AMSC808N-HW6

Chenyang Fang
https://github.com/Chenyang-Fang/AMSC808N

December 3, 2020

1. In the problem, set the number of vertices $n = 1000$. Define a grid with size 20 of values of z ranging between 0 and 4. For each z, generate $r = 100$ random graphs $G(n, p)$, where $p = z/(n-1)$. For each graph, the numerical solution and theoretical solution are given. For the numerical solution, the DFS is defined in the matlab code. The input is the generated random graph, the output are a predecessor subgraph for a given graph $G(V, E)$. For the theoretical solution, the fraction S of vertices in the ginat component is the largest solution of

$$S = 1 - exp(-zS).$$

The matlab build-in function ***vpasolve*** is used to calculate the solution. Then two results are showing in the figure 1.
In the similar way, the theoretical solution for the average size of the non-giant component to which v belongs is

$$\langle s \rangle = \frac{1}{1 - z + zS}.$$

And the results are plotting in the figure 2. Here are some comments on the findings:

- In the figure 1, it is illustrate that the numerical solution can achieve better performance and is close to the theoretical solution. While in the figure 2, the numerical solution works well except when z is near to the 1.
- When z is close to 1, the value of $s(z)$ is very large. This is because when $z = 1$, the value of $S(z)$ is close to 0, so the value of $s(z) = \frac{1}{1-z+zS} \to \infty$

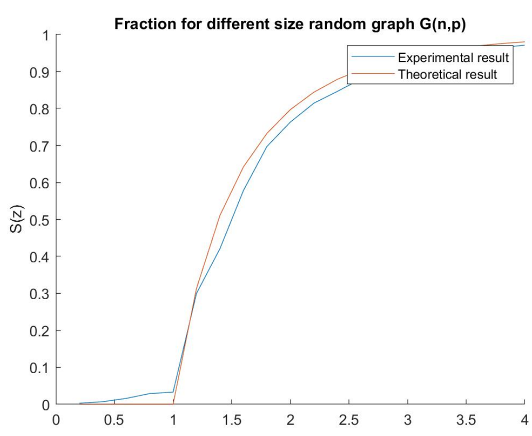

Figure 1: The fraction S of vertices verse z.

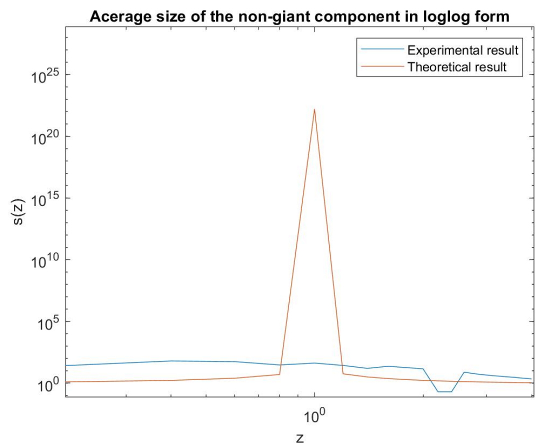

Figure 2: The average size of non-giant component verse z.

2. In the problem, set $z = 4$ so that almost all vertices belong to the giant component. For $n = 2^p$, $p = 10, 11, 12, 13$, generate a random graph $G(n, p)$. For each graph, randomly select $r = 100$ vertices and use each of them as a seed for the BFS. For each graph, numerical solution and theoretical solution are given. The BFS procedure is defined in the matlab code. The theoretical solution for the average length of shortest paths in the Poisson random graph is

$$l(n) \approx \frac{log(n)}{log(z)}.$$

Two results are plotted in the figure 3. Here are some comments on the observations.

- The numerical solution can achieve a good performance and is pretty close to the numerical solution. The slope of two curves are almost identical which can illustrate that numerical solution is correct.
- Equation (29) in the lecture note allows us to estimate the average shortest path length in the giant component of the random graph provided that there is one. Assume that the giant component embraces alomost all the vertices in the graph. We assume that $z_2 \gg z_1$ and set the expected size of lth neighborhood to be equal to the whole graph n:

$$z_l = [\frac{z_2}{z_1}]^{l-1}$$

For here, we obtain the following estimate for the average shortest path length:

$$l \approx 1 + \frac{log(n/z_1)}{log(z_2/z_1)}$$

It is illustrated that $l(n)$ has the same slope with $\frac{\log(n)}{\log(z)}$.

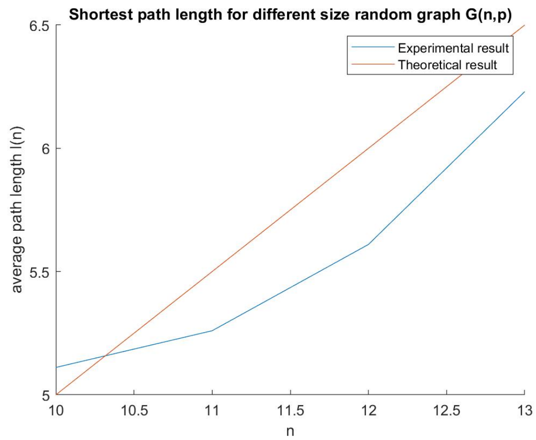

Figure 3: Average size of shortest path verse n.

3. In this problem, a detailed derivation of the formula for the average size of the component to which v belongs to:

$$\langle s \rangle = 1 + \frac{zu^2}{[1-S][1-G_1'(u)]}$$

is given as following. Supposed that for a unibartite undirected graph, define the generating function $G_0(x)$ for the probability distribution of vertex degrees k as

$$G_0(x) = \sum_{k=0}^{\infty} p_k x^k,$$

where p_k is the probability that a randomly chosen vertex on the graph has degree k. The distribution p_k is assumed correctly normalized, so that

$$G_0(1) = 1.$$

Based on the property of function $G_0(x)$: Moments, the average over the probability distribution generated by a generating function is given by

$$z = \langle k \rangle = \sum_k k p_k = G_0'(1).$$

If we start at a randomly chosen vertex and follow each of the edges at that vertex to reach the k nearest neighbors, then the vertices arrived at each have the distribution of remaining outgoing edges generated by this function, less one power of x. to allow for the edge that we arrived along. Thus the distribution of outgoing edges is generated by the function

$$G_1(x) = \frac{G_0'(x)}{G_0'(1)} = \frac{1}{z} G_0'(x),$$

where z is the average vertex degree. Then we need consider the distribution of the sizes of connected components in the graph. Let $H_1(x)$ be the generating function for the distribution of the sizes of components that are reached by choosing a random edge and following it to one of its ends. If we denote by q_k the probability that the initial site has k edges coming out of it other than the edge we came in along, then making use of the "powers" probability , $H_1(x)$ must satisfy a self-consistency condition of the form

$$H_1(x) = xq_0 + xq_1 H_1(x) + xq_2[H_1(x)]^2 + \cdots .$$

And q_k is the coefficient of x^k in the generation function $G_1(x)$, and above equation can be written as

$$H_1(x) = xG_1(H_1(x))$$

$$H_0(x) = xG_0(H_1(x))$$

The average size of the component to which a randomly chosen vertex belongs, for the case where there is no giant component in the graph, is given in the normal fashion by

$$\langle s \rangle = H_0'(1) = 1 + G_0'(1)H_1'(1).$$

And we have

$$H_1'(1) = 1 + G_1'(1)H_1'(1)$$

The generating function formalism still works when there is a giant component in the graph, $H_0(x)$ then generates the probability distribution of the sizes of components excluding the giant component. This means that $H_0(1)$ is no longer unity, as it is for the other generation functions considered so farm but instead takes the value $1 - S$, where S is the fraction of the graph occupied by tge giant component.

$$H_0(1) = 1 - S$$

$$S = 1 - G_0(u),$$

where $u \equiv H_1(1)$ is the smallest non-negative real solution of

$$u = G_1(u).$$

The general expression for the average component size, excluding the giant component, if there is one, is

$$\begin{aligned}\langle s \rangle = \frac{H_0'(1)}{H_0(1)} &= \frac{1}{H_0(1)}[G_0(H_1(1)) + \frac{G_0'(H_1(1)G_1(H_1(1)))}{1 - G_1'(H(1))}] \\ &= 1 + \frac{zu^2}{[1 - S][1 - G_1'(u)]}\end{aligned}$$