Random Graphs

Optional Project – See The Window

Ground Rules. You can work as a group with at most three people. The project would be submitted by the group. Submission should be by April 12. Your results must be described in a clear understandable way.

Take \( n \) vertices 1, \ldots, \( n \), initially with no edges. Each step select two random vertexes \( x, y \) and add the edge \( \{x, y\} \). Use a Union-Find data structure to keep track of the components and their sizes and (see below) their complexities. At step \( e \) (so now the graph has \( e \) edges) parametrize

\[
\frac{e}{\binom{n}{2}} = \frac{1}{n} + \frac{\lambda}{n^{4/3}}
\]

Stop the process at (say) \( \lambda = -4, -3, -2, -1, 0, 1, 2, 3, 4 \). For each \( \lambda \) give the top ten component sizes both raw and when divided by \( n^{2/3} \).

What will happen (if you did it correctly!) is that at \( \lambda = -4 \) (barely subcritical) there is an “asteroid belt” with the top sizes fairly close together. During the critical window these asteroids are merging and by \( \lambda = +4 \) (barely supercritical) a “Jupiter” has appeared, a component whose size is substantially larger than the others.

The complexity of a component is edges minus vertices plus one. (Trees are zero, unicyclic one, etc.) When two components merge the complexity of the new component is simply the sum of the old complexities. When an edge \( \{x, y\} \) is added with \( x, y \) in the same component, its complexity is incremented by one. What will happen (if you did it correctly!) is that at \( \lambda = -4 \) the complexities of the large components are pretty much all zero but that at \( \lambda = +4 \) the Jupiter will have a moderately large complexity, the other complexities pretty much all zero.

Repeated Edges. Occasionally you may select \( \{x, y\} \) that has already been selected. Bummer. But it turns out that this has a negligible effect. So simply ignore it, process this \( \{x, y\} \) the same as the others and implement \( e \) the same as before.

Open-Ended. Come up with some interesting data on the process in this region. Read Chapter 11 for ideas. Surprise me!

Movie. If you are a graphics type – how about a movie of what’s going on! How many edges \( n = 10^6 \), if well programmed, should be good. You can compare different \( n \) and many runs.