Homework Assignment 6: Network Flows

5273

Introduction

A water distribution system has a significant role in preserving and providing a desirable life to people, so the reliability of the supply is a key factor. Several works have defined the reliability of water distribution system [3, 4, 6] as the probability or ability of the system to meet the demands of consumers during a specified time. The demands are specified in terms of the flow rates at an adequate range of pressure.
The problem of determining connectivity of a network arises in issues of network design and reliability. In a network with random edge failures, the network could be partitioned at the minimum cuts. The minimum cut problem consists in given an undirected graph with n vertices and m edges, and it is need a partition of the vertices into two nonempty sets so as to minimize the number of edges crossing between them [5]. Su et al. [8] defines the minimum cut set as a set of system components(i.e., pipes) which, when failed, causes failure of the whole system. On the other hand, system failure will not happen if any of this set of components does not fail [2].
In this work it is studied a water distribution pipeline network throughout a connectivity analysis. It could be useful to have an idea of the reliability of the network, applying a minimum cut-set method [1], which allows the calculation of nodal pressures.
For this project the library NetworkX of Python has been used to generate an undirected powerlaw cluster graph of 121 nodes and 120 edges [7]. Edge weights have been generated randomly with non-negative values. This work it is run on an Intel Celeron CPU @ 1.10 GHz with 4 GB RAM laptop.

Case of Study: Water Distribution System

It is presented a case about a Water Distribution Supplier (WDS), represented as node 0. The system is composed of approximately 120 pipes and 121 demand nodes (corresponding to users) that are spread across an area. To analyze the interrelationship among the components, the system is first transformed into a network repesentation.

```python
G=nx.balanced_tree(3,4)

n_nodes=len(G)
n_edges=nx.number_of_edges(G)

print("Number of nodes:",n_nodes)
print("Number of edges:",n_edges)

#Assign normally distributed weights to edges:
weights = np.random.normal(3, 0.5, nx.number_of_edges(G))
w = 0
for u, v, d in G.edges(data=True):
    d['weight'] = weights[w]
    w += 1

#Change node properties:
color_map = []
node_sizes = []
for node in G:
    if node == 0:
        color_map.append('skyblue')
        node_sizes.append(900)
    else:
        color_map.append('red')
        node_sizes.append(200)
```

graphs_t6.py

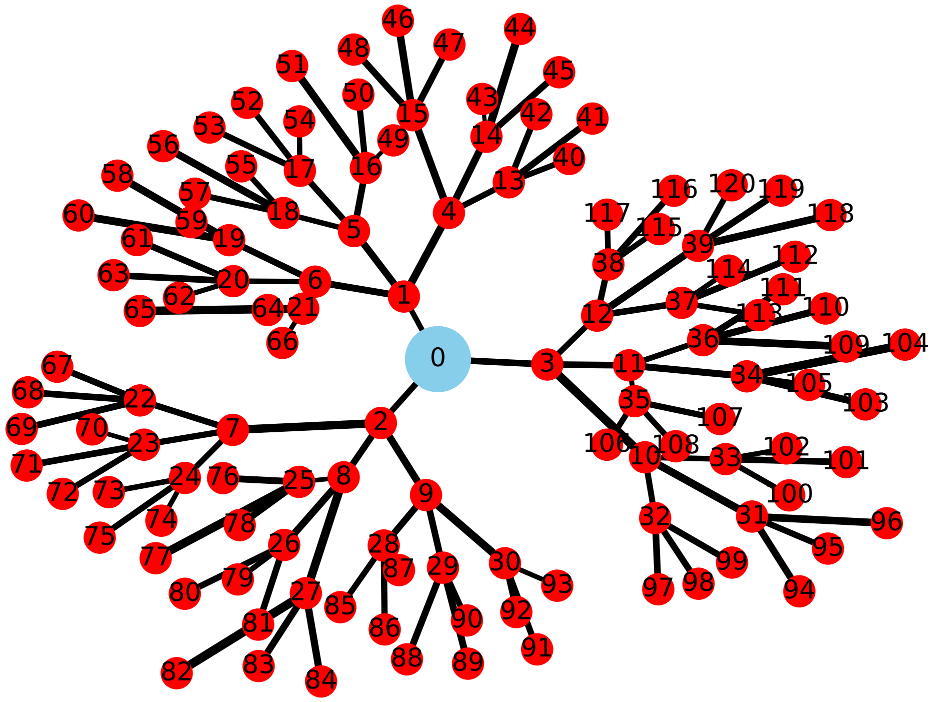

Figure 1: Water distribution network

Figure 1 shows this graph representation where can be seen 3 different clusters. This groups represent the neighbourhoods where water needs to be transported.
For Cluster 1 is run the maximum flow algorithm, collecting data of the flow values. The source node is always WDS and the target is the demanding node of that cluster. This algorithm is applied to every node of the cluster. The data has a mean of 2.61 units with a standard deviation of 0.474. With these flow values a histogram is constructed.

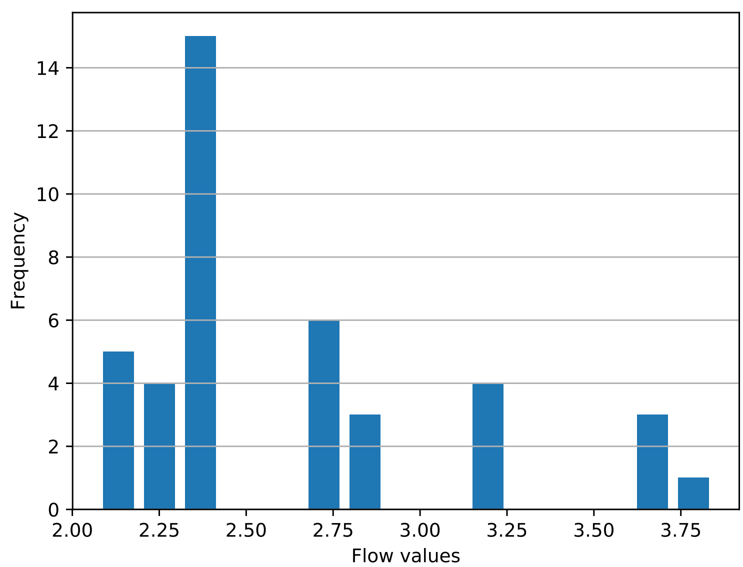

Figure 2: Histogram of water flow values for Cluster 1

Shapiro-Wilk Test is performed in order to demonstrate if the data is normally distributed. The test shows a $p-value$ of 9.42×10^{-5} so the data is not normally distributed.

```
# Analysis for Cluster 1:
for i in cluster1:
    #Maximum flow algorithm:
    flow_value,flow_dict = nx.maximum_flow(G, 0, i, capacity='weight')
    c1_values.append(flow_value)

    df=pd.DataFrame({'Cluster':[1],
                     'Flow_Value':flow_value})

    all_data=all_data.append(df)

mean=np.mean(c1_values)
std_dev=np.std(c1_values)

normality_test=stats.shapiro(c1_values)

print("Mean for Cluster 1:",mean)
print("Standard deviation for Cluster 1:",std_dev)
print("Normality test for Cluster 1:",normality_test,"\n")

#Histogram for cluster 1:
hist, bin_edges=np.histogram(c1_values,density=True)
first_edge, last_edge = np.min(c1_values),np.max(c1_values)

n_equal_bins = 15
bin_edges = np.linspace(start=first_edge, stop=last_edge,num=n_equal_bins + 1, endpoint=True)

plt.hist(c1_values,bins=bin_edges,rwidth=0.75)
plt.xlabel('Flow values')
plt.ylabel('Frequency')
plt.grid(axis='y', alpha=0.75)
plt.savefig("Histogram_Cluster1.eps", format="EPS")
plt.show(1)
```

graphs_t6.py

For Cluster 2 is run the maximum flow algorithm, collecting data of the flow values. The source node is always the WDS and the target is the demanding node of that cluster. This algorithm is applied to every node of the cluster. The data has a mean of 2.54 units with a standard deviation of 0.219. With these flow values a histogram is constructed.

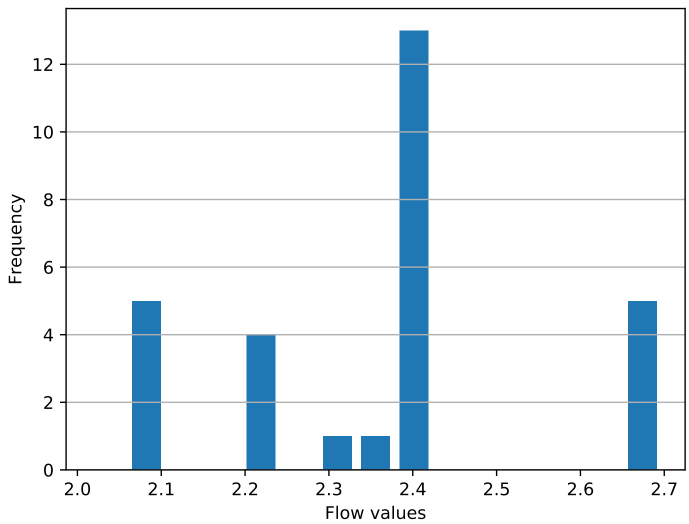

Figure 3: Histogram of water flow values for Cluster 2

Shapiro-Wilk Test is performed in order to demonstrate if the data is normally distributed. The test shows a $p-value$ of 2.90×10^{-7} so the data is not normally distributed.

```python
# Analysis for Cluster 2:
for j in cluster2:
    # Maximum flow algorithm:
    flow_value, flow_dict = nx.maximum_flow(G, 0, j, capacity='weight')
    c2_values.append(flow_value)

    df = pd.DataFrame({'Cluster': [2],
                       'Flow_Value': flow_value})

    all_data = all_data.append(df)

mean=np.mean(c2_values)
std_dev=np.std(c2_values)

normality_test=stats.shapiro(c2_values)

print("Mean for Cluster 2:",mean)
print("Standard deviation for Cluster 2:",std_dev)
print("Normality test for Cluster 2:",normality_test,"\n")

#Histogram of Cluster 2
hist, bin_edges=np.histogram(c2_values,density=True)
first_edge, last_edge = np.min(c2_values),np.max(c2_values)

n_equal_bins = 15
bin_edges = np.linspace(start=first_edge, stop=last_edge,num=n_equal_bins + 1, endpoint=True)

plt.hist(c1_values,bins=bin_edges,rwidth=0.75)
plt.xlabel('Flow values')
plt.ylabel('Frequency')
plt.grid(axis='y', alpha=0.75)
plt.savefig("Histogram_Cluster2.eps", format="EPS")
plt.show(2)
```

graphs_t6.py

For Cluster 3 is run the maximum flow algorithm, collecting data of the flow values. The source node is always WDS and the target is the demanding node of that cluster. This algorithm is applied to every node of the cluster. The data has a mean of 2.79 units with a standard deviation of 0.35. With these flow values a histogram is constructed.

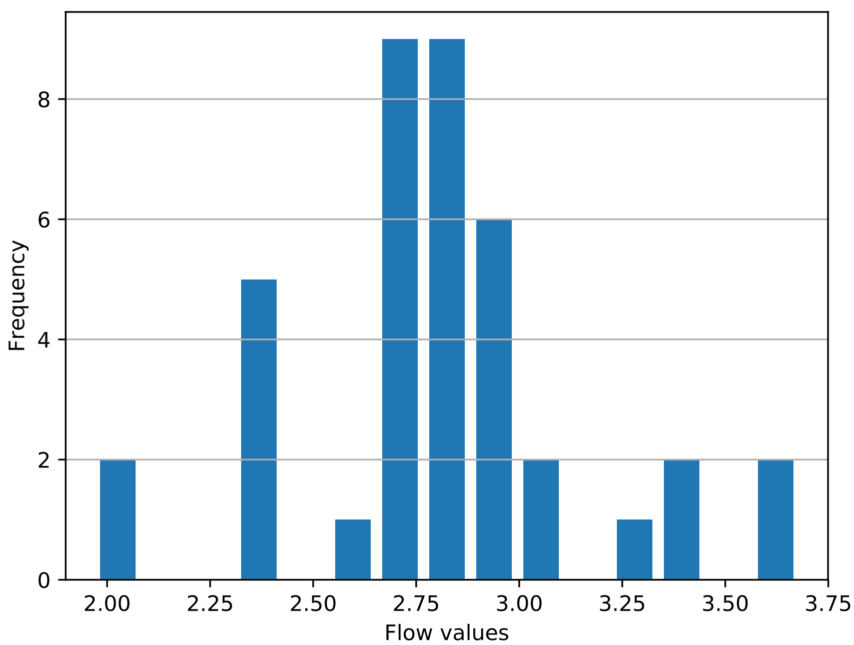

Figure 4: Histogram of water flow values for Cluster 3

Shapiro-Wilk Test is performed in order to demonstrate if the data is normally distributed. The test shows a $p - value$ of 0.044 so the data is not normally distributed.

```
hist, bin_edges=np.histogram(c3_values,density=True)
first_edge, last_edge = np.min(c3_values),np.max(c3_values)

n_equal_bins = 15
bin_edges = np.linspace(start=first_edge, stop=last_edge,num=n_equal_bins + 1, endpoint=True)

plt.hist(c3_values,bins=bin_edges,rwidth=0.75)
plt.xlabel('Flow values')
plt.ylabel('Frequency')
plt.grid(axis='y', alpha=0.75)
plt.savefig("Histogram_Cluster3.eps", format="EPS")
plt.show(3)
```

graphs_t6.py

In order to compare the demanding amount of water of every cluster a boxplot is constructed. Here we can see Cluster 3 is a bit more water demanding than the others. Cluster 1 is the one with more dispersion and also have the outlier with higuest flow value of all clusters.

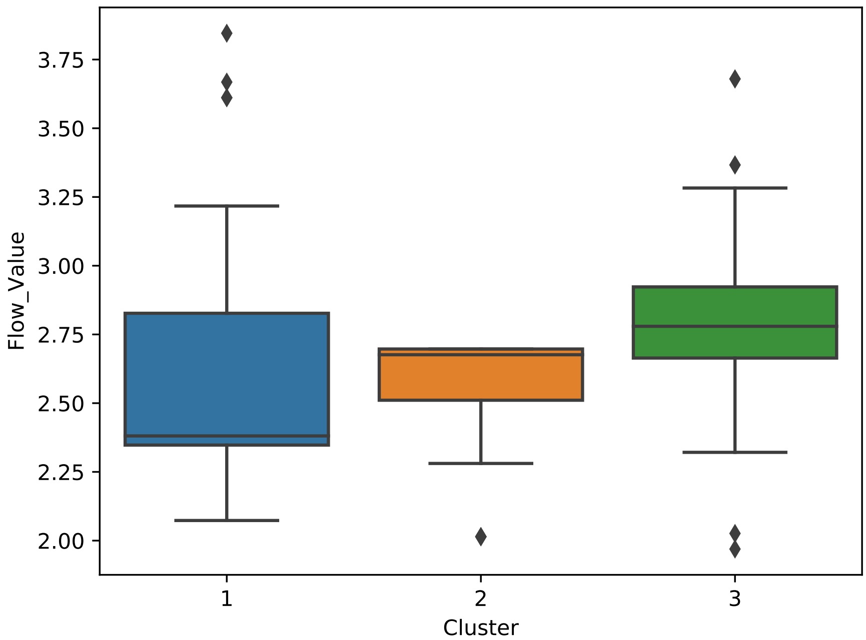

Figure 5: Boxplot of water flow values for all Clusters

The minimum cut set approach could be applied to calculate this system reliability. The impact of link failures on source-demand connectivity it could be a measure of the impact of the mechanical reliability of the water distribution network [9]. The focus is on whether a demand node can get any water from the available sources. After the cut the network is divided in to sets of nodes:

Set A	Set B
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111}	{37, 38, 39, 12, 112, 113, 114, 115, 116, 117, 118, 119, 120}

Table 1: Sets of nodes after the minimum cut.

The cut is performed at the pipes with lower demands (lower edge weights). Here the minimum cut value is the sum of the capacities of the cut edges. This value when the algorithm is executed is 2.664 units.

References

[1] Muhammad A Al-Zahrani and Juned Laiq Syed. Evaluation of municipal water distribution system reliability using minimum cut-set method. *Journal of King Saud University-Engineering Sciences*, 18(1):67–81, 2005.

[2] Roy Billinton and Ronald Norman Allan. *Reliability evaluation of engineering systems.* Springer, 1992.

[3] M John Cullinane, Kevin E Lansey, and Larry W Mays. Optimization-availability-based design of water-distribution networks. *Journal of Hydraulic Engineering*, 118(3):420–441, 1992.

[4] Ian Goulter. Analytical and simulation models for reliability analysis in water distribution systems. In *Improving efficiency and reliability in water distribution systems*, pages 235–266. Springer, 1995.

[5] David R Karger and Clifford Stein. A new approach to the minimum cut problem. *Journal of the ACM (JACM)*, 43(4):601–640, 1996.

[6] Arnold Kaufmann, Roger Cruon, and Daniel Grouchko. *Mathematical Models for the Study of the Reliability of Systems.* Elsevier, 1977.

[7] NetworkX. Source code for networkx.drawing.layout, 2018. `https://networkx.github.io/documentation/latest/_modules/networkx/drawing/layout.html`, Last accessed on 2019-05-10.

[8] Yu-Chun Su, Larry W Mays, Ning Duan, and Kevin E Lansey. Reliability-based optimization model for water distribution systems. *Journal of Hydraulic Engineering*, 113(12):1539–1556, 1987.

[9] Shu-Li Yang, Nien-Sheng Hsu, Peter WF Louie, and William WG Yeh. Water distribution network reliability: Connectivity analysis. *Journal of Infrastructure Systems*, 2(2):54–64, 1996.