodes = linspace(1,lenG,lenG);  % reset nodes

    for j = 1:lenG
        % select the node to remove
        index = floor(rand*length(nodes)) + 1;
        node = nodes(index);
        nodes = [nodes(1:index-1) nodes(index+1:length(nodes))];
        % remove the node
        for k = 1:lenG
            H(k,node) = 0;
            H(node,k) = 0;
        end

       alldata(j+1) = (maxCC(H)*.0001)./lenG + alldata(j+1);       % save data

    end
end

LCC = alldata;  %LCC = mean(alldata)
LCC(1) = lenG./lenG; %1st element is 1, maxCC = n with no nodes remove and
                       %devided by n gives 1
LCC (lenG+1) = 0; %Last element is 0 because after removing all the nodes the
                     %lcc will be 0
End
```

GetExpectedLCC.m
GetExpectedLCC is used to generate the expected value of the largest connected component of a network after random node removal. It requires G, the graph of the

metro network, as input and plots the expected value as a function of the node failure probability. The MATLAB code to realize this is given below:

```
function [Elcc, curve]=GetExpectedLCC(G)
%generates the expected value of largest connected component and plots the
%expected value of the largest connected component as a function of the
%node failure probability
%Input: G = metro network

[Poly]=pq(G);          %obtain the polynomial part
[LCC]=largestcc(G);       %compute LCC coefficients
Elcc=LCC*Poly;                  %compute expected value of LCC

e=sym2poly(sum(Elcc));
q=[0:.01:1];          % node failure probability
exp_val=polyval(e,q);
curve=plot(q,exp_val,'-b*');  % plot the curve
xlabel('node failure probability');
ylabel('Expected value of Largest Connected Component');
title ('Metro Name');
```