Challenge 1: Topology Design

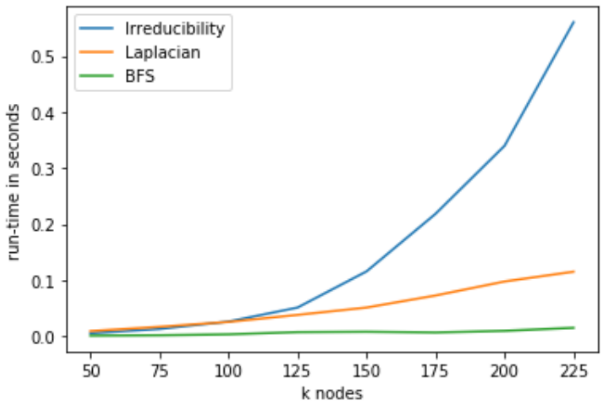
1 Connectivity

In this part of the assignment we aim to compare the complexity of different algorithms that check the connectivity of two types of graph, namely r-regular random graphs and Erdős-Rényi, shortly p-ER random graphs, which represent two different alternatives for generating a Jellyfish topology. Secondly, we also study how the probability of being connected of such graphs varies with regards to specific parameters.

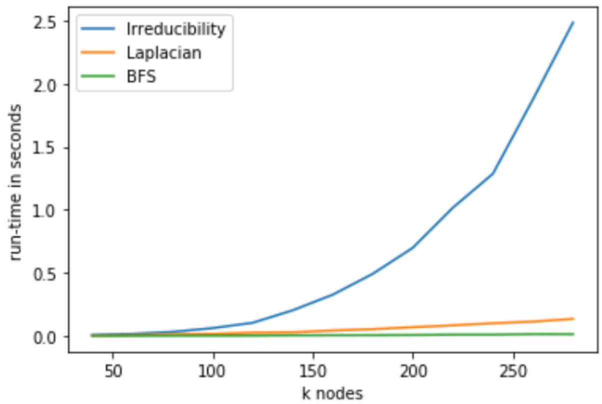
Let K denote the number of nodes of the graph.

1. The three methods used to check the connectivity of a given graph are: 1) Irreducibility of the adjacency matrix, 2) Nonnegativity of the second eigenvalue of the Laplacian matrix and 3) Breadth-first-search algorithm. In our analysis, the complexity measure is defined as the time required by a specific machine to execute the script. The plots below show the results versus the number of nodes:

p-ER random graph



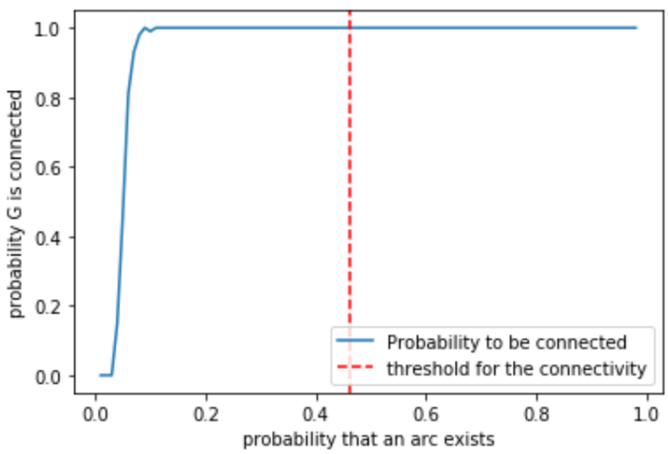
r-regular random graph



Analyzing the plots it's clear that the method that uses irreducibility has an higher complexity than the other two. The difference between the second and third methods is not so significant but *BFS* seems to be the most efficient one, especially when dealing with p-ER graphs.

2. Now we fix $K = 100$ and we plot the probability of a p-ER random graph of being connected as a function of the probability parameter p. We use *BFS* to check connectivity since the previous point suggests it is the most efficient criterion.

p-ER random graph

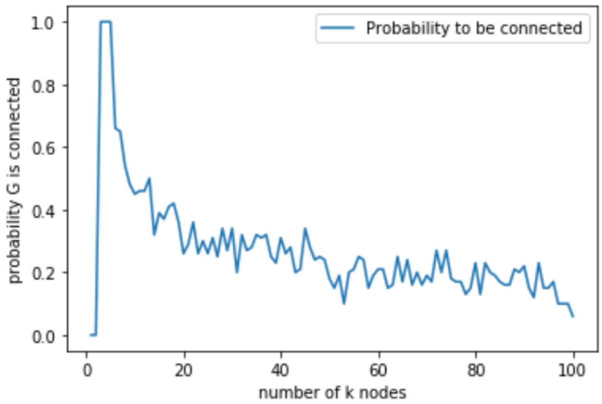


We can clearly see from the plot on the left that the probability of being connected grows very fast to 1 even when the probability that an arc is added in the graph is low. But this is actually strictly related to the number of nodes of the graph. In fact, a theoretical result by Erdős and Rényi[1] states that a sharp threshold for the connectivity of a p-ER graph given by $\frac{\ln K}{K}$. In our specific case $K = 100$, therefore we expect the graph to be connected for any $p > \frac{\ln K}{K} = \frac{\ln 100}{100} \approx 0046$.

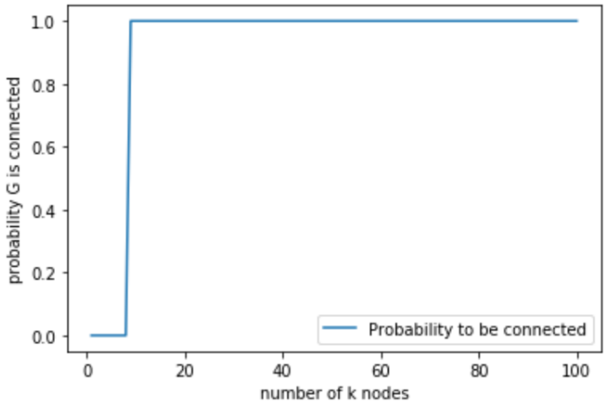
[1] P. Erdős, A. Rényi, *"On the evolution of random graphs"* (1960)

3. We repeat the same analysis for the case of r-regular random graphs. In this case we plot the probability of being connected as a function of the number K of nodes, for $d \leq 100$ and $r \in \{2, 8\}$.

2-regular random graph



8-regular random graph



In the graph on left, obtained for $r = 2$, we see that, unlike the case of p-ER graphs, the probability of being connected decreases when the number of nodes K increases. This means that for big graphs where the nodes can have only 2 neighbors is very difficult to be connected. The plot on the left shows instead a completely different situation. With $r = 8$ the probability that the graph is connected grows up to 1 also with small values of nodes. This behaviour is clarified by a theorem which states that, given a r-regular random graph having $r \geq 4$ and odd, a sufficient condition to be connected is to have $2r - 1$ nodes.

2 Traffic model and throughput

In this part of the assignment we compare the throughput bound of the Fat-tree topology with that of a r-regular random (i.e. Jellyfish) topology having the same equipment. Let N denote the number of servers, S the number of n-ports switches and L the number of bidirectional links of the network, also including all links between switches and servers.

1. Once n is fixed, the values of N, S and L can be immediately derived from the properties of the Fat-tree topology. Specifically, they can be written as functions of n as follows:

$$N = \frac{n^3}{4} \qquad S = \frac{5}{4}n^2 \qquad L = 3N = \frac{3}{4}n^3$$

In the Jellyfish topology the number r of switch ports to be connected to other switches is involved in the inequality $N \leq S(n-r)$. Therefore, assuming the network to be equipped with the maximum possible number of servers, we derive the following for r:

$$r = n - \frac{N}{S} = n - \frac{n^3}{4}\frac{4}{5n^2} = n - \frac{n}{5} = \frac{4}{5}n$$

. Notice that this choice of r also leads to same number of links of Fat-tree:

$$L_J = \frac{Sr}{2} + \frac{n^3}{4} = \frac{1}{2} \cdot \frac{5}{4}n^2 \cdot \frac{4}{5}n + \frac{n^3}{4} = \frac{n^3}{2} + \frac{n^3}{4} = \frac{3}{4}n^3 = L_{FT} = L$$

. We also check that this value of r satisfies the requirement of a general r-regular graph:

- $r < S$: $\frac{4}{5}n < \frac{5}{4}n^2 \Leftrightarrow n > \lceil \frac{16}{25} \rceil = 1$
- the product rS is even, i.e. $rS = 2k$ for some $k \in \mathbb{N}$
- since r must be an integer, there might be networks in which not all nodes have equal degree. In particular, if $S > N$, then $S - N$ switches will use $\lceil r \rceil$ ports to connect to other switches and N will only use $\lfloor r \rfloor$ ports, whereas if instead $N > S$, then N switches will use $\lceil r \rceil$ ports for inter-switches connection and $N - S$ will use $\lfloor r \rfloor$ ports.

2. We can now write the expression for the application-oblivious throughput bound TH for an all-to-all traffic matrix as a function of n and of the average shortest path length $\overline{h}$. In particular, we have that:

   - $\ell = L = 3N = \frac{3}{4}n^3$
   - $\nu_f = \binom{N}{2} = \frac{N(N-1)}{2} = \frac{n^3}{8}(\frac{n^3}{4} - 1) = \frac{n^3(n^3-4)}{32}$

   Therefore, substituting in the general formula we get the following:

$$TH = TH(n, \overline{h}) \leq \frac{\ell}{\overline{h}\nu_f} = \frac{6}{\overline{h}(N-1)} = \frac{24}{\overline{h}(n^3-4)}.$$

3. Finally, we compare numerically the bound on TH in both topologies for $n = 5\ell,\ \ell = 1, \dots, 10$. In the case of the Fat-tree topology, we can use the exact value of $\overline{h}$, which is given by:

$$\overline{h} = \frac{1}{\nu_f}\sum_{1=1}^{\nu_f} h_i = \left(2\left[\frac{n^3}{8}\left(\frac{n}{2}-1\right)\right] + 4\left[\frac{n^4}{16}\left(\frac{n}{2}-1\right)\right] + 6\left[\frac{n^5}{32}(n-1)\right]\right)\frac{1}{\nu_f} =$$
$$= 2 \cdot \frac{3n^3 - n^2 - 2n - 4}{n^3 - 4}.$$

   Instead, in order to estimate the lower bound on $\overline{h}$ in the case of the r-regular random graph, we first have to calculate these quantities:

$$K = K(n) = 1 + \left\lfloor \frac{\log(N - \frac{2(N-1)}{r})}{\log(r-1)} \right\rfloor = 1 + \left\lfloor \frac{\log(\frac{10n^3 - 25n^2 - 20}{8n})}{\log(\frac{4n-5}{5})} \right\rfloor$$

$$R = R(n, K) = N - 1 - \sum_{j=1}^{K-1} r(r-1)^{j-1} = \frac{5}{4}n^2 - 1 - \sum_{j=1}^{K-1} \frac{4n}{5}\left(\frac{4n}{5} - 1\right)^{j-1}$$

   Thus, from a known result we get the lower bound: $\overline{h} \geq \frac{\sum_{j=1}^{K-1} j\frac{4n}{5}(\frac{4n}{5}-1)^{j-1} + Kr}{\frac{5}{4}n^2 - 1}$

   Our results are summarized in the following table, where both TH_{FT} and TH_J are intended to be upper bound on the traffic flow:

n	N	S	L	TH_{FT}	TH_J
5	31.25	31.25	93.75	0.03571	0.05621
10	250	125	750	0.00417	0.00638
15	843.75	281.25	2531.25	0.00122	0.00185
20	2000	500	6000	0.00051	0.00077
25	3906.25	781.25	11718.75	0.00026	0.00039
30	6750	1125	20250	0.00015	0.00023
35	10718	1531.25	32156.35	9e-05	0.00014
40	16000	2000	48000	6e-05	9e-05
45	22781.25	2531.25	68343.75	4e-05	7e-05
50	31250	3125	93750	3e-05	5e-05

   What we learn from the table above is that the Jellyfish topology supports more flows at high throughput thanks to its average shortest path length.