PART IV

NETWORK MODELS

CHAPTER 12

RANDOM GRAPHS

An introduction to the most basic of network models, the random graph

SO FAR in this book we have looked at how we measure the structure of networks and at mathematical, statistical, and computational methods for making sense of the network data we get from our measurements. We have seen for instance how to measure the structure of the Internet, and once we have measured it how to determine its degree distribution, or the centrality of its vertices, or the best division of the network into groups or communities. An obvious next question to ask is, "If I know a network has some particular property, such as a particular degree distribution, what effect will that have on the wider behavior of the system?" It turns out that properties like degree distributions can in fact have huge effects on networked systems, which is one of the main reasons we are interested in them. And one of the best ways to understand and get a feel for these effects is to build mathematical models. The remainder of this book is devoted to the examination of some of the many network models in common use.

In Chapters 12 to 15 we consider models of the structure of networks, models that mimic the patterns of connections in real networks in an effort to understand the implications of those patterns. In Chapters 16 to 19 we consider models of processes taking place on networks, such as epidemics on social networks or search engines on the Web. In many cases these models of network processes are themselves built on top of our models of network structure, combining the two to shed light on the interplay between structure and dynamics in networked systems.

In Section 8.4, for instance, we noted that many networks have degree distributions that roughly follow a power law—the so-called scale-free networks. A reasonable question would be to ask how the structure and behavior of such scale-free networks differs from that of their non-scale-free counterparts. A good way to address this question would be to create, on a computer for example, two artificial networks, one with a power-law degree distribution and one without, and explore their differences empirically. Better still, one could create a large number of networks in each of the two classes, to see what statistically significant features appear in one class and not in the other. This is precisely the rationale behind random graph models, which are the topic of this chapter and the following one. In random graph models, one creates networks that possess particular properties of interest, such as specified degree distributions, but which are otherwise random. Random graphs are interesting in their own right for the light they shed on the structural properties of networks, but have also been widely used as a substrate for models of dynamical processes *on* networks. In Chapter 17, for instance, we examine their use in epidemic modeling.

We also look at a number of other types of network model in succeeding chapters. In Chapter 14 we look at generative models of networks, models in which the network is "grown" according to a specified set of growth rules. Generative models are particularly useful for understanding how network structure arises in the first place. By growing networks according to a variety of different rules and comparing the results with real networks, we can get a feel for which growth processes are plausible and which can be ruled out. In Chapter 15 we look at "small-world models," which model the phenomenon of network transitivity or clustering (see Section 7.9), and at "exponential random graphs," which are particularly useful when we want to create model networks that match the properties of observed networks as closely as possible.

12.1 RANDOM GRAPHS

In general, a *random graph* is a model network in which some specific set of parameters take fixed values, but the network is random in other respects. One of the simplest examples of a random graph is the network in which we fix only the number of vertices n and the number of edges m. That is, we take n vertices and place m edges among them at random. More precisely, we choose m pairs of vertices uniformly at random from all possible pairs and connect them with an edge. Typically one stipulates that the network should be a simple graph, i.e., that it should have no multiedges or self-edges (see Section 6.1), in which case the position of each edge should be chosen among only those pairs that are distinct and not already connected.[176] This model is often referred to by its mathematical name $G(n, m)$.

Another entirely equivalent definition of the model is to say that the network is created by choosing uniformly at random among the set of all simple graphs with exactly n vertices and m edges.

Strictly, in fact, the random graph model is not defined in terms of a single randomly generated network, but as an *ensemble* of networks, i.e., a probability distribution over possible networks. Thus the model $G(n, m)$ is correctly defined as a probability distribution $P(G)$ over all graphs G in which $P(G) = 1/\Omega$ for simple graphs with n vertices and m edges and zero otherwise, where Ω is the total number of such simple graphs. We will see more complicated examples of random graph ensembles shortly.

When one talks about the properties of random graphs one typically means the average properties of the ensemble. For instance, the "diameter" of $G(n, m)$ would mean the diameter $\ell(G)$ of a graph G, averaged over the ensemble thus

$$\langle \ell \rangle = \sum_G P(G)\ell(G) = \frac{1}{\Omega}\sum_G \ell(G).$$

(12.1)

This is a useful definition for a several of reasons. First, it turns out to lend itself well to analytic calculations; many such average properties of random graphs can be calculated exactly, at least in the limit of large graph size. Second, it often reflects exactly the thing we want to get at in making our model network in the first place. Very often we are interested in the typical properties of networks. We might want to know, for instance, what the typical diameter is of a network with a given number of edges. Certainly there are special cases of such networks that have particularly large or small diameters, but these don't reflect the typical behavior. If it's typical behavior we are after, then the ensemble average of a property is often a good guide. Third, it can be shown that the distribution of values for many network measures is sharply peaked, becoming concentrated more and more narrowly around the ensemble average as the size of the network becomes large, so that in the large n limit essentially all values one is likely to encounter are very close to the mean.

Some properties of the random graph $G(n, m)$ are straightforward to calculate: obviously the average number of edges is m, for instance, and the average degree is $\langle k \rangle = 2m/n$. Unfortunately, other properties are not so easy to calculate, and most mathematical work has actually been

conducted on a slightly different model that is considerably easier to handle. This model is called $G(n, p)$. In $G(n, p)$ we fix not the number but the *probability* of edges between vertices. Again we have n vertices, but now we place an edge between each distinct pair with independent probability p. In this network the number of edges is not fixed. Indeed it is possible that the network could have no edges at all, or could have edges between every distinct pair of vertices. (For most values of p these are not likely outcomes, but they could happen.)

Again, the technical definition of the random graph is not in terms of a single network, but in terms of an ensemble, a probability distribution over all possible networks. To be specific, $G(n, p)$ is the ensemble of networks with n vertices in which each simple graph G appears with probability where m is the number of edges in the graph, and non-simple graphs have probability zero.

$$P(G) = p^m(1-p)^{\binom{n}{2}-m},$$

(12.2)

$G(n, p)$ was first studied, to this author's knowledge, by Solomonoff and Rapoport [303], but it is most closely associated with the names of Paul Erdős and Alfréd Rényi, who published a celebrated series of papers about the model in the late 1950s and early 1960s [105-107]. If you read scientific papers on this subject, you will sometimes find the model referred to as the "Erdős-Rényi model" or the "Erdős-Rényi random graph" in honor of their contribution. It is also sometimes called the "Poisson random graph" or the "Bernoulli random graph," names that refer to the distributions of degrees and edges in the model. And sometimes the model is referred to simply as "the" random graph—there are many random graph models, but $G(n, p)$ is the most fundamental and widely studied of them, so if someone is talking about a random graph but doesn't bother to mention which one, they are probably thinking of this one.

In this chapter we describe the basic mathematics of the random graph $G(n, p)$, focusing particularly on the degree distribution and component sizes, which are two of the model's most illuminating characteristics. The techniques we develop in this chapter will also prove useful for some of the more complex models examined later in the book.

12.2 MEAN NUMBER OF EDGES AND MEAN DEGREE

Let us start our study of the random graph $G(n, p)$ with a very simple calculation, the calculation of the expected number of edges in our model network. We have said that the number of edges in the model is not fixed, but we can calculate its mean or expectation value as follows. The number of graphs with exactly n vertices and m edges is equal to the number of ways of picking the positions of the edges from the $\binom{n}{2}$ distinct vertex pairs. Each of these graphs appears with the same probability $P(G)$, given by Eq. (12.2), and hence the total probability of drawing a graph with m edges from our ensemble is

$$P(m) = \binom{\binom{n}{2}}{m} p^m (1-p)^{\binom{n}{2}-m},$$

(12.3)

which is just the standard binomial distribution. Then the mean value of m is

$$\langle m \rangle = \sum_{m=0}^{\binom{n}{2}} m P(m) = \binom{n}{2} p.$$

(12.4)

This result comes as no surprise. The expected number of edges between any individual pair of vertices is just equal to the probability p of an edge between the same vertices, and Eq. (12.4) thus says merely that the expected total number of edges in the network is equal to the expected number p between any pair of vertices, multiplied by the number of pairs.

We can use this result to calculate the mean degree of a vertex in the random graph. As pointed out in the previous section, the mean degree in a graph with exactly m edges is $k = 2m/n$, and hence the mean degree in $G(n, p)$ is

$$\langle k \rangle = \sum_{m=0}^{\binom{n}{2}} \frac{2m}{n} P(m) = \frac{2}{n}\binom{n}{2} p = (n-1)p,$$

(12.5)

where we have used Eq. (12.4) and the fact that n is constant. The mean degree of a random

graph is often denoted c in the literature, and we will adopt this convention here also, writing

$$c = (n-1)p.$$

(12.6)

This result is also unsurprising. It says that the expected number of edges connected to a vertex is equal to the expected number p between the vertex and any other vertex, multiplied by the number $n - 1$ of other vertices.

12.3 DEGREE DISTRIBUTION

Only slightly more taxing is the calculation of the degree distribution of $G(n, p)$. A given vertex in the graph is connected with independent probability p to each of the $n - 1$ other vertices. Thus the probability of being connected to a particular k other vertices and not to any of the others is $p^k(1 - p)^{n-1-k}$. There are $\binom{n-1}{k}$ ways to choose those k other vertices, and hence the total probability of being connected to exactly k others is

$$p_k = \binom{n-1}{k} p^k (1-p)^{n-1-k},$$

(12.7)

which is a binomial distribution again. In other words, $G(n, p)$ has a binomial degree distribution.

In many cases we are interested in the properties of large networks, so that n can be assumed to be large. Furthermore, as discussed in Section 6.9, many networks have a mean degree that is approximately constant as the network size becomes large. (For instance, the typical number of friends a person has does not depend strongly on the total number of people in the world.) In such a case Eq. (12.7) simplifies as follows.

Equation (12.6) tells us that $p = c/(n - 1)$ will become vanishingly small as $n \to \infty$, which allows us to write

$$\begin{aligned} \ln\left[(1-p)^{n-1-k}\right] &= (n-1-k)\ln\left(1-\frac{c}{n-1}\right) \\ &\simeq -(n-1-k)\frac{c}{n-1} \simeq -c, \end{aligned}$$

(12.8)

where we have expanded the logarithm as a Taylor series, and the equalities become exact as $n \to \infty$. Taking exponentials of both sizes, we thus find that $(1 - p)^{n-1-k} = e^{-c}$ in the large-n limit. Also for large n we have

$$\binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)!\,k!} \simeq \frac{(n-1)^k}{k!},$$

(12.9)

and thus Eq. (12.7) becomes

$$p_k = \frac{(n-1)^k}{k!} p^k \mathrm{e}^{-c} = \frac{(n-1)^k}{k!} \left(\frac{c}{n-1}\right)^k \mathrm{e}^{-c} = \mathrm{e}^{-c} \frac{c^k}{k!},$$

(12.10)

in the limit of large n.

Equation (12.10) is the Poisson distribution: in the limit of large n, $G(n, p)$ has a Poisson degree distribution. This is the origin of the name *Poisson random graph*, which we will use occasionally to distinguish this model from some of the more sophisticated random graphs in the following chapter that don't in general have Poisson degree distributions.

12.4 CLUSTERING COEFFICIENT

A very simple quantity to calculate for the Poisson random graph is the clustering coefficient. Recall that the clustering coefficient C is a measure of the transitivity in a network (Section 7.9) and is defined as the probability that two network neighbors of a vertex are also neighbors of each other. In a random graph the probability that *any* two vertices are neighbors is exactly the same—all such probabilities are equal to $p = c/(n-1)$. Hence

$$C = \frac{c}{n-1}, \tag{12.11}$$

This is one of several respects in which the random graph differs sharply from most from real-world networks, many of which have quite high clustering coefficients—see Table 8.1—while Eq. (12.11) tends to zero in the limit $n \to \infty$ if the mean degree c stays fixed. This discrepancy is discussed further in Section 12.8.

12.5 GIANT COMPONENT

Consider the Poisson random graph $G(n, p)$ for $p = 0$. In this case there are no edges in the network at all and it is completely disconnected. Each vertex is an island on its own; the network has n separate components of exactly one vertex each.

In the opposite limit, when $p = 1$, every possible edge in the network is present and the network is an n-vertex clique in the technical sense of the word (see Section 7.8.1) meaning that every vertex is connected directly to every other. In this case, all the vertices are connected together in a single component that spans the entire network.

Now let us focus on the size of the largest component in the network in each of these cases. In the first case ($p = 0$) the largest component has size 1. In the second ($p = 1$) the largest component has size n. Apart from the second being much larger than the first, there is an important qualitative difference between these two cases: in the first case the size of the largest component is independent of the number of vertices n in the network; in the second it is proportional to n, or *extensive* in the jargon of theoretical physics. In the first case, the largest component will stay the same size if we make the network larger, but in the second it will grow with the network.

The distinction between these two cases is an important one. In many applications of networks it is crucial that there be a component that fills most of the network. For instance, in the Internet it is important that there be a path through the network from most computers to most others. If there were not, the network wouldn't be able to perform its intended role of providing computer-to-computer communications for its users. Moreover, as discussed in Section 8.1, most networks do in fact have a large component that fills most of the network. We can gain some useful insights about what is happening in such networks by considering how the components in our random graph behave. Although the random graph is a very simple network model and doesn't provide an accurate representation of the Internet or other real-world networks, we will see that when trying to understand the world it can be very helpful to study such simplified models.

So let us consider the largest component of our random graph, which, as we have said, has constant size 1 when $p = 0$ and extensive size n when $p = 1$. An interesting question to ask is how the transition between these two extremes occurs if we construct random graphs with gradually increasing values of p, starting at 0 and ending up at 1. We might guess, for instance, that the size of the largest component somehow increases gradually with p, becoming extensive only in the limit where $p = 1$. In reality, however, something much more interesting happens. As we will see, the size of the largest component undergoes a sudden change, or *phase transition*, from constant size to extensive size at one particular special value of p. Let us take a look at this transition.

A network component whose size grows in proportion to n we call a *giant component*. We can calculate the size of the giant component in the Poisson random graph exactly in the limit of large network size $n \to \infty$ as follows. We denote by u the average fraction of vertices in the random graph that do *not* belong to the giant component. Thus if there is no giant component in our graph, we will have $u = 1$, and if there is a giant component we will have $u < 1$. Alternatively, we can regard u as the probability that a randomly chosen vertex in the graph does not belong to the giant component.

For a vertex i not to belong to the giant component it must not be connected to the giant component via any other vertex. That means that for every other vertex j in the graph either (a) i is not connected to j by an edge, or (b) i is connected to j but j is itself not a member of the giant component. The probability of outcome (a) is simply $1 - p$, the probability of not having an edge between i and j, and the probability of outcome (b) is pu, where the factor of p is the probability of having an edge and the factor of u is the probability that vertex j doesn't belong to the giant component.[177] Thus the total probability of not being connected to the giant component via vertex j

is $1 - p + pu$.

Then the total probability of not being connected to the giant component via any of the $n - 1$ other vertices in the network is

$$u = (1 - p + pu)^{n-1} = \left[1 - \frac{c}{n-1}(1-u)\right]^{n-1},$$

(12.12)

where we have used Eq. (12.6). Now we take logs of both sides thus:

$$\begin{aligned} \ln u &= (n-1)\ln\left[1 - \frac{c}{n-1}(1-u)\right] \\ &\simeq -(n-1)\frac{c}{n-1}(1-u) = -c(1-u), \end{aligned}$$

(12.13)

where the approximate equality becomes exact in the limit of large n. Taking exponentials of both sides, we then find that

$$u = \mathrm{e}^{-c(1-u)}.$$

(12.14)

But if u is the fraction of vertices not in the giant component, then the fraction of vertices that are in the giant component is $S = 1 - u$. Eliminating u in favor of S then gives us

$$S = 1 - \mathrm{e}^{-cS}.$$

(12.15)

This equation, which was first given by Erdős and Rényi in 1959 [105], tells us the size of the giant component as a fraction of the size of the network in the limit of large network size, for any given value of the mean degree c. Unfortunately, though the equation is very simple it doesn't have a simple solution for S in closed form.[178] We can however get a good feeling for its behavior from a graphical solution. Consider Fig. 12.1. The three curves show the function $y = 1 - \mathrm{e}^{-cS}$ for different values of c. Note that S can take only values from zero to one, so only this part of the curve is shown. The dashed line in the figure is the function $y = S$. Where line and curve cross we have $S = 1 - \mathrm{e}^{-cS}$ and the corresponding value of S is a solution to Eq. (12.15).

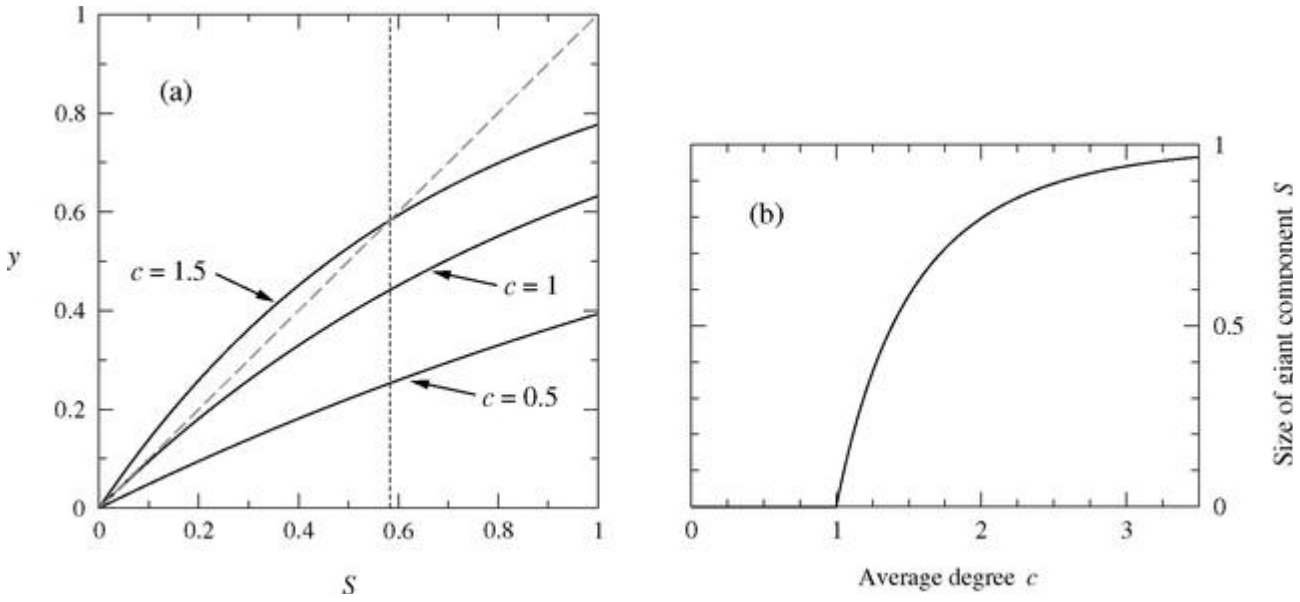


Figure 12.1: Graphical solution for the size of the giant component. (a) The three curves in the left panel show $y = 1 - e^{-cS}$ for values of c as marked, the diagonal dashed line shows $y = S$, and the intersection gives the solution to Eq. (12.15), $S = 1 - e^{-cS}$. For the bottom curve there is only one intersection, at $S = 0$, so there is no giant component, while for the top curve there is a solution at $S = 0.583$ ... (vertical dashed line). The middle curve is precisely at the threshold between the regime where a non-trivial solution for S exists and the regime where there is only the trivial solution $S = 0$. (b) The resulting solution for the size of the giant component as a function of c.

As the figure shows, depending on the value of c there may be either one solution for S or two. For small c (bottom curve in the figure) there is just one solution at $S = 0$, which implies that there is no giant component in the network. (You can confirm for yourself that $S = 0$ is a solution directly from Eq. (12.15).) On the other hand, if c is large enough (top curve) then there are two solutions, one at $S = 0$ and one at $S > 0$. Only in this regime can there be a giant component.

The transition between the two regimes corresponds to the middle curve in the figure and falls at the point where the gradient of the curve and the gradient of the dashed line match at $S = 0$. That is, the transition takes place when

$$\frac{\mathrm{d}}{\mathrm{d}S}(1 - \mathrm{e}^{-cS}) = 1,$$

(12.16)

or

$$c\mathrm{e}^{-cS} = 1.$$

(12.17)

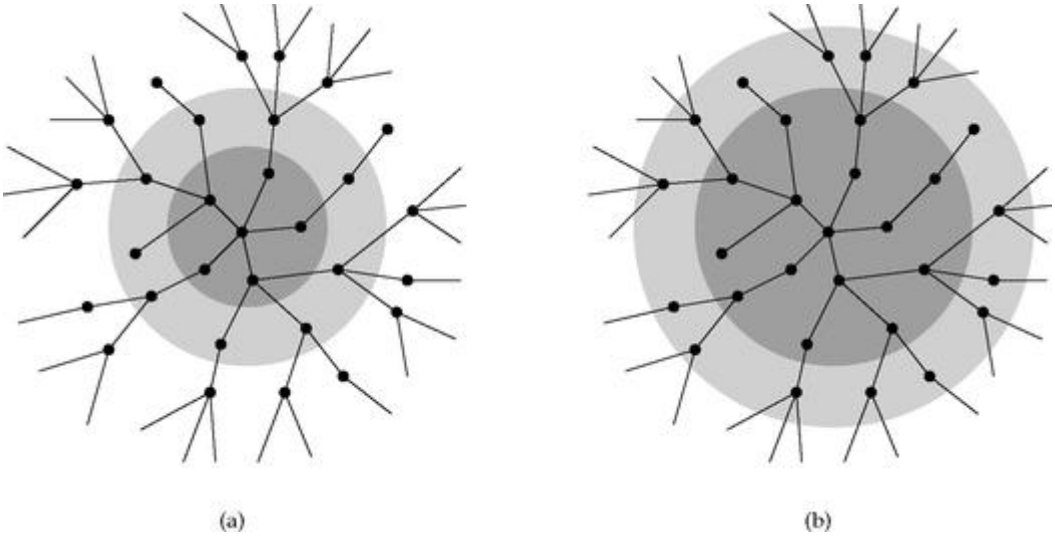


Figure 12.2: Growth of a vertex set in a random graph. (a) A set of vertices (inside the gray circles) consists of a core (dark gray) and a periphery (lighter). (b) If we grow the set by adding to it those vertices immediately adjacent to the periphery, then the periphery vertices become a part of the new core and a new periphery is added.

Setting $S = 0$ we then deduce that the transition takes place at $c = 1$.

In other words, the random graph can have a giant component only if $c > 1$. At $c = 1$ and below we have $S = 0$ and there is no giant component.

This does not entirely solve the problem, however. Technically we have proved that there can be no giant component for $c \leq 1$, but not that there has to be a giant component at $c > 1$—in the latter regime there are two solutions for S, one of which is the solution $S = 0$ in which there is no giant component. So which of these solutions is the correct one that describes the true size of the giant component?

In answering this question, we will see another way to think about the formation of the giant component. Consider the following process. Let us find a small set of connected vertices somewhere in our network—say a dozen or so, as shown in Fig. 12.2a. In the limit of large $n \to \infty$ such a set is bound to exist somewhere in the network, so long as $c > 0$. We will divide the set into its *core* and its *periphery*. The core is the vertices that have connections only to other vertices in the set—the darker gray region in the figure. The *periphery* is the vertices that have at least one neighbor outside the set—the lighter gray.

Now imagine enlarging our set by adding to it all those vertices that are immediate neighbors, connected by at least one edge to the set—Fig. 12.2b. Now the old periphery is part of the core and there is a new periphery consisting of the vertices just added. How big is this new periphery? We don't know for certain, but we know that each vertex in the old periphery is connected with independent probability p to every other vertex. If there are s vertices in our set, then there are $n - s$ vertices outside the set, and the average number of connections a vertex in the periphery has to outside vertices is where the equality becomes exact in the limit $n \to \infty$. This means that the average number of immediate neighbors of the set—the size of the new periphery when we grow the set—is c times the size of the old periphery.

$$p(n-s) = c\frac{n-s}{n-1} \simeq c,$$

(12.18)

We can repeat this argument, growing the set again and again, and each time the average size of the periphery will increase by another factor of c. Thus if $c > 1$ the average size of the periphery will grow exponentially. On the other hand, if $c < 1$ it will shrink exponentially and eventually dwindle to zero. Furthermore, if it grows exponentially our connected set of vertices will eventually form a component comparable in size to the whole network—a giant component—while if it dwindles the set will only ever have finite size and no giant component will form.

So we see that indeed we expect a giant component if (and only if) $c > 1$. And when there is a giant component the size of that giant component will be given by the larger solution to Eq. (12.15). This now allows us to calculate the size of the giant component for all values of c. (For $c > 1$ we have to solve for the larger solution of Eq. (12.15) numerically, since there is no exact solution, but this is easy enough to do.) The results are shown in Fig. 12.1. As the figure shows, the size of the giant component grows rapidly from zero as the value of c passes 1, and tends towards $S = 1$ as c becomes large.

12.6 SMALL COMPONENTS

In this section we look at the properties of random graphs from a different point of view, the point of view of the non-giant components. We have seen that in a random graph with $c > 1$ there exists a giant component that fills an extensive fraction of the network. That fraction is typically less than 100%, however. What is the structure of the remainder of the network? The answer is that it is made up of many small components whose average size is constant and doesn't increase with the size of the network.

The first step in demonstrating this result and shedding light on the structure of the small components is to show that there is only one giant component in a random graph, and hence that all other components are "non-giant" components. This is fairly easy to establish. Suppose that there were two or more giant components in a random graph. Take any two giant components, which have size $S_1 n$ and $S_2 n$, where S_1 and S_2 are the fractions of the network filled by each. The number of distinct pairs of vertices (i, j), where i is in the first giant component and j is in the second, is just $S_1 n \times S_2 n = S_1 S_2 n^2$. Each of these pairs is connected by an edge with probability p, or not with probability $1 - p$. For the two giant components to be separate components we require that there be zero edges connecting them together, which happens with probability q given by

$$q = (1-p)^{S_1 S_2 n^2} = \left(1 - \frac{c}{n-1}\right)^{S_1 S_2 n^2},$$

(12.19)

where we have made use of Eq. (12.6).

Taking logs of both sides and going to the limit $n \to \infty$, we then find

$$\begin{aligned} \ln q &= S_1 S_2 \lim_{n\to\infty} \left[n^2 \ln\left(1 - \frac{c}{n-1}\right)\right] = S_1 S_2 \left[-c(n+1) + \tfrac{1}{2}c^2\right] \\ &= c S_1 S_2 \left[-n + (\tfrac{1}{2}c - 1)\right], \end{aligned}$$

(12.20)

where we have dropped terms of order $1/n$. Taking the exponential again, we get

$$q = q_0 \, \mathrm{e}^{-c S_1 S_2 n},$$

(12.21)

where $q_0 = e^{c(c/2-1)S_1S_2}$, which is independent of n if c is constant. Thus, for constant c, the probability that the two giant components are really separate components dwindles exponentially with increasing n, and in the limit of large n will vanish altogether. In a large random graph, therefore, there is only the very tiniest of probabilities that we will have two giant components, and for infinite n the probability is formally zero and it will never happen.

Given then that there is only one giant component in our random graph and that in most situations it does not fill the entire network, it follows that there must also be some non-giant components, i.e., components whose size does not increase in proportion to the size of the network. These are the *small components*.

12.6.1 SIZES OF THE SMALL COMPONENTS

The small components can, in general, come in various different sizes. We can calculate the distribution of these sizes as follows.

The basic quantity we focus on is the probability π_s that a randomly chosen vertex belongs to a small component of size exactly s vertices total. Note that if there is a giant component in our network then some vertices do not belong to a small component of any size and hence π_s is not normalized to unity. The sum of π_s over all sizes s is equal to the fraction of vertices that are not in the giant component. That is,

$$\sum_{s=0}^{\infty} \pi_s = 1 - S,$$

(12.22)

where S is, as before, the fraction of vertices in the giant component.

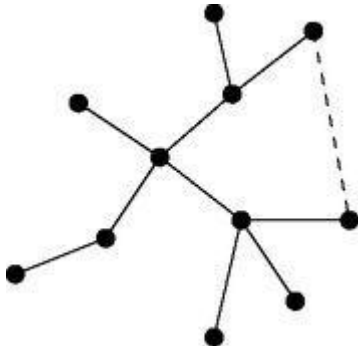
Recall that a tree is a graph or subgraph that has no loops—see Section 6.7.

The crucial insight that allows us to calculate π_s is that the small components are trees, as we can see by the following argument. Consider a small component of s vertices that takes the form of a tree. A tree of s vertices contains $s - 1$ edges, as shown in Section 6.7, and this is the smallest number of edges that is needed to connect this many vertices together. If we add another edge to our component then we will create a loop, since we will be adding a new path between two vertices that are already connected (see figure). In a Poisson random graph the probability of such edge being present is the same as for any other edge, $p = c/(n - 1)$. The total number of places where we could add such an extra edge to the component is given by the number of distinct pairs of vertices minus the number that are already connected by an edge, or

$$\binom{s}{2} - (s-1) = \tfrac{1}{2}(s-1)(s-2),$$

(12.23)

and the total number of extra edges in the component is $\frac{1}{2}(s-1)(s-2)c/(n-1)$. Assuming that s increases more slowly than $\sqrt{n}$ (and we will shortly see that it does), this probability tends to zero in the limit $n \to \infty$, and hence there are no loops in the component and the component is a tree.



If we add an edge (dashed) to a tree we create a loop.

We can use this observation to calculate the probability π_s as follows. Consider a vertex i in a small component of a random graph, as depicted in Fig. 12.3. Each of i's edges leads to a separate subgraph—the shaded regions in the figure—and because the whole component is a tree we know that these subgraphs are not connected to one another, other than via vertex i, since if they were there would be a loop in the component and it would not be a tree. Thus the size of the component to which i belongs is the sum of the sizes of the subgraphs reachable along each of its edges, plus 1 for vertex i itself. To put that another way, vertex i belongs to a component of size s if the sizes of the subgraphs to which its neighbors n_1, n_2, . . . belong sum to $s - 1$.

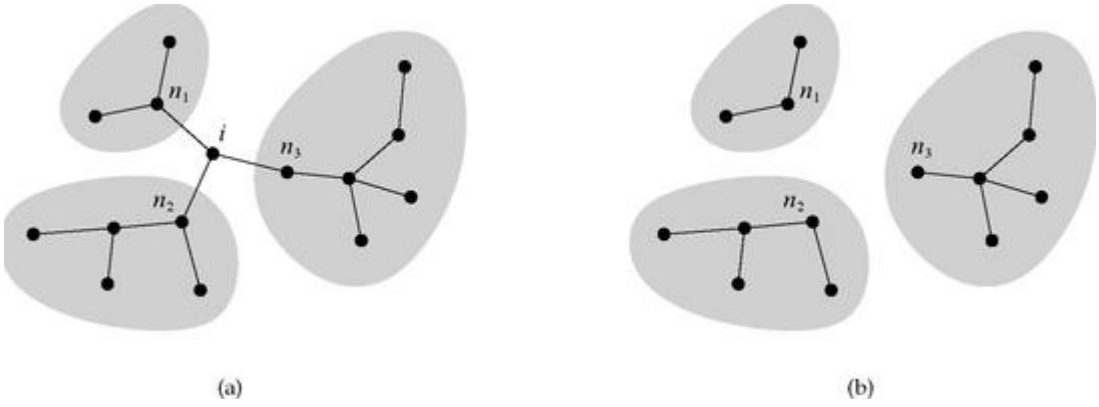


Figure 12.3: The size of one of the small components in a random graph. (a) The size of the component to which a vertex i belongs is the sum of the number of vertices in each of the subcomponents (shaded regions) reachable via i's neighbors n_1, n_2, n_3, plus one for i itself. (b) If vertex i is removed the subcomponents become components in their own right.

Bearing this in mind, consider now a slightly modified network, the network in which vertex i is completely removed, along with all its edges.[179] This network is still a random graph with the same value of p—each possible edge is still present with independent probability p—but the number of vertices has decreased by one, from n to $n - 1$. In the limit of large n, however, this decrease is negligible. The average properties, such as size of the giant component and size of the small components will be indistinguishable for random graphs with sizes n and n - 1, but the same p.

In this modified network, what were previously the subgraphs of our small component are now

separate small components in their own right. And since the network has the same average properties as the original network for large n, that means that the probability that neighbor n_1 belongs to a small component of size s_1 (or a subgraph of size s_1 in the original network) is itself given by π_{s_1}. We can use this observation to develop a self-consistent expression for the probability π_s.

Suppose that vertex i has degree k. As we have said, the probability that neighbor n_1 belongs to a small component of size s_1 when i is removed from the network is π_{s_1} . So the probability $P(s|k)$ that vertex i belongs to a small component of size s, given that its degree is k, is the probability that its k neighbors belong to small components of sizes s , ... ,s_k—which is $\prod_{j=1}^{k} \pi_{s_j}$—and that those sizes add up to $s - 1$:

$$P(s|k) = \sum_{s_1=1}^{\infty} \cdots \sum_{s_k=1}^{\infty} \left[\prod_{j=1}^{k} \pi_{s_j} \right] \delta\left(s - 1, \sum_j s_j\right),$$

(12.24)

where $\delta(m,n)$ is the Kronecker delta.

To get π_s, we now just average $P(s|k)$ over the distribution p_k of the degree thus:

$$\begin{aligned} \pi_s &= \sum_{k=0}^{\infty} p_k P(s|k) = \sum_{k=0}^{\infty} p_k \sum_{s_1=1}^{\infty} \cdots \sum_{s_k=1}^{\infty} \left[\prod_{j=1}^{k} \pi_{s_j} \right] \delta\left(s - 1, \sum_j s_j\right) \\ &= \mathrm{e}^{-c} \sum_{k=0}^{\infty} \frac{c^k}{k!} \sum_{s_1=1}^{\infty} \cdots \sum_{s_k=1}^{\infty} \left[\prod_{j=1}^{k} \pi_{s_j} \right] \delta\left(s - 1, \sum_j s_j\right), \end{aligned}$$

(12.25)

where we have made use of Eq. (12.10) for the degree distribution of the random graph.

This expression would be easy to evaluate if it were not for the delta function: one could separate the terms in the product, distribute them among the individual summations, and complete the sums in closed form. With the delta function, however, it is difficult to see how the sum can be completed.

Luckily there is a trick for problems like these, a trick that we will use many times in the rest of this book. We introduce a *generating function* or *z-transform*, defined by

$$h(z) = \pi_1 z + \pi_2 z^2 + \pi_3 z^3 + \ldots = \sum_{s=1}^{\infty} \pi_s z^s.$$

(12.26)

This generating function is a polynomial or series in z whose coefficients are the probabilities π_s .

It encapsulates all of the information about the probability distribution in a single function. Given $h(z)$ we can recover the probabilities by differentiating:

$$\pi_s = \frac{1}{s!}\left.\frac{\mathrm{d}^s h}{\mathrm{d}z^s}\right|_{z=0}.$$

(12.27)

Thus $h(z)$ is a complete representation of our probability distribution and if we can calculate it, then we can calculate π_s. We will look at generating functions in more detail in the next section, but for now let us complete the present calculation.

We can calculate $h(z)$ by substituting Eq. (12.25) into Eq. (12.26), which gives

$$\begin{aligned} h(z) &= \sum_{s=1}^{\infty} z^s \mathrm{e}^{-c} \sum_{k=0}^{\infty} \frac{c^k}{k!} \sum_{s_1=1}^{\infty} \cdots \sum_{s_k=1}^{\infty} \left[\prod_{j=1}^{k} \pi_{s_j}\right] \delta\bigl(s-1, \textstyle\sum_j s_j\bigr) \\ &= \mathrm{e}^{-c} \sum_{k=0}^{\infty} \frac{c^k}{k!} \sum_{s_1=1}^{\infty} \cdots \sum_{s_k=1}^{\infty} \left[\prod_{j=1}^{k} \pi_{s_j}\right] z^{1+\sum_j s_j} \\ &= z\mathrm{e}^{-c} \sum_{k=0}^{\infty} \frac{c^k}{k!} \sum_{s_1=1}^{\infty} \cdots \sum_{s_k=1}^{\infty} \left[\prod_{j=1}^{k} \pi_{s_j} z^{s_j}\right] \\ &= z\mathrm{e}^{-c} \sum_{k=0}^{\infty} \frac{c^k}{k!} \left[\sum_{s=1}^{\infty} \pi_s z^s\right]^k = z\mathrm{e}^{-c} \sum_{k=0}^{\infty} \frac{c^k}{k!} \bigl[h(z)\bigr]^k \\ &= z \exp\bigl[c\bigl(h(z)-1\bigr)\bigr]. \end{aligned}$$

(12.28)

Thus we have a simple, self-consistent equation for $h(z)$ that eliminates the awkward delta function of (12.25).

Unfortunately, like the somewhat similar Eq. (12.15), this equation doesn't have a known closed-form solution for $h(z)$, but that doesn't mean the expression is useless. In fact we can calculate many useful things from it without solving for $h(z)$ explicitly. For example, we can calculate the mean size of the component to which a randomly chosen vertex belongs, which is given by

$$\langle s \rangle = \frac{\sum_s s\pi_s}{\sum_s \pi_s} = \frac{h'(1)}{1-S},$$

(12.29)

where $h'(z)$ denotes the first derivative of $h(z)$ with respect to its argument and we have made

use of Eqs. (12.22) and (12.26). (The denominator in this expression is necessary because π_s is not normalized to 1.)

From Eq. (12.28) we have

$$\begin{aligned} h'(z) &= \exp[c(h(z)-1)] + czh'(z)\exp[c(h(z)-1)] \\ &= \frac{h(z)}{z} + ch(z)h'(z), \end{aligned}$$

(12.30)

or, rearranging,

$$h'(z) = \frac{h(z)}{z[1-ch(z)]},$$

(12.31)

and thus

$$h'(1) = \frac{h(1)}{1-ch(1)}.$$

(12.32)

But $h(1) = \sum_s \pi_s = 1 - S$, from Eqs. (12.22) and (12.26), so that

$$h'(1) = \frac{1-S}{1-c+cS}.$$

(12.33)

And so the average size s of Eq. (12.29) becomes

$$\langle s \rangle = \frac{1}{1-c+cS}.$$

(12.34)

When $c < 1$ and there is no giant component, this gives simply $\langle s \rangle = 1/(1-c)$. When there is a giant component, the behavior is more complicated, because we have to solve for S first before finding the value of $\langle s \rangle$, but the calculation can still be done. We first solve Eq. (12.15) for S and then substitute into Eq. (12.34).

It's interesting to note that Eq. (12.34) diverges when $c = 1$. (At this point $S = 0$, so the denominator vanishes.) Thus, if we slowly increase the mean degree c of our network from some small initial value less than 1, the average size of the component to which a vertex belongs gets bigger and bigger and finally becomes infinite exactly at the point where the giant component appears. For $c > 1$ Eq. (12.34) measures only the sizes of the non-giant components and the equation tells us that these get smaller again above $c = 1$. Thus the general picture we have is in one in which the small components get larger up to $c = 1$, where they diverge and the giant component appears, then smaller again as the giant component grows larger. Figure 12.4 shows a plot of $\langle s \rangle$ as a function of c with the divergence clearly visible.

Although the random graph is certainly not a realistic model of most networks, this general picture of the component structure of the network turns out to be a good guide to the behavior of networks in the real world. If a network has a low density of edges then typically it consists only of small components, but if the density is becomes enough then a single large component forms, usually accompanied by many separate small ones. Moreover, the small components tend on average to be smaller if the largest component is very large. This is a good example of the way in which simple models of networks can give us a feel for how more complicated real-world systems should behave in general.

12.6.2 AVERAGE SIZE OF A SMALL COMPONENT

A further important point to notice about Eq. (12.34) is that the average size of the small components does not grow with the number of vertices n. The typical size of the small components in a random graph remains constant as the graph gets larger. We must, however, be a little careful with these statements. Recall that π_s is the probability that a randomly chosen vertex belongs to a component of size s, and hence s as calculated here is not strictly the average size of a component, but the average size of the component to which a randomly chosen vertex belongs. Because larger components have more vertices in them, the chances of landing on them when we choose a random vertex is larger, in proportion to their size, and hence s is a biased estimate of the actual average component size. To get a correct figure for the average size of a component we need to make a slightly different calculation.

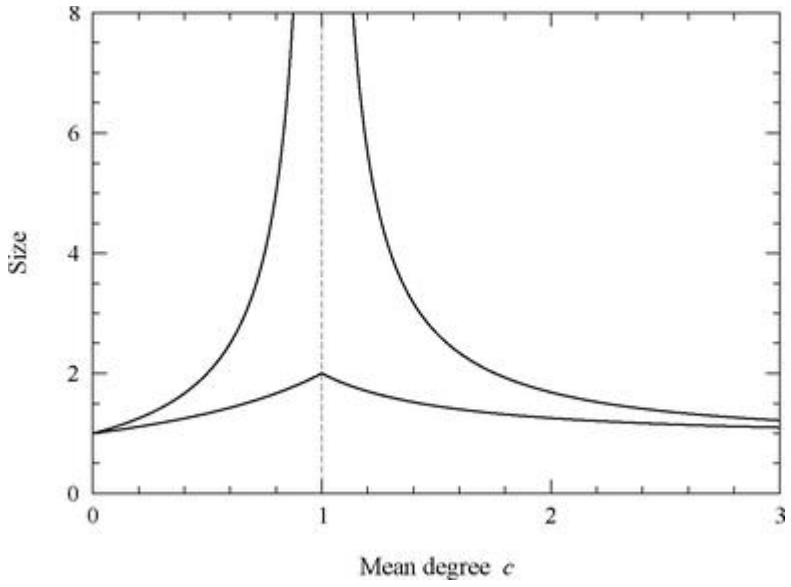


Figure 12.4: Average size of the small components in a random graph. The upper curve shows the average size s of the component to which a randomly chosen vertex belongs, calculated from Eq. (12.34). The lower curve shows the overall average size R of a component, calculated from Eq. (12.40). The dotted vertical line marks the point $c = 1$ at which the giant component appears. Note that, as discussed in the text, the upper curve diverges at this point but the lower one does not.

Let n_s be the actual number of components of size s in our random graph. Then the number of vertices that belong to components of size s is sn_s and hence the probability of a randomly chosen vertex belonging to such a component is

$$\pi_s = \frac{sn_s}{n},$$

(12.35)

The average size of a component, which we will denote R, is

$$R = \frac{\sum_s s n_s}{\sum_s n_s} = \frac{n \sum_s \pi_s}{n \sum_s \pi_s / s} = \frac{1 - S}{\sum_s \pi_s / s},$$

(12.36)

where we have made use of Eq. (12.22). The remaining sum we can again evaluate using our generating function by noting that

$$\int_0^1 \frac{h(z)}{z}\,\mathrm{d}z = \sum_{s=1}^{\infty} \pi_s \int_0^1 z^{s-1}\,\mathrm{d}z = \sum_{s=1}^{\infty} \frac{\pi_s}{s}.$$

(12.37)

A useful expression for $h(z)/z$ can be obtained by rearranging Eq. (12.31) to yield

$$\frac{h(z)}{z} = [1 - ch(z)]\frac{\mathrm{d}h}{\mathrm{d}z},$$

(12.38)

and hence we find that

$$\begin{aligned}\sum_{s=1}^{\infty} \frac{\pi_s}{s} &= \int_0^1 [1 - ch(z)]\frac{\mathrm{d}h}{\mathrm{d}z}\mathrm{d}z = \int_0^{1-S} (1 - ch)\,\mathrm{d}h \\ &= 1 - S - \tfrac{1}{2}c(1 - S)^2,\end{aligned}$$

(12.39)

where we have used $h(1) = \sum_s \pi_s = 1 - S$ for the upper integration limit.

Substituting this result into Eq. (12.36), we find that the average component size is

$$R = \frac{2}{2 - c + cS}.$$

(12.40)

As with Eq. (12.34), this expression is independent of n, so the average size of a small component indeed does not grow as the graph becomes large.

On the other hand, R does not diverge at $c = 1$ as s does. At $c = 1$, with $S = 0$, Eq. (12.40) gives just $R = 2$. The reason for this is that, while the largest component in the network for $c = 1$ does become infinite in the limit of large n, so also does the total number of components. So the average size of a component is the ratio of two diverging quantities. Depending on the nature of the divergences, such a ratio could be infinite itself, or zero, or finite but non-zero in the special case where the two divergences have the same asymptotic form. In this instance the latter situation holds—both quantities are diverging linearly with n—and the average component size remains finite. A plot of R is included in Fig. 12.4 for comparison with s .

12.6.3 THE COMPLETE DISTRIBUTION OF COMPONENT SIZES

So far we have calculated the average size of a small component in the random graph, but not the individual probabilities π_s that specify the complete distribution of sizes. In principle, we should be able to calculate the π_s by solving Eq. (12.28) for the generating function $h(z)$ and then differentiating according to Eq. (12.27) to get π_s . Unfortunately we cannot follow this formula in practice because, as mentioned above, Eq. (12.28) does not have a known solution.

Remarkably, however, it turns out that we can still calculate the values of the individual π_s, by an alternative route. The calculations involve some more advanced mathematical techniques and if you are not particularly interested in the details it will do no harm to skip this section. If you're interested in this rather elegant development, however, read on.

To calculate an explicit expression for the probabilities π_s of the component sizes we make use of a beautiful result from the theory of complex variables, the *Lagrange inversion formula*. The Lagrange inversion formula is a formula that allows the explicit solution of equations of the form

$$f(z) = z\phi(f(z))$$

(12.41)

for the unknown function $f(z)$, where $\varphi(f)$ is a known function which at $f = 0$ is finite, non-zero, and differentiable.

Equation (12.41) has precisely the form of the equation for our generating function, Eq. (12.28). What's more, the Lagrange formula gives a solution for $f(z)$ in terms of the coefficients of the series expansion of $f(z)$ in powers of z, which is precisely what we want in the present case, since the coefficients are the probabilities π_s , which is what we want to calculate. The Lagrange formula is thus perfectly suited to the problem in hand. Here we first derive the general form of the formula then apply it to the current problem.[180]

Let us write the function $f(z)$ in Eq. (12.41) as a series expansion thus:

$$f(z) = \sum_{s=1}^{\infty} a_s z^s,$$

(12.42)

The coefficient a_s in this expansion is given explicitly by

$$a_s = \frac{1}{s!}\frac{d^s f}{dz^s}\bigg|_{z=0} = \frac{1}{s!}\left[\frac{d^{s-1}}{dz^{s-1}}\left(\frac{df}{dz}\right)\right]_{z=0}.$$

(12.43)

Cauchy's formula for the nth derivative of a function $g(z)$ at $z = z_0$ says that

$$\left.\frac{\mathrm{d}^n g}{\mathrm{d}z^n}\right|_{z=z_0} = \frac{n!}{2\pi \mathrm{i}} \oint \frac{g(z)}{(z-z_0)^{n+1}}\,\mathrm{d}z,$$

(12.44)

where the integral is around a contour that encloses z_0 in the complex plane but encloses no poles in $g(z)$. We will use an infinitesimal circle around z_0 as our contour.

Applying Cauchy's formula to (12.43) with $g(z) = f'(z)$, $z_0 = 0$, and $n = s - 1$, we get

$$a_s = \frac{1}{2\pi \mathrm{i} s} \oint \frac{1}{z^s} \frac{\mathrm{d}f}{\mathrm{d}z}\,\mathrm{d}z = \frac{1}{2\pi \mathrm{i} s} \oint \frac{\mathrm{d}f}{z^s},$$

(12.45)

where the second integral is now around a contour in f rather than z. In this equation we are now thinking of z as being a function of f, $z = z(f)$, rather than the other way around. We are perfectly entitled to do this—knowing either quantity specifies the value of the other.[181]

It will be important later that the contour followed by f surrounds the origin, so let us pause for a moment to demonstrate that it does. Our choice of contour for z in the first integral of Eq. (12.45) is an infinitesimal circle around the origin. Expanding Eq. (12.41) to leading order around the origin, we find that

$$f(z) = z\phi(f(0)) + \mathrm{O}(z^2) = z\phi(0) + \mathrm{O}(z^2),$$

(12.46)

where we have made use of the fact that $f(0) = 0$, which is easily seen from Eq. (12.41) given that $\varphi(f)$ is non-zero and finite at $f = 0$ by hypothesis. In the limit of small $|z|$ where the terms of order z^2 can be neglected, Eq. (12.46) implies that f traces a contour about the origin if z does, since the two are proportional to one another.

We now rearrange our original equation, Eq. (12.41), to give the value of z in terms of f thus

$$z(f) = \frac{f}{\phi(f)},$$

(12.47)

and then substitute into Eq. (12.45) to get

$$a_s = \frac{1}{2\pi i s} \oint \frac{[\phi(f)]^s}{f^s}\, df.$$

(12.48)

Since, as we have said, the contour encloses the origin, this expression can be written in terms of a derivative evaluated at the origin by again making use of Cauchy's formula, Eq. (12.44):

$$a_s = \frac{1}{s!}\left[\frac{d^{s-1}}{df^{s-1}}[\phi(f)]^s\right]_{f=0}.$$

(12.49)

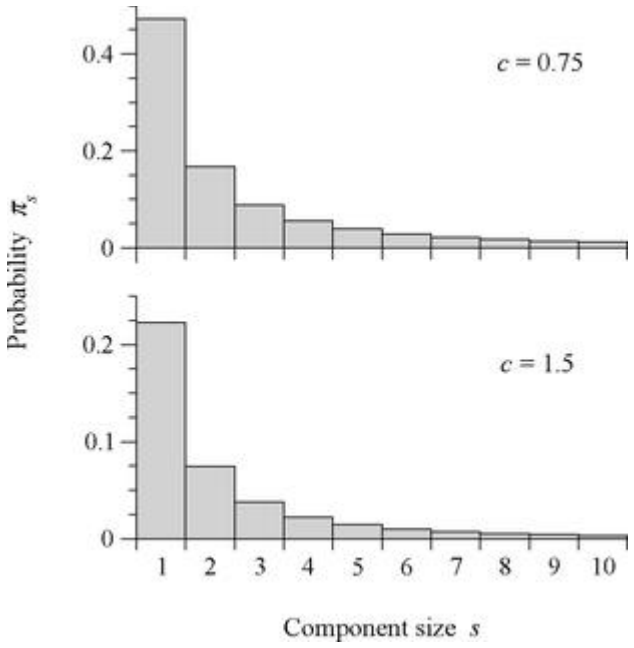


Figure 12.5: Sizes of small components in the random graph. This plot shows the probability π_s that a randomly chosen vertex belongs to a small component of size s in a Poisson random graph with $c = 0.75$ (top), which is in the regime where there is no giant component, and $c = 1.5$ (bottom), where there is a giant component.

This is the Lagrange inversion formula. This remarkably simple formula gives us, in effect, a complete series solution to Eq. (12.41).

To apply the formula to the current problem, of the component size distribution for the random graph, we set $f(z) \to h(z)$ and $\varphi(f) \to e^{c(h-1)}$. Then the coefficients π_s of $h(z)$ are given by

$$\pi_s = \frac{1}{s!}\left[\frac{d^{s-1}}{dh^{s-1}} e^{sc(h-1)}\right]_{h=0} = \frac{e^{-sc}(sc)^{s-1}}{s!}.$$

(12.50)

These are the probabilities that a randomly chosen vertex belongs to a small component of size s in a random graph with mean degree c. Figure 12.5 shows the shape of π_s as a function of s for two different values of c. As the plot shows, the distribution is heavily skewed, with many components of small size and only a few larger ones.

12.7 PATH LENGTHS

In Sections 3.6 and 8.2 we discussed the small-world effect, the observation that the typical lengths of paths between vertices in networks tend to be short. Most people find the small-world effect surprising upon first learning about it. We can use the random graph model to shed light on how the effect arises by examining the behavior of the network diameter in the model.

See Section 6.10.1 for a discuss of geodesic distances and diameters.

Recall that the diameter of a network is the longest geodesic distance between any two vertices in the same component of the network. As we now show, the diameter of a random graph varies with the number n of vertices as $\ln n$. Since $\ln n$ is typically a relatively small number even when n is large, this offers some explanation of the small-world effect, although it also leaves some questions open, as discussed further below.

The basic idea behind the estimation of the diameter of a random graph is simple. As discussed in Section 12.5, the average number of vertices s steps away from a randomly chosen vertex in a random graph is c^s. Since this number grows exponentially with s it doesn't take very many such steps before the number of vertices reached is equal to the total number of vertices in the whole network; this happens when $c^s \simeq n$ or equivalently $s \simeq \ln n / \ln c$. At this point, roughly speaking, every vertex is within s steps of our starting point, implying that the diameter of the network is approximately $\ln n / \ln c$.

Although the random graph is, as we have said, not an accurate model of most real-world networks, this is, nonetheless, believed to be the basic mechanism behind the small-world effect in most networks: the number of vertices within distance s of a particular starting point grows exponentially s and hence the diameter is logarithmic in n. We discuss the comparison with real-world networks in more detail below.

The argument above is only approximate. It's true that there are on average c^s vertices s steps away from any starting point so long as s is small. But once c^s becomes comparable with n the result has to break down since clearly the number of vertices at distance s cannot exceed the number of vertices in the whole graph. (Indeed it cannot exceed the number in the giant component.)

One way to deal with this problem is to consider two different starting vertices i and j. The average numbers of vertices s and t steps from them respectively will then be equal to c^s and c^t so long as we stay in the regime where both these numbers are much less than n. In the following calculation we consider only configurations in which both remain smaller than order n in the limit $n \to \infty$ so as to satisfy this condition.

The situation we consider is depicted in Fig. 12.6, with the two vertices i and j each surrounded by a "ball" or neighborhood consisting of all vertices with distances up to and including s and t respectively. If there is an edge between the "surface" (i.e., most distant vertices) of one neighborhood and the surface of the other, as depicted by the dashed line, then it is straightforward to show that there is also an edge between the surfaces of any pair of neighborhoods with larger s or t (or both). Turning that statement around, if there is no edge between the surfaces of our neighborhoods, then there is also no edge between any smaller neighborhoods, which means that the shortest path between i and j must have length greater than $s + t + 1$. The reverse is also trivially true, that a shortest path longer than $s + t + 1$ implies there is no edge between our surfaces. Thus the absence of an edge between the surfaces is a necessary and sufficient condition for the distance d_{ij} between i and j to be greater than $s + t + 1$. This in turn implies that the probability $P(d_{ij} > s + t + 1)$ is equal to the probability that there is no edge between the two

surfaces.

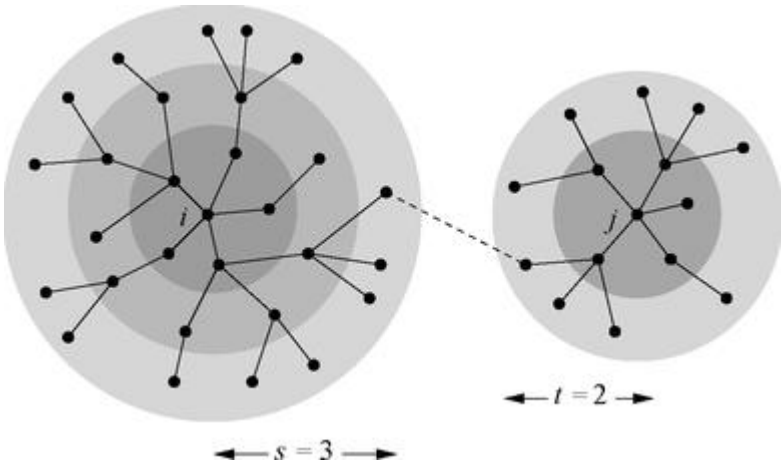


Figure 12.6: Neighborhoods of two vertices in a random graph. In the argument given in the text we consider the sets of vertices within distances s and t respectively of two randomly chosen vertices i and j. If there is an edge between any vertex on the surface of one neighborhood and any vertex on the surface of the other (dashed line), then there is a path between i and j of length $s + t + 1$.

There are on average $c^s \times c^t$ pairs of vertices such that one lies on each surface, and each pair is connected with probability $p = c/(n - 1) \simeq c/n$ (assuming n to be large) or not with probability 1 - p. Hence $P(d_{ij} > s + t + 1) = (1 - p)^{c^{s+1}}$. Defining for convenience $\ell = s + t + 1$, we can also write this as

$$P(d_{ij} > \ell) = (1 - p)^{c^{\ell-1}} = \left(1 - \frac{c}{n}\right)^{c^{\ell-1}}.$$

(12.51)

Taking logs of both sides, we find

$$\ln P(d_{ij} > \ell) = c^{\ell-1} \ln\left(1 - \frac{c}{n}\right) \simeq -\frac{c^\ell}{n},$$

(12.52)

where the approximate inequality becomes exact as $n \to \infty$. Thus in this limit

$$P(d_{ij} > \ell) = \exp\left(-\frac{c^\ell}{n}\right).$$

(12.53)

The diameter of the network is the smallest value of l such that $P(d_{ij} > l)$ is zero, i.e., the value such that no matter which pair of vertices we happen to pick there is zero chance that they will be separated by a greater distance. In the limit of large n, Eq. (12.53) will tend to zero only if c^ℓ grows faster than n, meaning that our smallest value of ℓ is the value such that $c^\ell = an^{1+\in}$ with a constant and $\in \to 0$ from above. Note that we can, as promised, achieve this while keeping both c^s and c^t smaller than order n, so that our argument remains valid.

Rearranging for $\in l$, we now find our expression for the diameter:

$$\ell = \frac{\ln a}{\ln c} + \lim_{\epsilon \to 0} \frac{(1+\epsilon)\ln n}{\ln c} = A + \frac{\ln n}{\ln c},$$

(12.54)

where A is a constant.[182] Apart from the constant, this is the same result as we found previously using a rougher argument. The constant is known—it has a rather complicated value in terms of the Lambert W-function [114]—but for our purposes the important point is that it is (asymptotically) independent of n. Thus the diameter indeed increases only slowly with n, as $\ln n$, making it relatively small in large random graphs.

The logarithmic dependence of the diameter on n offers some explanation of the small-world effect of Section 3.6. Even in a network such as the acquaintance network of the entire world, with nearly seven billion inhabitants (at the time of writing), the value of $\ln n/\ln c$ can be quite small. Supposing each person to have about a thousand acquaintances,[183] we would get

$$\ell = \frac{\ln n}{\ln c} = \frac{\ln 6 \times 10^9}{\ln 1000} = 3.3\ldots,$$

(12.55)

which is easily small enough to account for the results of, for example, the small-world experiments of Milgram and others [93, 219, 311].

On the other hand, although this calculation gives us some insight into the nature of the small-world effect, this cannot be the entire explanation. There are clearly many things wrong with the random graph as a model of real social networks, as we now discuss.

12.8 PROBLEMS WITH THE RANDOM GRAPH

The Poisson random graph is one of the best studied models of networks. In the half century since its first proposal it has given us a tremendous amount of insight into the expected structure of networks of all kinds, particularly with respect to component sizes and network diameters. The fact that it is both simple to describe and straightforward to study using analytic methods makes it an excellent tool for investigating all sorts of network phenomena. We will return to the random graph many times in the remainder of this book to help us understand the way networks behave.

The random graph does, however, have some severe shortcomings as a network model. There are many ways in which it is completely unlike the real-world networks we have seen in the previous chapters. One clear problem is that it shows essentially no transitivity or clustering. In Section 12.4 we saw at the clustering coefficient of a random graph is $C = c/(n-1)$, which tends to zero in the limit of large n. And even for the finite values of n appropriate to real-world networks the value of C in the random graph is typically very small. For the acquaintance network of the human population of the world, with its $n \simeq 7$ billion people, each having about 1000 acquaintances [175], a random graph with the same n and c would have a clustering coefficient of

$$C \simeq \frac{1000}{7\,000\,000\,000} \simeq 10^{-7}. \tag{12.56}$$

Whether the clustering coefficient of the real acquaintance network is 0.01 or 0.5 hardly matters. (It is probably somewhere in between.) Either way it is clear that the random graph and the true network are in strong disagreement.[184]

The random graph also differs from real-world networks in many other ways. For instance, there is no correlation between the degrees of adjacent vertices—necessarily so, since the edges are placed completely at random. The degrees in real networks, by contrast, are usually correlated, as discussed in Section 8.7. Many, perhaps most, real-world networks also show grouping of their vertices into "communities," as discussed on Section 11.2.1, but random graphs have no such structure. And there are many other examples of interesting structure in real networks that is absent from the random graph.

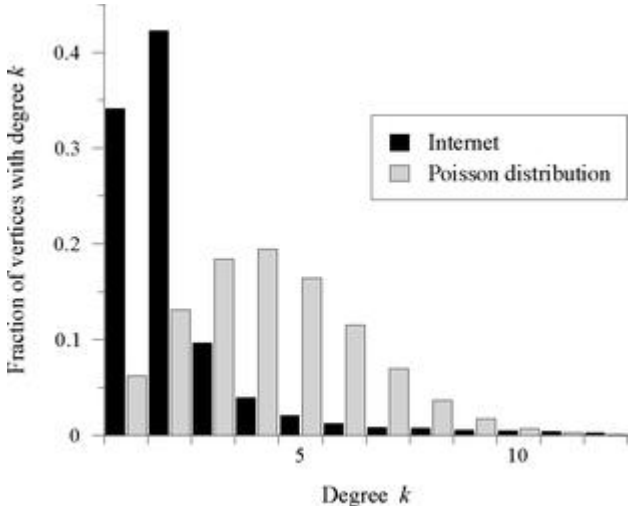


Figure 12.7: Degree distribution of the Internet and a Poisson random graph. The dark bars in this plot show the fraction of vertices with the given degrees in the network representation of the Internet at the level of autonomous systems. The lighter bars represent the same measure for a random graph with the same average degree as the Internet. Even though the two distributions have the same averages, it is clear that they are entirely different in shape.

However, perhaps the most significant respect in which the properties of random graphs diverge from those of real-world networks is the shape of their degree distribution. As discussed in Section 8.3, real networks typically have right-skewed degree distributions, with most vertices having low degree but with a small number of high-degree "hubs" in the tail of the distribution. The random graph on the other hand has a Poisson degree distribution, Eq. (12.10), which is not right-skewed to any significant extent. Consider Fig. 12.7, for example, which shows a histogram of the degree distribution of the Internet (darker bars), measured at the level of autonomous systems (Section 2.1.1). The right-skewed form is clearly visible in this example. On the same figure we show the Poisson degree distribution of a random graph (lighter bars) with the same average degree c as the Internet example. Despite having the same averages, the two distributions are clearly entirely different. It turns out that this difference has a profound effect on all sorts of properties of the network—we will see many examples in this book. This makes the Poisson random graph inadequate to explain many of the interesting phenomena we see in networks today, including resilience phenomena, epidemic spreading processes, percolation, and many others.

Luckily it turns out to be possible to generalize the random graph model to allow for non-Poisson degree distributions. This development, which leads to some of the most beautiful results in the mathematics of networks, is described in the next chapter.

PROBLEMS

12.1 Consider the random graph $G(n, p)$ with mean degree c.

a. Show that in the limit of large n the expected number of triangles in the network is $\frac{1}{6}c^3$. This means that the number of triangles is constant, neither growing nor vanishing in the limit of large n.
b. Show that the expected number of connected triples in the network (as defined on page 200) is $\frac{1}{2}nc^2$.
c. Hence calculate the clustering coefficient C, as defined in Eq. (7.41), and confirm that it agrees for large n with the value given in Eq. (12.11).

12.2 Consider the random graph $G(n, p)$ with mean degree c.

a. Argue that the probability that a vertex of degree k belongs to a small component is $(1 - S)^k$, where S is the fraction of the network occupied by the giant component.
b. Thus, using Bayes' theorem (or otherwise) show that the fraction of vertices in small components that have degree k is $e^{-c}c^k (1 - S)^{k-1}/k!$.

12.3 Starting from the generating function $h(z)$ defined in Eq. (12.26), or otherwise, show that

a. the mean-square size of the component in a random graph to which a randomly chosen vertex belongs is $1/(1 - c)^3$ in the regime where there is no giant component;
b. the mean-square size of a randomly chosen component in the same regime is $1/[(1-c)(1-\frac{1}{2}c)]$.

Note that both quantities diverge at the phase transition where the giant component appears.

12.4 In Section 7.8.2 we introduced the idea of a bicomponent. A vertex in a random graph belongs to a bicomponent if two or more of its neighbors belong to the giant component of the network (since the giant component completes a loop between those neighbors forming a bicomponent). In principle, a vertex can also be in a bicomponent if two or more of its neighbors belong to the same small component, but in practice this never happens, since that would imply that the small component in question contained a loop and, as we have seen, the small components in a random graph are trees and so have no loops.

a. Show that the fraction of vertices in a random graph that belong to a bicomponent is $S_2 = (1 - cu)(1 - u)$, where u is defined by Eq. (12.14).
b. Show that this expression can be rewritten as $S_2 = S + (1 - S)\ln(1 - S)$, where S is the size of the giant component.
c. Hence argue that the random graph contains a giant bicomponent whenever it contains an ordinary giant component.

12.5 The *cascade model* is a simple mathematical model of a directed acyclic graph, sometimes used to model food webs. We take n vertices labeled $i = 1 \ldots n$ and place an undirected edge between each distinct pair with independent probability p, just as in the ordinary random graph. Then we add directions to the edges such that each edge runs from the vertex with numerically higher label to the vertex with lower label. This ensures that all directed paths in the network run from higher to lower labels and hence that the network is acyclic, as discussed in Section 6.4.2.

a. Show that the average in-degree of vertex i in the ensemble of the cascade model is $\langle k_i^{\text{in}} \rangle = (n-i)p$ and the average out-degree is $\langle k_i^{\text{out}} \rangle = (i-1)p$.
b. Show that the expected number of edges that connect to vertices i and lower from vertices above i is $(ni - i^2)\,p$.
c. Assuming n is even, what are the largest and smallest values of this quantity and where do they occur?

In a food web this expected number of edges from high–to low-numbered vertices is a rough measure of energy flow and the cascade model predicts that energy flow will be largest in the middle portions of a food web and smallest at the top and bottom.

12.6 We can make a simple random graph model of a network with clustering or transitivity as follows. We take n vertices and go through each distinct trio of three vertices, of which there are $\binom{n}{3}$, and with independent probability p we connect the members of the trio together using three edges to form a triangle, where $p = c/\binom{n-1}{2}$ with c a constant.

a. Show that the mean degree of a vertex in this model network is $2c$.
b. Show that the degree distribution is

$$p_k = \begin{cases} \mathrm{e}^{-c} c^{k/2}/(k/2)! & \text{if } k \text{ is even,} \\ 0 & \text{if } k \text{ is odd.} \end{cases}$$

c. Show that the clustering coefficient, Eq. (7.41), is $C = 1/(2c+1)$.
d. Show that when there is a giant component in the network its expected size S as a fraction of network size satisfies $S = 1 - \mathrm{e}^{-cS(2-S)}$.
e. What is the value of the clustering coefficient when the giant component fills half of the network?

CHAPTER 13

RANDOM GRAPHS WITH GENERAL DEGREE DISTRIBUTIONS

This chapter describes more sophisticated random graph models that mimic networks with arbitrary degree distributions

IN THE previous chapter we looked at the classic random graph model, in which pairs of vertices are connected at random with uniform probabilities. Although this model has proved tremendously useful as a source of insight into the structure of networks, it also has, as described in Section 12.8, a number of serious shortcomings. Chief among these is its degree distribution, which follows the Poisson distribution and is quite different from the degree distributions seen in most real-world networks. In this chapter we show how we can create more sophisticated random graph models, which incorporate arbitrary degree distributions and yet are still exactly solvable for many of their properties in the limit of large network size.

The fundamental mathematical tool that we will use to derive the results of this chapter is the probability generating function. We have already seen in Section 12.6 one example of a generating function, which was useful in the calculation of the distribution of component sizes in the Poisson random graph. We begin this chapter with a more formal introduction to generating functions and to some of their properties which will be useful in later calculations. Readers interested in pursuing the mathematics of generating functions further may like to look at the book by Wilf [329].[185]

13.1 GENERATING FUNCTIONS

Suppose we have a probability distribution for a non-negative integer variable, such that separate instances, occurrences, or draws of this variable are independent and have value k with probability p_k. A good example of such a distribution is the distribution of the degrees of randomly chosen vertices in a network. If the fraction of vertices in a network with degree k is p_k then p_k is also the probability that a randomly chosen vertex from the network will have degree k.

The *generating function* for the probability distribution p_k is the polynomial

$$g(z) = p_0 + p_1 z + p_2 z^2 + p_3 z^3 + \ldots = \sum_{k=0}^{\infty} p_k z^k.$$

(13.1)

Sometimes a function of this kind is called a *probability generating function* to distinguish it from another common type of function, the *exponential generating function*. We will not use exponential generating functions in this book, so for us all generating functions will be probability generating functions.

If we know the generating function for a probability distribution p_k then we can recover the values of p_k by differentiating:

$$p_k = \frac{1}{k!} \frac{\mathrm{d}^k g}{\mathrm{d}z^k}\bigg|_{z=0}.$$

(13.2)

Thus the generating function gives us complete information about the probability distribution and vice versa. The distribution and the generating function are really just two different representations of the same thing. As we will see, it is easier in many cases to work with the generating function than with the probability distribution and doing so leads to many useful new results about networks.

13.1.1 EXAMPLES

Right away let us look at some examples of generating functions. Suppose our variable k takes only the values 0, 1, 2, and 3, with probabilities p_0, p_1, p_2, and p_3, respectively, and no other values. In that case the corresponding generating function would take the form of a cubic polynomial:

$$g(z) = p_0 + p_1 z + p_2 z^2 + p_3 z^3.$$

(13.3)

For instance, if we had a network in which vertices of degree 0, 1, 2, and 3 occupied 40%, 30%, 20%, and 10% of the network respectively then

$$g(z) = 0.4 + 0.3\,z + 0.2\,z^2 + 0.1\,z^3.$$

(13.4)

As another example, suppose that k follows a Poisson distribution with mean c:

$$p_k = \mathrm{e}^{-c}\frac{c^k}{k!}.$$

(13.5)

Then the corresponding generating function would be

$$g(z) = \mathrm{e}^{-c}\sum_{k=0}^{\infty}\frac{(cz)^k}{k!} = \mathrm{e}^{c(z-1)}.$$

(13.6)

Alternatively, suppose that k follows an exponential distribution of the form

$$p_k = C\,\mathrm{e}^{-\lambda k},$$

(13.7)

with $\lambda > 0$. The normalizing constant is fixed by the condition that $\sum_k p_k = 1$, which gives $C = 1 - \mathrm{e}^{-\lambda}$ and hence

$$p_k = (1 - \mathrm{e}^{-\lambda})\,\mathrm{e}^{-\lambda k}.$$

(13.8)

Then

$$g(z) = (1 - \mathrm{e}^{-\lambda}) \sum_{k=0}^{\infty} (\mathrm{e}^{-\lambda} z)^k = \frac{\mathrm{e}^{\lambda} - 1}{\mathrm{e}^{\lambda} - z},$$

(13.9)

so long as $z < \mathrm{e}^{\lambda}$. (If $z \geq \mathrm{e}^{\lambda}$ the generating function diverges. Normally, however, we will be interested in generating functions only in the range $0 \leq z \leq 1$ so, given that $\lambda > 0$ and hence $^{\lambda} > 1$, the divergence at e^{λ} will not be a problem.)

13.1.2 POWER-LAW DISTRIBUTIONS

One special case of particular interest in the study of networks is the power-law distribution. As we saw in Section 8.4, a number of networks, including the World Wide Web, the Internet, and citation networks, have degree distributions that follow power laws quite closely and this turns out to have interesting consequences that set these networks apart from others. To create and solve models of these networks it will be important for us to be able to write down generating functions for power-law distributions.

There are various forms that are used to represent power laws in practice but the simplest choice, which we will use in many of our calculations, is the "pure" power law

$$p_k = C k^{-\alpha},$$

(13.10)

for constant $\alpha > 0$. This expression cannot apply all the way down to $k = 0$, however, or it would diverge. So commonly one stops at $k = 1$. The normalization constant C can then be calculated from the condition that $\sum_k p_k = 1$, which gives

$$C \sum_{k=1}^{\infty} k^{-\alpha} = 1.$$

(13.11)

The sum unfortunately cannot be performed in closed form. It is, however, a common enough sum that it has a name—it is called the *Riemann zeta function*, denoted $\zeta(\alpha)$:

$$\zeta(\alpha) = \sum_{k=1}^{\infty} k^{-\alpha}.$$

(13.12)

Thus we can write $C = 1/\zeta(\alpha)$ and

$$p_k = \begin{cases} 0 & \text{for } k = 0, \\ k^{-\alpha}/\zeta(\alpha) & \text{for } k \geq 1. \end{cases}$$

(13.13)

Although there is no closed-form expression for the zeta function, there exist good numerical methods for calculating its value accurately, and many programming languages and numerical software packages include functions to calculate it.

For this probability distribution the generating function is

$$g(z) = \frac{1}{\zeta(\alpha)} \sum_{k=1}^{\infty} k^{-\alpha} z^k.$$

(13.14)

Again the sum cannot be expressed in closed form, but again it has a name—it is called the *polylogarithm* of z and is denoted $\mathrm{Li}_\alpha(z)$:

$$\mathrm{Li}_\alpha(z) = \sum_{k=1}^{\infty} k^{-\alpha} z^k.$$

(13.15)

Thus we can write

$$g(z) = \frac{\mathrm{Li}_\alpha(z)}{\zeta(\alpha)}.$$

(13.16)

This is not completely satisfactory. We would certainly prefer a closed-form expression as in the case of the Poisson and exponential distributions of Eqs. (13.6) and (13.9). But we can live with it. Enough properties of the polylogarithm and zeta functions are known that we can carry out useful manipulations of the generating function. In particular, since derivatives of our generating functions will be important to us, we note the following useful relation:

$$\frac{\partial \mathrm{Li}_\alpha(z)}{\partial z} = \frac{\partial}{\partial z} \sum_{k=1}^{\infty} k^{-\alpha} z^k = \sum_{k=1}^{\infty} k^{-(\alpha-1)} z^{k-1} = \frac{\mathrm{Li}_{\alpha-1}(z)}{z}.$$

(13.17)

We should note also that in real-world networks the degree distribution does not usually follow a power law over its whole range—the distribution is not a "pure" power law in the sense above. Instead, it typically obeys a power law reasonably closely for values of k above some minimum value $k_{\min}$ but below that point it has some other behavior. In this case the generating function will take the form

$$g(z) = Q_{k_{\min}-1}(z) + C \sum_{k=k_{\min}}^{\infty} k^{-\alpha} z^k,$$

(13.18)

where $Q_n(z) = \sum_{k=0}^{n} p_k z^k$ is a polynomial in z of degree n and C is a normalizing constant. The sum in Eq. (13.18) also has its own name: it is called the *Lerch transcendent*.[186] In the calculations in this book we will stick to the pure power law, since it illustrates nicely the interesting properties of power-law degree distributions and is relatively simple to deal with, but for serious modeling one might sometimes have to use the cut-off form, Eq. (13.18).

13.1.3 NORMALIZATION AND MOMENTS

Let us now look briefly at some of the properties of generating functions that will be useful to us. First of all, note that if we set $z = 1$ in the definition of the generating function, $g(z) = \sum_k p_k z^k$ (Eq. (13.1)), we get

$$g(1) = \sum_{k=0}^{\infty} p_k.$$

(13.19)

If the probability distribution is normalized to unity, $\sum_k p_k = 1$, as are all the examples above, then this immediately implies that

$$g(1) = 1.$$

(13.20)

For most of the generating functions we will look at, this will be true, but not all. As a counter-example, consider the generating function for the sizes of the small components in the Poisson random graph defined in Eq. (12.26). The probabilities π_s appearing in this generating function were the probabilities that a randomly chosen vertex belongs to a small component of size s. If we are in the regime where there is a giant component in the network then not all vertices belong to a small component, and hence the probabilities π_s do not add up to one. In fact, their sum is equal to the fraction of vertices not in the giant component.

The derivative of the generating function $g(z)$ of Eq. (13.1) is

$$g'(z) = \sum_{k=0}^{\infty} k p_k z^{k-1}.$$

(13.21)

(We will use the primed notation $g'(z)$ for derivatives of generating functions extensively in this chapter, as it proves much less cumbersome than the more common notation dg/dz.)

If we set $z = 1$ in Eq. (13.21) we get

$$g'(1) = \sum_{k=0}^{\infty} k p_k = \langle k \rangle,$$

(13.22)

which is just the average value of k. Thus, for example, if p_k is a degree distribution, we can calculate the average degree directly from the generating function by differentiating. This is a very convenient trick. In many cases we will calculate a probability distribution of interest by calculating first its generating function. In principle, we can then extract the distribution itself by applying Eq. (13.2) and so derive any other quantities we want such as averages. But Eq. (13.22) shows us that we don't always have to do this. Some of the quantities we will be interested in can be calculated directly from the generating function without going through any intermediate steps.

In fact, this result generalizes to higher moments of the probability distribution as well. For instance, note that

$$z\frac{\mathrm{d}}{\mathrm{d}z}\left(z\frac{\mathrm{d}g}{\mathrm{d}z}\right) = \sum_{k=0}^{\infty} k^2 p_k z^k,$$

(13.23)

and hence, setting z = 1, we can write

$$\langle k^2 \rangle = \left[\left(z\frac{\mathrm{d}}{\mathrm{d}z}\right)^2 g(z)\right]_{z=1}.$$

(13.24)

It is not hard to show that this result generalizes to all higher moments as well:

$$\langle k^m \rangle = \left[\left(z\frac{\mathrm{d}}{\mathrm{d}z}\right)^m g(z)\right]_{z=1}.$$

(13.25)

This result can also be written as

$$\langle k^m \rangle = \left. \frac{\mathrm{d}^m g}{\mathrm{d}(\ln z)^m} \right|_{z=1}.$$

(13.26)

13.1.4 POWERS OF GENERATING FUNCTIONS

Perhaps the most useful property of generating functions—and the one that makes them important for the study of networks—is the following. Suppose we are given a distribution p_k with generating function $g(z)$. And suppose we have m integers k_i, $i = 1 \ldots m$, which are independent random numbers drawn from this distribution. For instance, they could be the degrees of m randomly chosen vertices in a network with degree distribution p_k. Then the probability distribution of the sum $\sum_{i=1}^{m} k_i$ of those m integers has generating function $[g(z)]^m$. This is a very powerful result and it is worth taking a moment to see how it arises and what it means.

Given that our integers are independently drawn from the distribution p_k, the probability that they take a particular set of values $\{k_i\}$ is simply $\prod_i p_{k_i}$ and the probability π_s that the values drawn add up to a specific sum s is the sum of these probabilities over all sets $\{k_i\}$ that add up to s:

$$\pi_s = \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} \delta\Big(s, \sum_i k_i\Big) \prod_{i=1}^{m} p_{k_i},$$

(13.27)

where $\delta(a, b)$ is the Kronecker delta. Then the generating function $h(z)$ for the distribution π_s is

$$\begin{aligned} h(z) &= \sum_{s=0}^{\infty} \pi_s z^s \\ &= \sum_{s=0}^{\infty} z^s \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} \delta\Big(s, \sum_i k_i\Big) \prod_{i=1}^{m} p_{k_i} \\ &= \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} z^{\sum_i k_i} \prod_{i=1}^{m} p_{k_i} \\ &= \sum_{k_1=0}^{\infty} \cdots \sum_{k_m=0}^{\infty} \prod_{i=1}^{m} p_{k_i} z^{k_i} = \left[\sum_{k=0}^{\infty} p_k z^k\right]^m \\ &= \big[g(z)\big]^m. \end{aligned}$$

(13.28)

Thus, for example, if we know the degree distribution of a network, it is a straightforward matter to calculate the probability distribution of the sum of the degrees of m randomly chosen vertices from that network. This will turn out to be important in the developments that follow.

13.2 THE CONFIGURATION MODEL

Let us turn now to the main topic of this chapter, the development of the theory of random graphs with general degree distributions.

We can turn the random graph of Chapter 12 into a much more flexible model for networks by modifying it so that the degrees of its vertices are no longer restricted to having a Poisson distribution, and in fact it is possible to modify the model so as to give the network any degree distribution we please. Just as with the Poisson random graph, which can be defined in several slightly different ways, there is more than one way to define random graphs with general degree distributions. Here we describe two of them, which are roughly the equivalent of the $G(n, m)$ and $G(n, p)$ random graphs of Section 12.1.

The most widely studied of the generalized random graph models is the *configuration model*. The configuration model is actually a model of a random graph with a given degree *sequence*, rather than degree distribution. That is, the exact degree of each individual vertex in the network is fixed, rather than merely the probability distribution from which those degrees are chosen. This in turn fixes the number of edges in the network, since the number of edges is given by Eq. (6.21) to be $m = \frac{1}{2}\sum_i k_i$. Thus this model is in some ways analogous to G(n, m), which also fixes the number of edges. (It is quite simple, however, to modify the model for cases where only the degree distribution is known and not the exact degree sequence. We describe how this is done at the end of this section.)

> See Section 8.3 for a discussion of the distinction between degree sequences and degree distributions.

Suppose then that we specify the degree k_i that each vertex $i = 1 \ldots n$ in our network is to take. We can create a random network with these degrees as follows. We give each vertex i a total of k_i "stubs" of edges as depicted in Fig. 13.1. There are $\sum_i k_i = 2m$ stubs in total, where m is the total number of edges. Then we choose two of the stubs uniformly at random and we create an edge by connecting them to one another, as indicated by the dashed line in the figure. Then we choose another pair from the remaining $2m - 2$ stubs, connect those, and so on until all the stubs are used up. The end result is a network in which every vertex has exactly the desired degree.

More specifically the end result is a particular *matching* of the stubs, a particular set of pairings of stubs with other stubs. The process above generates each possible matching of stubs with equal probability. Technically the configuration model is defined as the ensemble in which each matching with the chosen degree sequence appears with the same probability (those with any other degree sequence having probability zero), and the process above is a process for drawing networks from the configuration model ensemble.

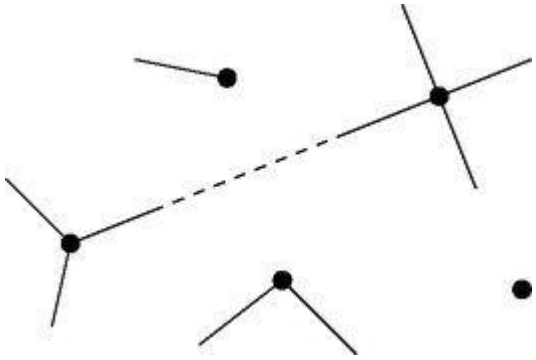


Figure 13.1: The configuration model. Each vertex is given a number of "stubs" of edges equal to its desired degree. Then pairs of stubs are chosen at random and connected together to form edges (dotted line).

The uniform distribution over matchings in the configuration model has the important consequence that any stub in a configuration model network is equally likely to be connected to any other. This, as we will see, is the crucial property that makes the model solvable for many of its properties.

There are a couple of minor catches with the network generation process described here. First, there must be an even number of stubs overall if we want to end up with a network consisting only of vertices and edges, with no dangling stubs left over. This means that the sum $\sum_i k_i$ of the degrees must add up to an even number. We will assume that the degrees we have chosen satisfy this condition, otherwise it is clearly not possible to create a graph with the given degree sequence.

A second issue is that the network may contain self-edges or multiedges, or both. There is nothing in the network generation process that prevents us from creating an edge that connects a vertex to itself or that connects two vertices that are already connected by another edge. One might imagine that one could avoid this by rejecting the creation of any such edges during the process, but it turns out that this is not a good idea. A network so generated is no longer drawn uniformly from the set of possible matchings, which means that properties of the model can no longer be calculated analytically, at least by any means currently known. It can also mean that the network creation process breaks down completely. Suppose, for example, that we come to the end of the process, when there are just two stubs left to be joined, and find that those two both belong to the same vertex so that joining them would create a self-edge. Then either we create the self-edge or the network generation process fails.

In practice, therefore, it makes more sense to allow the creation of both multiedges and self-edges in our networks and the standard configuration model does so. Although some real-world networks have self-edges or multiedges in them, most do not, and to some extent this makes the configuration model less satisfactory as a network model. However, as shown below, the average number of self-edges and multiedges in the configuration model is a constant as the network becomes large, which means that the density of self-edges and multiedges tends to zero in this limit. This means, to all intents and purposes, that we can ignore the self-edges and multiedges in the large size limit.[187]

A further issue with the configuration model is that, while all matchings of stubs appear with equal probability in the model, that does not mean that all *networks* appear with equal probability because more than one matching can correspond to the same network, i.e., the same topological connections between vertices. If we label the stubs to keep track of which is which, then there are typically many different ways we can join up pairs of labeled stubs to create the same final configuration of edges. Figure 13.2 shows an example of a set of eight matchings that all correspond to the same three-vertex network.

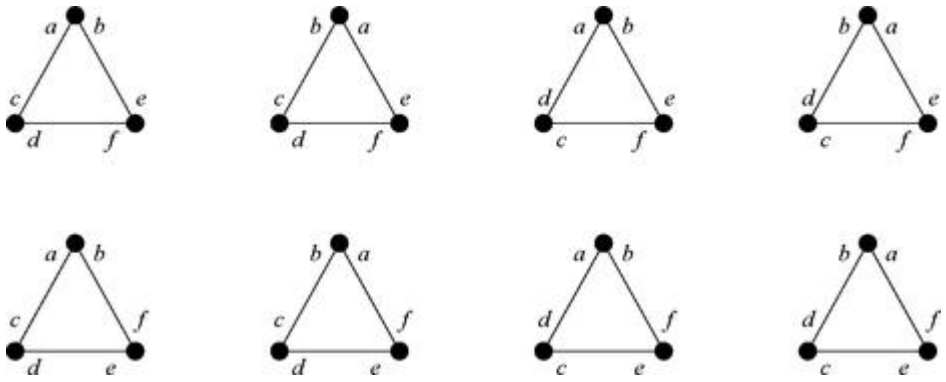


Figure 13.2: Eight stub matchings that all give the same network. This small network is composed of three vertices of degree two and hence having two stubs each. The stubs are lettered to identify them and there are two distinct permutations of the stubs at each vertex for a total of eight permutations overall. Each permutation gives rise to a different matching of stub to stub but all matchings correspond to the same topological configuration of edges, and hence there are eight ways in which this particular configuration can be generated by the stub matching process.

In general, one can generate all the matchings that correspond to a given network by taking any one matching for that network and permuting the stubs at each vertex in every possible way. Since the number of permutations of the k_i stubs at a vertex i is $k_i!$, this implies that the number of matchings corresponding to each network is $N(\{k_i\}) = \prod_i k_i!$, which takes the same value for all networks, since the degrees are fixed. This implies that in fact networks occur with equal probability in the configuration model: if there are $\Omega(\{k_i\})$ matchings, each occurring with the same probability, then each *network* occurs with probability N/Ω.

However, this is not completely correct. If a network contains self-edges or multiedges then not all permutations of the stubs in the network result in a new matching of stubs. Consider Fig. 13.3. Panel (a) shows a network with the same degree sequence as those of Fig. 13.2, but a different matching of the stubs that creates a network with one self-edge and a multiedge consisting of two parallel single edges. In panel (b) we have permuted the stubs *a* and *b* at the ends of the self-edge but, as we can see, this has not resulted in a new matching of the stubs themselves. Stubs *a* and *b* are still connected to one another just as they were before. (The network is *drawn* differently now, but in terms of the matching and the topology of the edges nothing has changed from panel (a).) In panel (c) we have identically permuted the stubs at both ends of the multiedge. Again this has no effect on which stubs are matched with which others.

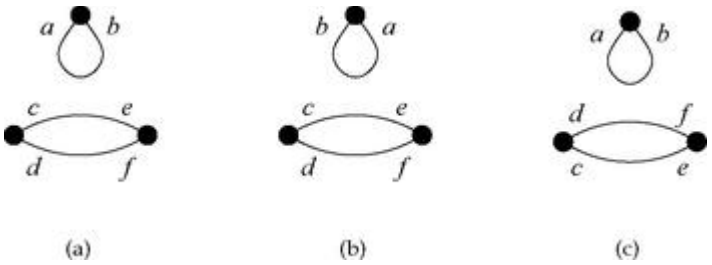


Figure 13.3: Permutations that do not produce new matchings. (a) The network shown here has the same degree sequence as those of Fig. 13.2 but a different configuration of edges, having one self-loop and a multiedge consisting of two parallel edges. (b) If we permute the stubs *a* and *b*

of the self-edge we do not generate a new matching, because a is still matched with b, just as before. (c) If we permute the stubs at either end of a multiedge in exactly the same way we do not generate a new matching, since each stub at one end of the multiedge is still matched with the same stub at the other end.

In general, for each multiedge in a network a permutation of the stubs at one end fails to generate a new matching if we simultaneously permute the stubs at the other end in the same way. This means that the total number of matchings is reduced by a factor of $A_{ij}!$, since A_{ij} is equal to the multiplicity of the edge between i and j. Indeed, this expression is correct even for vertex pairs not connected by a multiedge, if we adopt the convention that 0! = 1. For self-edges there is a further factor of two because the interchange of the two ends of the edge does not generate a new matching. Combining these results, the number of matchings corresponding to a network turns out to be

$$N = \frac{\prod_i k_i!}{\prod_{i<j} A_{ij}! \prod_i A_{ii}!!},$$

(13.29)

where $n!! = n(n - 2)(n - 4) \ldots 2$ with n even is the so-called double factorial of n. Then the total probability of a particular network within the configuration model ensemble is N/Ω as before. Since the denominator in Eq. (13.29) depends not only on the degree sequence but also on the structure of the network itself, different networks do appear with different probabilities.

As we mentioned, however, the average densities of self-edges and multiedges in the configuration model vanish as n becomes large, so that the variation in probabilities is relatively small in the large-n limit, but it nonetheless does occasionally assume some importance and is therefore worth bearing in mind (see, for instance, Ref. [220]).

As discussed above, we are sometimes (indeed often) interested in the case where it is the degree distribution of the network that is specified rather than the degree sequence. That is, we specify the probability distribution p_k from which the degree sequence is drawn rather than the sequence itself. We can define an obvious extension of the configuration model to this case: we draw a degree sequence from the specified distribution and then generate a network with that degree sequence using the technique described above. More precisely, we define an ensemble in which each degree sequence $\{k_i\}$ appears with probability $\prod_i p_{k_i}$. Then if we can calculate an average value $X(\{k_i\})$ for some quantity of interest X in the standard configuration model, the average value in the extended model is given by

$$\langle X \rangle = \sum_{k_1=0}^{\infty} \cdots \sum_{k_n=0}^{\infty} X(\{k_i\}) \prod_{i=0}^{n} p_{k_i}.$$

(13.30)

In practice the difference between the two models is not actually very great. As we will see, the crucial parameter that enters into most of our configuration model calculations is the fraction of vertices that have each possible degree k. In the extended model above, this fraction is, by definition, equal to p_k in the limit of large n. If, on the other hand, the degree sequence is fixed then

we simply calculate the fraction from the degree sequence and then use those numbers. In either the case the formulas for calculated quantities are the same.

13.2.1 EDGE PROBABILITY IN THE CONFIGURATION MODEL

A central property of the configuration model is the probability p_{ij} of the occurrence of an edge between two specified vertices, i and j. Obviously if either vertex i or vertex j has degree zero then the probability of an edge is zero, so let us assume that k_i, $k_j > 0$. Now consider any one of the stubs that emerges from vertex i. What is the probability that this stub is connected by an edge to any of the stubs of vertex j? There are $2m$ stubs in total, or $2m - 1$ excluding the one connected to i that we are currently looking at. Of those $2m - 1$, exactly k_j of them are attached to vertex j. So, given that any stub in the network is equally likely to be connected to any other, the probability that our particular stub is connected to any of those around vertex j is $k_j/(2m-1)$. But there are k_i stubs around vertex i, so the total probability of a connection between i and j is

$$p_{ij} = \frac{k_i k_j}{2m-1}.$$

(13.31)

Technically, since we have added the probabilities of independent events, this is really the average number of edges between i and j, rather than the probability of having an edge at all. But in the limit of large m, this number becomes small (for given k_i, k_j), and the average number of edges and the probability of an edge become equal. Also in the limit of large m we can ignore the -1 in the denominator and hence we can write

$$p_{ij} = \frac{k_i k_j}{2m}.$$

(13.32)

Note that, even though we assumed $k_i, k_j > 0$, this expression also gives the right result if either degree is zero, namely that in that case the probability of connection is zero.

We can use this result, for example, to calculate the probability of having two edges between the same pair of vertices. The probability of having one edge between vertices i and j is p_{ij} as above. Once we have one edge between the vertices the number of available stubs at each is reduced by one, and hence the probability of having a second edge is given by Eq. (13.32) but with k_i and k_j each reduced by one: $(k_i-1)(k_j-1)/2m$. Thus the probability of having (at least) two edges, i.e., of having a multiedge between i and j, is $k_i k_j (k_i-1)\,(k_j-1)/(2m)^2$ and, summing this probability over all vertices and dividing by two (to avoid double counting of vertex pairs), we find that the expected total number of multiedges in the network is

$$\frac{1}{2(2m)^2}\sum_{ij}k_ik_j(k_i-1)(k_j-1) = \frac{1}{2\langle k\rangle^2 n^2}\sum_i k_i(k_i-1)\sum_j k_j(k_j-1)$$
$$= \frac{1}{2}\left[\frac{\langle k^2\rangle - \langle k\rangle}{\langle k\rangle}\right]^2, \qquad (13.33)$$

where

$$\langle k\rangle = \frac{1}{n}\sum_i k_i, \qquad \langle k^2\rangle = \frac{1}{n}\sum_i k_i^2, \qquad (13.34)$$

and we have used $2m = \sum_i k_i = n\langle k\rangle$ (see Eq. (6.23)). Thus the expected number of multiedges remains constant as the network grows larger, so long as $\langle k^2\rangle$ is constant and finite, and the density of multiedges—the number per vertex—vanishes as $1/n$. We used this result in a number of our earlier arguments.[188]

Another way to derive the expression in Eq. (13.32) is to observe that there are k_ik_j possible edges we could form between vertices i and j, while the total number of possible edges in the whole graph is the number of ways of choosing a pair of stubs from the $2m$ total stubs, or $\binom{2m}{2} = m(2m-1)$. The probability that any particular edge falls between i and j is thus given by the ratio $k_ik_j/m(2m-1)$, and if we make a total of m edges then the expected total number of edges between i and j is m times this quantity, which gives us Eq. (13.31) again.

The only case in which this derivation is not quite right is for self-edges. In that case the number of pairs of stubs is not k_ik_j but instead is $\binom{k_i}{2} = \frac{1}{2}k_i(k_i-1)$ and hence the probability of a self-edge from vertex i to itself is

$$p_{ii} = \frac{k_i(k_i-1)}{4m}. \qquad (13.35)$$

We can use this result to calculate the expected number of self-edges in the network, which is given by the sum over all vertices i:

$$\sum_i p_{ii} = \sum_i \frac{k_i(k_i-1)}{4m} = \frac{\langle k^2\rangle - \langle k\rangle}{2\langle k\rangle},$$

(13.36)

This expression remains constant as $n \to \infty$ provided k^2 remains constant, and hence, as with the multiedges, the density of self-edges in the network vanishes as $1/n$ in the limit of large network size.

We can use Eqs. (13.32) and (13.35) to calculate a number of other properties of vertices in the configuration model. For instance, we can calculate the expected number n_{ij} of common neighbors that vertices i and j share. The probability that i is connected to another vertex l is p_{il} and the probability that j is connected to the same vertex would likewise normally be p_{jl}. However, as with the calculation of multiedges above, if we already know that i is connected to l, then the number of available stubs at vertex l is reduced by one and, rather than being given by the normal expression (13.32), the probability of a connection between j and l is $k_j\ (k_l - 1)/2m$. Multiplying the probabilities for the two edges and summing over l, we then get our expression for the expected number of common neighbors of i and j:

$$n_{ij} = \sum_l \frac{k_i k_l}{2m} \frac{k_j(k_l - 1)}{2m} = \frac{k_i k_j}{2m} \frac{\sum_l k_l(k_l - 1)}{n\langle k \rangle}$$
$$= p_{ij} \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}.$$

(13.37)

Thus the probability of sharing a common neighbor is equal to the probability $p_{ij} = k_i k_j/2m$ of having a direct connection times a multiplicative factor that depends only on the mean and variance of the degree distribution but not on the properties of the vertices i and j themselves.

In this calculation we have ignored the fact that the probability of self-edges, Eq. (13.35), is different from the probability for other edges. As we have seen, however, the density of self-edges in the configuration model tends to zero as $n \to \infty$, so in that limit it is usually safe to make the approximation that Eq. (13.32) applies for all i and j.

13.2.2 RANDOM GRAPHS WITH GIVEN EXPECTED DEGREE

The configuration model of the previous section is, as we have said, similar in some ways to the standard random graph $G(n, m)$ described in Section 12.1, in which we distribute a fixed number m of edges at random between n vertices. In the configuration model the total number of edges is again fixed, having value $m = \frac{1}{2}\sum_i k_i$, but in addition we now also fix the individual degree of every vertex as well.

It is natural to ask whether there is also an equivalent of $G(n, p)$—the model in which only the probability of edges is fixed and not their number—and indeed there is. We simply place an edge between each pair of vertices i, j with independent probabilities taking the form of Eq. (13.32). We define a parameter c_i for each vertex and then place an edge between vertices i and j with probability $p_{ij} = c_i c_j/2m$. As with the configuration model, we must allow self-edges if the model is to be tractable, and again self-edges have to be treated a little differently from ordinary edges. It turns out that the most satisfactory definition of the edge probability is

$$p_{ij} = \begin{cases} c_i c_j/2m & \text{for } i \neq j, \\ c_i^2/4m & \text{for } i = j, \end{cases}$$

(13.38)

where m is now defined by

$$\sum_i c_i = 2m.$$

(13.39)

With this choice the average number of edges in the network is

$$\sum_{i \le j} p_{ij} = \sum_{i<j} \frac{c_i c_j}{2m} + \sum_i \frac{c_i^2}{4m} = \sum_{ij} \frac{c_i c_j}{4m} = m,$$

(13.40)

as before. We can also calculate the average number of ends of edges connected to a vertex i, i.e., its average degree k_i. Allowing for the fact that a self-edge contributes two ends of edges to the degree, we get

$$\langle k_i \rangle = 2p_{ii} + \sum_{j(\neq i)} p_{ij} = \frac{c_i^2}{2m} + \sum_{j(\neq i)} \frac{c_i c_j}{2m} = \sum_j \frac{c_i c_j}{2m} = c_i.$$

(13.41)

In other words the parameters c_i appearing in the definition of p_{ij}, Eq. (13.38), are the average or expected degrees in this model, just as the parameter c in $G(n, p)$ is the average degree of a vertex. The *actual* degree of a vertex could in principle take almost any value, depending on the luck of the draw about which edges happen to get randomly created and which do not. In fact one can show that the degree of vertex i will have a Poisson distribution with mean c_i, meaning that in practice it will be quite narrowly distributed about c_i, but there will certainly be some variation, unless c_i is zero.[191] Note that c_i does not have to be an integer, unlike the degrees k_i appearing in the configuration model.

Thus in this model we specify the expected number of edges m and the expected degree sequence $\{c_i\}$ of the network but not the actual number of edges and actual degree sequence. This is again analogous to $G(n, p)$, in which we specify only the expected number of edges and not the actual number. Unfortunately, this means we usually cannot choose the degree *distribution* of our network, because the distribution of the actual degrees k_i is not the same as the distribution of the expected degrees c_i. This is a substantial disadvantage of the model since the degree distribution is widely considered to be a crucial property of networks.[192]

This is unfortunate, because this model is in other respects a very nice one. It is straightforward to treat analytically and many of the derivations are substantially simpler for this model than for the configuration model. Nonetheless, because we place such a premium on being able to choose the degree distribution, this model is in fact hardly ever used in real calculations of the properties of networks. Instead, most calculations are made using the configuration model and this is the direction that we will take in this book as well. In the following sections, we describe how one can make use of the machinery of generating functions to calculate many of the properties of the configuration model exactly in the limit of large network size.

13.3 EXCESS DEGREE DISTRIBUTION

In the remainder of this chapter we describe the calculation of a variety of properties of the configuration model. We begin our discussion with some fundamental observations about the model—and networks in general—that will prove central to later developments.

Consider a configuration model with degree distribution p_k, meaning that a fraction p_k of the vertices have degree k. (We can consider either the standard version of the model in which the degree sequence is fixed, as in Section 13.2, or the version of Eq. (13.30) in which only the distribution is fixed but not the exact degree sequence.) The distribution p_k tells us the probability that a vertex chosen uniformly at random from our network has degree k. But suppose instead that we take a vertex (randomly chosen or not) and follow one of its edges (assuming it has at least one) to the vertex at the other end. What is the probability that this vertex will have degree k?

The answer cannot just be p_k. For instance, there is no way to reach a vertex with degree zero by following an edge in this way, because a vertex with degree zero has no edges. So the probability of finding a vertex of degree zero is itself zero, and not p_0.

In fact, the correct probability for general k is not hard to calculate. We know that an edge emerging from a vertex in a configuration model network has equal chance of terminating at any "stub" of an edge anywhere else in the network (see Section 13.2). Since there are $\sum_i k_i = 2m$ stubs in total, or $2m-1$ excluding the one at the beginning of our edge, and k of them are attached to any particular vertex with degree k, our edge has probability $k/(2m-1)$ of ending at any particular vertex of degree k. In the limit of large network size, where m becomes large (assuming the degree distribution, and hence the average degree, remain constant), we can ignore the -1 and just write this as $k/2m$.

Given that p_k is the total fraction of vertices in the network with degree k, the total number of such vertices is np_k, and hence the probability of our edge attaching to *any* vertex with degree k is

$$\frac{k}{2m} \times np_k = \frac{kp_k}{\langle k \rangle},$$

(13.42)

where $\langle k \rangle$ is the average degree over the whole network and we have made use of the fact that $2m = n\langle k \rangle$, Eq. (6.23).

Thus the probability that we reach a vertex of degree k upon following an edge in this way is proportional not to p_k but to kp_k. To put that another way, the vertex you reach by following an edge is not a typical vertex in the network. It is more likely to have high degree than a typical vertex. Physically, the reasoning behind this observation is that a vertex with degree k has k edges attached to it, and you can reach that vertex by following any one of them. Thus if we choose an edge and follow it you have k times the chance of reaching a vertex with degree k that you have of reaching a vertex with degree 1.

It is important to recognize that this is a property specifically of the configuration model (or similar random graph models). In the real world, the degrees of adjacent vertices in networks are

often correlated (see Section 7.13) and hence the probability of reaching a vertex of degree k when we follow an edge depends on what vertex we are coming from.[193] Nonetheless, it is found to apply approximately to many real-world networks, which is one of the reasons why insights gained from the configuration model are useful for understanding the world around us.

Equation (13.42) has some strange and counter-intuitive consequences. As an example, consider a randomly chosen vertex in the configuration model and let us calculate the average degree of a neighbor of that vertex. If we were using the configuration model to model a friendship network, for instance, the average degree of an individual's network neighbor would correspond to the average number of friends their friend has. This number is the average of the distribution in Eq. (13.42), which we get by multiplying by k and then summing over k thus:

$$\text{average degree of a neighbor} = \sum_k k \frac{k p_k}{\langle k \rangle} = \frac{\langle k^2 \rangle}{\langle k \rangle}. \tag{13.43}$$

Note that the average degree of a neighbor is thus different from the average degree k of a typical vertex in the network. In fact, it is in general larger, as we can show by calculating the difference

$$\frac{\langle k^2 \rangle}{\langle k \rangle} - \langle k \rangle = \frac{1}{\langle k \rangle}\left(\langle k^2 \rangle - \langle k \rangle^2\right) = \frac{\sigma_k^2}{\langle k \rangle}, \tag{13.44}$$

where $\sigma_k^2 = \langle k^2 \rangle - \langle k \rangle^2$ is the variance of the degree distribution. The variance, which is the square of the standard deviation, is necessarily non-negative and indeed is strictly positive unless every single vertex in the network has the same degree. Let us assume that there is some variation in the degrees so that σ_k^2 is greater than zero. The average degree k is also greater than zero, unless all vertices have degree zero. Thus Eq. (13.44) implies that $k^2 / k - k > 0$, or

$$\frac{\langle k^2 \rangle}{\langle k \rangle} > \langle k \rangle. \tag{13.45}$$

In other words, the average degree of the neighbor of a vertex is greater than the average degree of a vertex. In colloquial terms, "Your friends have more friends than you do."

At first sight, this appears to be a very strange result. Certainly it seems likely that there will be some vertices in the network with higher degree than the average. But there will also be some who have lower degree and when you average over all neighbors of all vertices surely the two should

cancel out. Surely the average degree of a neighbor should be the same as the average degree in the network as a whole. Yet Eq. (13.45) tells us that this is not so. And the equation really is correct. You can create a configuration model network on a computer and average the degrees of the neighbors of every vertex, and you'll find that the formula works to very high accuracy. Even more remarkably, as first shown by Feld [113], you can do the same thing with real networks and, although the configuration model formula doesn't apply exactly to these networks, the basic principle still seems to hold. Here, for instance, are some measurements for two academic collaboration networks, in which scientists are connected together by edges if they have coauthored scientific papers, and for a recent snapshot of the structure of the Internet at the autonomous system level:

Network	n	Average degree	Average neighbor degree	$\frac{\langle k^2 \rangle}{\langle k \rangle}$
Biologists	1 520 252	15.5	68.4	130.2
Mathematicians	253 339	3.9	9.5	13.2
Internet	22 963	4.2	224.3	261.5

According to these results a biologist's collaborators have, on average, more than four times as many collaborators as they do themselves. On the Internet, a node's neighbors have more than 50 times the average degree! Note that in each of the cases in the table the configuration model value of k^2 / k overestimates the real average neighbor degree, in some cases by a substantial margin.[195] This is typical of calculations using simplified network models: they can give you a feel for the types of effect one might expect to see, or the general directions of changes in quantities. But they usually don't give quantitatively accurate predictions for the behavior of real networks.

The fundamental reason for the result, Eq. (13.45), is that when you go through the vertices of a network and average the degrees of the neighbors of each one, many of those neighbors appear in more than one average. In fact, a vertex with degree k will appear as one of the neighbors of exactly k other vertices, and hence appear in k of the averages. This means that high-degree vertices are over-represented in the calculations compared with low-degree ones and it is this bias that pushes up the overall average value.

In most of the calculations that follow, we will be interested not in the total degree of the vertex at the end of an edge but in the number of edges attached to that vertex *other* than the one we arrived along. For instance, if we want to calculate the size of the component to which a vertex i belongs then we will want to know first of all how many neighbors i has, and then how many neighbors those neighbors have, *other than* i, and so on.

The number of edges attached to a vertex other than the edge we arrived along is called the *excess degree* of the vertex and it is just one less than the total degree. Since the vertex at the end of an edge always has degree at least 1 (because of that edge) the minimum value of the excess degree is zero.

We can calculate the probability distribution of the excess degree from Eq. (13.43). The probability q_k of having excess degree k is simply the probability of having total degree $k + 1$ and, putting $k \to k + 1$ in Eq. (13.43), we get

$$q_k = \frac{(k+1)p_{k+1}}{\langle k \rangle}. \tag{13.46}$$

(Note that the denominator is still just k, and not $k+1$, as you can verify for yourself by checking that Eq. (13.46) is correctly normalized so that $\sum_{k=0}^{\infty} q_k = 1$.)

The distribution q_k is called the *excess degree distribution* and it will come up repeatedly in the sections that follow. It is the probability distribution, for a vertex reached by following an edge, of the number of other edges attached to that vertex.

13.4 CLUSTERING COEFFICIENT

As a simple application of the excess degree distribution, let us calculate the clustering coefficient for the configuration model. Recall that the clustering coefficient is the average probability that two neighbors of a vertex are neighbors of each other.

Consider then a vertex v that has at least two neighbors, which we will denote i and j. Being neighbors of v, i and j are both at the ends of edges from v, and hence the number of other edges connected to them, k_i and k_j are distributed according to the excess degree distribution, Eq. (13.46). The probability of an edge between i and j is then $k_i k_j/2m$ (see Eq. (13.32)) and, averaging both k_i and k_j over the distribution q_k, we get an expression for the clustering coefficient thus:

$$\begin{aligned} C &= \sum_{k_i,k_j=0}^{\infty} q_{k_i} q_{k_j} \frac{k_i k_j}{2m} = \frac{1}{2m}\left[\sum_{k=0}^{\infty} k q_k\right]^2 \\ &= \frac{1}{2m\langle k\rangle^2}\left[\sum_{k=0}^{\infty} k(k+1)p_{k+1}\right]^2 \\ &= \frac{1}{2m\langle k\rangle^2}\left[\sum_{k=0}^{\infty} (k-1)k p_k\right]^2 \\ &= \frac{1}{n}\frac{\left[\langle k^2\rangle - \langle k\rangle\right]^2}{\langle k\rangle^3}, \end{aligned}$$

(13.47)

where we have made use of $2m = n\ \ k\ \ $, Eq. (6.23).

Like the clustering coefficient of the Poisson random graph, Eq. (12.11), this expression goes as n^{-1} for fixed degree distribution, and so vanishes in the limit of large system size. Hence, like the Poisson random graph, the configuration model appears to be an unpromising model for real-world networks with high clustering. Note, however, that Eq. (13.47) contains the second moment $\ \ k^2$ of the degree distribution in its numerator which can become large, for instance in networks with power-law degree distributions (see Section 8.4.2). This can result in surprisingly large values of C in the configuration model. For further discussion of this point see Section 8.6.

13.5 GENERATING FUNCTIONS FOR DEGREE DISTRIBUTIONS

In the calculations that follow, we will make heavy use of the generating functions for the degree distribution and the excess degree distribution of a network. We will denote these generating functions by $g_0(z)$ and $g_1(z)$ respectively. They are defined by

$$g_0(z) = \sum_{k=0}^{\infty} p_k z^k,$$

(13.48)

$$g_1(z) = \sum_{k=0}^{\infty} q_k z^k.$$

(13.49)

Although it will be convenient to have separate notations for these two commonly occurring functions, they are not really independent, since the excess degree distribution is itself defined in terms of the ordinary degree distribution via Eq. (13.46). Using Eq. (13.46) we can write $g_1(z)$ as

$$\begin{aligned} g_1(z) &= \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} (k+1) p_{k+1} z^k = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} k p_k z^{k-1} \\ &= \frac{1}{\langle k \rangle} \frac{\mathrm{d}g_0}{\mathrm{d}z}. \end{aligned}$$

(13.50)

But Eq. (13.22) tells us that the average vertex degree is $\langle k \rangle = g_0'(1)$, so

$$g_1(z) = \frac{g_0'(z)}{g_0'(1)}.$$

(13.51)

Thus if we can find $g_0(z)$, we can also find $g_1(z)$ directly from it, without the need to calculate the excess degree distribution explicitly.

For example, suppose our degree distribution is a Poisson distribution with mean c:

$$p_k = \mathrm{e}^{-c}\frac{c^k}{k!}.$$

(13.52)

Then its generating function is given by Eq. (13.6) to be

$$g_0(z) = \mathrm{e}^{c(z-1)}.$$

(13.53)

Applying Eq. (13.51), we then find that

$$g_1(z) = \mathrm{e}^{c(z-1)}.$$

(13.54)

In other words, $g_0(z)$ and $g_1(z)$ are identical in this case. (This is one reason why calculations are relatively straightforward for the Poisson random graph—there is no difference between the degree distribution and the excess degree distribution in that case, a fact you can easily demonstrate for yourself by substituting Eq. (13.52) directly into Eq. (13.46).)

A more complicated example is the power-law distribution, Eq. (13.10), which has a generating function given by Eq. (13.16) to be

$$g_0(z) = \frac{\mathrm{Li}_\alpha(z)}{\zeta(\alpha)},$$

(13.55)

where $\mathrm{Li}_\alpha(z)$ is the polylogarithm function and α is the exponent of the power law. Substituting this result into Eq. (13.51) and making use of Eq. (13.17) gives

$$g_1(z) = \frac{\mathrm{Li}_{\alpha-1}(z)}{z\,\mathrm{Li}_{\alpha-1}(1)} = \frac{\mathrm{Li}_{\alpha-1}(z)}{z\zeta(\alpha-1)},$$

(13.56)

where we have made use of the fact that $\mathrm{Li}_\alpha(1) = \zeta(\alpha)$ (see Eqs. (13.12) and (13.15)).

13.6 NUMBER OF SECOND NEIGHBORS OF A VERTEX

Armed with these results, we are now in a position to make some more detailed calculations of the properties of the configuration model. The first question we will address is a relatively simple one: what is the probability $p_k^{(2)}$ that a vertex has exactly k second neighbors in the network?

Let us break this probability down by writing it in the form

$$p_k^{(2)} = \sum_{m=0}^{\infty} p_m P^{(2)}(k|m),$$

(13.57)

where $P^{(2)}(k|m)$ is the probability of having k second neighbors given that we have m first neighbors and p_m is the ordinary degree distribution. Equation (13.57) says that the total probability of having k second neighbors is the probability of having k second neighbors given that we have m first neighbors, averaged over all possible values of m. We assume that we are given the degree distribution p_m; we need to find $P^{(2)}(k|m)$ and then complete the sum.

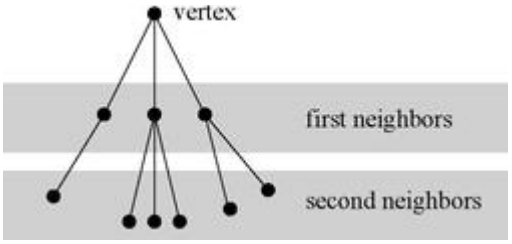


Figure 13.4: Calculation of the number of second neighbors of a vertex. The number of second neighbors of a vertex (top) is equal to the sum of the excess degrees of the first neighbors.

As illustrated in Fig. 13.4, the number of second neighbors of a vertex is equal to the sum of the excess degrees of the first neighbors. And as discussed in the previous section, the excess degrees are distributed according to the distribution q_k, Eq. (13.46), so that the probability that the excess degrees of our m first neighbors take the values $j_1 \ldots j_m$ is $\prod_{r=1}^{m} q_{j_r}$. Summing over all sets of values $j_1 \ldots j_m$, the probability that the excess degrees sum to k and hence that we have k second neighbors is

$$P^{(2)}(k|m) = \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \delta\Big(k, \sum_{r=1}^{m} j_r\Big) \prod_{r=1}^{m} q_{j_r}.$$

(13.58)

Substituting this expression into (13.57), we find that

$$p_k^{(2)} = \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \delta\left(k, \sum_{r=1}^{m} j_r\right) \prod_{r=1}^{m} q_{j_r}.$$

(13.59)

By now, you may be starting to find sums of this type familiar. We saw them previously in Eqs. (12.25) and (13.27), for example. We can handle this o ne by the same trick we used before: instead of trying to calculate $p_k^{(2)}$ directly, we calculate instead its generating function $g^{(2)}(z)$ thus:

$$\begin{aligned} g^{(2)}(z) &= \sum_{k=0}^{\infty} p_k^{(2)} z^k \\ &= \sum_{k=0}^{\infty} z^k \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \delta\left(k, \sum_{r=1}^{m} j_r\right) \prod_{r=1}^{m} q_{j_r} \\ &= \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} z^{\sum_{r=1}^{m} j_r} \prod_{r=1}^{m} q_{j_r} \\ &= \sum_{m=0}^{\infty} p_m \sum_{j_1=0}^{\infty} \cdots \sum_{j_m=0}^{\infty} \prod_{r=1}^{m} q_{j_r} z^{j_r} \\ &= \sum_{m=0}^{\infty} p_m \left[\sum_{j=0}^{\infty} q_j z^j \right]^m . \end{aligned}$$

(13.60)

But now we notice an interesting thing: the sum in square brackets in the last line is none other than the generating function $g_1(z)$ for the excess degree distribution, Eq. (13.49). Thus Eq. (13.60) can be written as

$$g^{(2)}(z) = \sum_{m=0}^{\infty} p_m [g_1(z)]^m = g_0(g_1(z)),$$

(13.61)

where $g_0(z)$ is the generating function for the ordinary degree distribution, defined in Eq. (13.48). So once we know the generating functions for our two basic degree distributions the generating function for the distribution of the second neighbors is very simple to calculate.

In fact, there was no need to go through this lengthy calculation to reach Eq. (13.61). We can derive the same result much more quickly by making use of the "powers" property of generating functions that we derived in Section 13.1.4. There we showed (Eq. (13.28)) that, given a quantity k distributed according to a distribution with generating function $g(z)$, m independent quantities drawn from the same distribution have a sum whose distribution is given by the generating function $[g(z)]^m$. We can apply this result here, by noting that the m excess degrees of the first neighbors of our vertex are just such a set of independent quantities. Given that $g_1(z)$ is the generating function for the distribution of a single one of them (Eq. (13.49)), the distribution $P^{(2)}(k|m)$ of their sum—which is the number of second neighbors—has generating function $[g_1(z)]^m$. That is,

$$\sum_{k=0}^{\infty} P^{(2)}(k|m)z^k = [g_1(z)]^m.$$

(13.62)

Now, using Eq. (13.57), the generating function for $p_k^{(2)}$ is

$$\begin{aligned} g^{(2)}(z) &= \sum_{k=0}^{\infty} p_k^{(2)} z^k = \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} p_m P^{(2)}(k|m) z^k \\ &= \sum_{m=0}^{\infty} p_m \sum_{k=0}^{\infty} P^{(2)}(k|m) z^k = \sum_{m=0}^{\infty} p_m [g_1(z)]^m \\ &= g_0(g_1(z)). \end{aligned}$$

(13.63)

In future calculations, we will repeatedly make use of this shortcut to get our results, rather than taking the long route exemplified in Eq. (13.60).

We can also use similar methods to calculate the probability distribution of the number of third neighbors. The number of third neighbors is the sum of the excess degrees of each of the second neighbors. Thus, if there are m second neighbors, then the probability distribution $P^{(3)}(k|m)$ of the number of third neighbors has generating function $[g_1(z)]^m$ and the overall probability of having k third neighbors is exactly analogous to Eq. (13.63):

$$\begin{aligned} g^{(3)}(z) &= \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} p_m^{(2)} P^{(3)}(k|m) z^k = \sum_{m=0}^{\infty} p_m^{(2)} \sum_{k=0}^{\infty} P^{(3)}(k|m) z^k \\ &= \sum_{m=0}^{\infty} p_m^{(2)} [g_1(z)]^m = g^{(2)}(g_1(z)) \\ &= g_0(g_1(g_1(z))). \end{aligned}$$

(13.64)

Indeed, the generating function for the number of neighbors at any distance d can be expressed this way as

$$\begin{aligned} g^{(d)}(z) &= \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} p_m^{(d-1)} P^{(d)}(k|m) z^k \\ &= \sum_{m=0}^{\infty} p_m^{(d-1)} \sum_{k=0}^{\infty} P^{(d)}(k|m) z^k = \sum_{m=0}^{\infty} p_m^{(d-1)} [g_1(z)]^m \\ &= g^{(d-1)}(g_1(z)). \end{aligned}$$

(13.65)

In other words $g^{(d)}(z) = g_0\,(g_1\,(\ldots\,g_1\,(z)\ldots))$, with d - 1 copies of g_1 nested inside a single g_0. This expression is correct at arbitrary distances on an infinite network. On a finite network it will break down if d becomes large enough but will be accurate for small values of d.

These results are all very good, but what use are they? Even given the generating function $g^{(2)}(z)$ it is typically quite difficult to extract explicit probabilities for numbers of second neighbors in the network. For instance, if our degree distribution were Poisson with mean c then $g_0(z) = g_1\,(z) = e^{c(z-1)}$ as in Eqs. (13.53) and (13.54) and

$$g^{(2)}(z) = e^{c(e^{c(z-1)}-1)}.$$

(13.66)

But to find the actual probabilities we have to apply Eq. (13.2), which involves calculating derivatives of $g^{(2)}\,(z)$. One can, with a little work, calculate the first few derivatives, but finding a general formula for the nth derivative is hard.[196]

What we can do, however, is calculate the average number of neighbors at distance d. The average of a distribution is given by the first derivative of its generating function evaluated at $z = 1$ (see Eq. (13.22)) and the derivative of Eq. (13.63) is

$$\frac{dg^{(2)}}{dz} = g_0'(g_1(z))\,g_1'(z).$$

(13.67)

Setting $z = 1$ and recalling that $g_1(1) = 1$ (Eq. (13.20)), we find that the average number c_2 of second neighbors is

$$c_2 = g_0'(1)g_1'(1).$$

(13.68)

But $g_0'(1) = \langle k \rangle$ and

$$\begin{aligned} g_1'(1) &= \sum_{k=0}^{\infty} k q_k \\ &= \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} k(k+1) p_{k+1} = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} (k-1) k p_k \\ &= \frac{1}{\langle k \rangle} (\langle k^2 \rangle - \langle k \rangle). \end{aligned}$$

(13.69)

where we have used Eq. (13.46). Thus the mean number of second neighbors can also be written

$$c_2 = \langle k^2 \rangle - \langle k \rangle.$$

(13.70)

We can take this approach further and calculate the mean number c_d of neighbors at any distance d. Differentiating Eq. (13.65) we get

$$\frac{\mathrm{d}g^{(d)}}{\mathrm{d}z} = g^{(d-1)'}(g_1(z))\, g_1'(z),$$

(13.71)

and setting $z = 1$ we get

$$c_d = g^{(d-1)'}(1)\, g_1'(1) = c_{d-1}\, g_1'(1).$$

(13.72)

Making use of Eq. (13.68) to write $g_1'(1) = c_2/c_1$ where $c_1 = k$, this can be expressed in the simple form

$$c_d = c_{d-1}\frac{c_2}{c_1},$$

(13.73)

which implies that

$$c_d = \left(\frac{c_2}{c_1}\right)^{d-1} c_1.$$

(13.74)

In other words, once we know the mean numbers of first and second neighbors, c_1 and c_2, we know everything. What's more, the average number of neighbors at distance d either grows or falls off exponentially, depending on whether c_2 is greater or less than c_1. This observation is strongly reminiscent of the argument we made in Section 12.5 for the appearance of a giant component in a random graph. There we argued that if the number of vertices you can reach within a certain distance is increasing with that distance (on average) then you must have a giant component in the network, while if it is decreasing there can be no giant component. Applying the same reasoning here, we conclude that the configuration model has a giant component if and only if we have

$$c_2 > c_1.$$

(13.75)

Using Eq. (13.70) for c_2 and putting $c_1 = k$, we can also write this condition as $k^2 - k > k$ or

$$\langle k^2 \rangle - 2\langle k \rangle > 0.$$

(13.76)

This condition for the existence of a giant component in the configuration model was first given by Molloy and Reed [224] in 1995.[197]

13.7 GENERATING FUNCTIONS FOR THE SMALL COMPONENTS

In this section and the following one we examine the sizes of components in the configuration model. As we will see, the situation is qualitatively similar to that for the Poisson random graph in that a configuration model network generally has at most one giant component, plus a large number of small components. We will approach the calculation of component sizes by a route different from the one we took for the Poisson random graph and examine first the properties of the small components. We will see that it is possible to calculate the distribution of the sizes of the small components by a method similar to the one we used in the Poisson case. Then we can use these results to get at the properties of the giant component: once we have the sizes of the small components, we can subtract them from the size of the graph as a whole and whatever is left, if anything, must be the giant component.

Let π_s be the probability that a randomly chosen vertex belongs to a small (non-giant) component of size s. We will calculate π_s by first calculating its generating function

$$h_0(z) = \sum_{s=1}^{\infty} \pi_s z^s.$$

(13.77)

Note that the minimum value of s is 1, since every vertex belongs to a component of size at least one (namely itself).

By an argument exactly analogous to that of Section 12.6.1 we can show that the small components in the configuration model are trees (in the limit of large n, provided the degree distribution is held constant as we go to the limit). We can use this fact to derive an expression for the distribution of small component sizes as follows.

Consider Fig. 13.5 (which is actually the same as the figure for the Poisson random graph in the previous chapter (Fig. 12.3), but it works just as well as an illustration of the configuration model). If vertex i is a member of a small component then that component is necessarily a tree. Just as in the Poisson case, this implies that the sets of vertices reachable along each of its edges (shaded areas in Fig. 13.5a) are not connected, other than via vertex i, since if they were connected there would be a loop in the component and hence it would not be a tree.

Now, taking a hint from our argument in the Poisson case, let us remove vertex i from the network along with all its edges—see Fig. 13.5b. The shaded areas in the figure are now not connected to one another at all and hence are each now separate components in their own right. And the size of the component to which vertex i belongs on the original network is equal to the sum of the sizes of these new components, plus one for vertex i itself.

A crucial point to notice, however, is that the neighbors n_1, n_2, ... of vertex i are, by definition, reached by following an edge. Hence, as we have discussed, these are not typical network vertices, being more likely to have high degree than the typical vertex. Thus the components that they belong to in Fig. 13.5b—the shaded regions in the figure—are not distributed according to π_s. Instead they must have some other distribution. Let us denote this distribution by ρ_s. More specifically, let ρ_s be the probability that the vertex at the end of an edge belongs to a small

component of size s after that edge is removed. Let us also define the generating function for this distribution to be

$$h_1(z) = \sum_{s=0}^{\infty} \rho_s z^s.$$

(13.78)

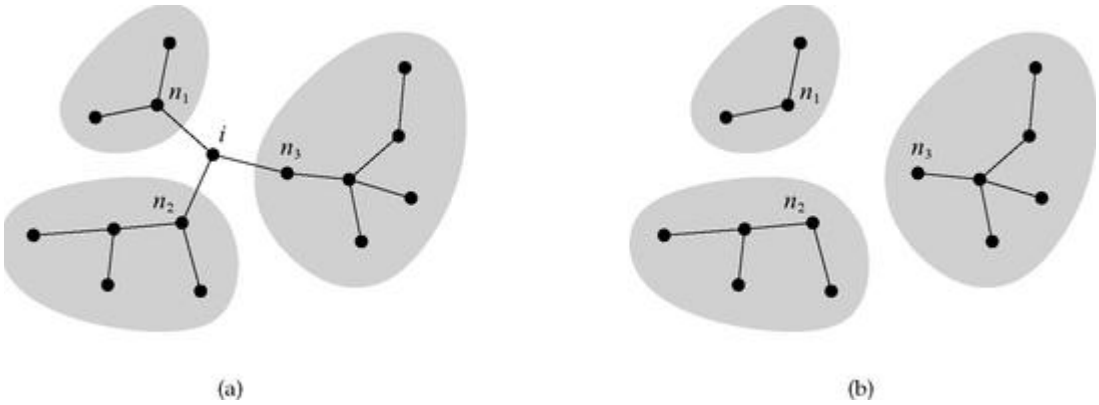


Figure 13.5: The size of one of the small components in the configuration model. (a) The size of the component to which a vertex i belongs is the sum of the number of vertices in each of the subcomponents (shaded regions) reachable via is neighbors n_1, n_2, n_3, plus one for i itself. (b) If vertex i is removed the subcomponents become components in their own right.

We don't yet know the value of ρ_s or its generating function and we will have to calculate them later, but for the moment let us proceed with the information we have.

Suppose that vertex i on the original network has degree k and let us denote by $P(s|k)$ the probability that, after i is removed, its k neighbors belong to small components of sizes summing to exactly s. Alternatively, $P(s-1|k)$ is the probability that i itself belongs to a small component of size s given that its degree is k. Then the total probability π_s that i belongs to a small component of size s is this probability averaged over k thus:

$$\pi_s = \sum_{k=0}^{\infty} p_k P(s-1|k).$$

(13.79)

Substituting this expression into Eq. (13.77) we then get an expression for the generating function for π_s as follows:

$$h_0(z) = \sum_{s=1}^{\infty} \sum_{k=0}^{\infty} p_k P(s-1|k) z^s = z \sum_{k=0}^{\infty} p_k \sum_{s=1}^{\infty} P(s-1|k) z^{s-1}$$
$$= z \sum_{k=0}^{\infty} p_k \sum_{s=0}^{\infty} P(s|k) z^s.$$

(13.80)

The final sum in this expression is the generating function for the probability that the k neighbors belong to small components whose size sums to s. But the sizes of the small components are independent of one another and hence we can use the "powers" property of generating functions (Section 13.1.4), which tells us that the generating function we want is just equal to the generating function for the size of the component any single neighbor belongs to—the function that we denoted $h_1(z)$ above—raised to the kth power. Thus

$$h_0(z) = z \sum_{k=0}^{\infty} p_k [h_1(z)]^k = z g_0(h_1(z)).$$

(13.81)

We still don't know the generating function $h_1(z)$ but we can derive it now quite easily. We consider the network in which vertex i is removed and ask what is the probability ρ_s that one of the neighbors of i belongs to a component of size s in this network. In the limit of large network size, the removal of the single vertex i will have no effect on the degree distribution, so the network still has the same distribution as before, which means that if the neighbor has degree k then its probability of belonging to a component of size s is $P(s-1|k)$, just as before. Note, however, that the degree k does not follow the ordinary degree distribution. Since the neighbor was reached by following an edge from i, its degree, discounting the edge to i that has been removed, follows the excess degree distribution q_k defined in Eq. (13.46), rather than the ordinary degree distribution. Thus

$$\rho_s = \sum_{k=0}^{\infty} q_k P(s-1|k),$$

(13.82)

and, substituting this expression into Eq. (13.78), we have

$$h_1(z) = \sum_{s=1}^{\infty} \sum_{k=0}^{\infty} q_k P(s-1|k) z^s = z \sum_{k=0}^{\infty} q_k \sum_{s=0}^{\infty} P(s|k) z^s.$$

(13.83)

As before, the last sum is the generating function for $P(s|k)$, which is equal to $[h_1(z)]^k$, and hence

$$h_1(z) = z\sum_{k=0}^{\infty} q_k [h_1(z)]^k = z g_1(h_1(z)).$$

(13.84)

Collecting together our results, the generating functions for π_s and ρ_s thus satisfy

$$h_0(z) = z g_0(h_1(z)),$$

(13.85)

$$h_1(z) = z g_1(h_1(z)).$$

(13.86)

If we can solve the second of these equations for $h_1(z)$ then we can substitute the result into the first equation and we have our answer for $h_0(z)$. In practice, it is often not easy to solve for $h_1(z)$, and, even if it is, extracting the actual component size distribution from the generating function can be difficult. But that does not mean that these results are useless. On the contrary, there are many useful things we can deduce from them. One important quantity we can calculate is the size of the giant component.

13.8 GIANT COMPONENT

Given the definition $h_0(z) = \sum_s \pi_s z^s$, where π_s is the probability that a randomly chosen vertex belongs to a small component of size s, we have $h_0(1) = \sum_s \pi_s$, which is the total probability that a randomly chosen vertex belongs to a small component. Unlike most generating functions, it is not necessarily the case that $h(1) = 1$ because there may be a giant component in the network. If there is a giant component then some of the vertices do not belong to any small component and $\sum_s \pi_s$ will be less than 1. In fact, $\sum_s \pi_s$ will be simply the fraction of vertices that belong to small components and hence the fraction S of vertices belonging to the giant component is

$$S = 1 - \sum_{s=0}^{\infty} \pi_s = 1 - h_0(1) = 1 - g_0(h_1(1)),$$

(13.87)

where we have used Eq. (13.85). The value of $h_1(1)$ we can get from Eq. (13.86):

$$h_1(1) = g_1(h_1(1)).$$

(13.88)

The quantity $h_1(1)$ will occur frequently in subsequent developments, so for convenience let us define the shorthand notation

$$u = h_1(1),$$

(13.89)

in which case Eqs. (13.87) and (13.88) can be written

$$S = 1 - g_0(u),$$

(13.90)

$$u = g_1(u).$$

(13.91)

In other words, u is a fixed point of the function $g_1(z)$—a point where the function is equal to its own argument—and if we can find this fixed point then we need only substitute the result into Eq. (13.90) and we have the size of the giant component.

Since $g_1(1) = 1$ (see Eq. (13.20) and the discussion that precedes it), there is always a fixed point of g_1 at $u = 1$, but this solution gives $S = 1 - g_0(1) = 0$ and hence no giant component. If there is to be a giant component there must be at least one other non-trivial solution to Eq. (13.91). We will see some examples of such solutions shortly.

The quantity $u = h_1(1)$ has a simple physical interpretation. Recall that $h_1(z) = \sum_s \rho_s z^s$ is the generating function for the probability ρ_s that the vertex reached by following an edge belongs to a small component of size s if that edge is removed. Thus $h_1(1) = \sum_s \rho_s$ is the total probability that such a vertex belongs to a small component of any size, or equivalently the probability that it doesn't belong to the giant component.

This observation suggests an alternative and simpler derivation of Eqs. (13.90) and (13.91) for the size of a giant component, as follows. To belong to the giant component, a vertex A must be connected to the giant component via at least one of its neighbors. Or equivalently, A does not belong to the giant component if (and only if) it is not connected to the giant component via any of its neighbors. Let us *define* u to be the average probability that a vertex is not connected to the giant component via its connection to some particular neighboring vertex. If vertex A has k neighbors, then the probability that it is not connected to the giant component via any of them is thus u^k. And the average of this probability over the whole network is $\sum_k p_k u^k = g_0(u)$, which is the average probability that a vertex is not in the giant component. But this probability is also, by definition, equal to $1 - S$, where S is the fraction of the graph occupied by the giant component and hence $1 - S = g_0(u)$ or

$$S = 1 - g_0(u),$$

(13.92)

which is Eq. (13.90) again.

Now let us ask what the value of u is. The probability that you are not connected to the giant component via a particular neighboring vertex is equal to the probability that *that* vertex is not connected to the giant component via any of its other neighbors. If there are k of those other neighbors, then that probability is again u^k. But because we are talking about a neighboring vertex, k is now distributed according to the excess degree distribution q_k, Eq. (13.46), and hence taking the average, we find that $u = \sum_k q_k u^k$ or

$$u = g_1(u),$$

(13.93)

which is Eq. (13.91) again. Thus we have rederived our two equations for the size of the giant component, but by a much shorter route. The main disadvantage of this method is that it only gives the size of the giant component and not the complete generating function for all the other components as well, and this is the reason why we took the time to go through the longer derivation. There are many further results we can derive by knowing the entire generating function, as we show in the next section.

13.8.1 EXAMPLE

Let's take a look at a concrete example and see how calculations for the configuration model work out in practice. Consider a network like that of the first example in Section 13.1.1 that has vertices only of degree 0, 1, 2 and 3, and no vertices of higher degree. Then the generating functions $g_0(z)$ and $g_1(z)$ take the form

$$g_0(z) = p_0 + p_1 z + p_2 z^2 + p_3 z^3,$$

(13.94)

$$\begin{aligned} g_1(z) &= \frac{g_0'(z)}{g_0'(1)} = \frac{p_1 + 2p_2 z + 3p_3 z^2}{p_1 + 2p_2 + 3p_3} \\ &= q_0 + q_1 z + q_2 z^2. \end{aligned}$$

(13.95)

Equation (13.91) is thus quadratic in this case, $u = q_0 + q_1 u + q_2 u^2$, which has the solutions

$$u = \frac{1 - q_1 \pm \sqrt{(1-q_1)^2 - 4q_0 q_2}}{2q_2}.$$

(13.96)

However, we know that $\sum_k q_k = 1$, and hence in this case $1 - q_1 = q_0 + q_2$. Using this result to eliminate q_1 we get

$$\begin{aligned} u &= \frac{(q_0 + q_2) \pm \sqrt{(q_0 + q_2)^2 - 4q_0 q_2}}{2q_2} \\ &= \frac{(q_0 + q_2) \pm (q_0 - q_2)}{2q_2} \\ &= 1 \quad \text{or} \quad \frac{q_0}{q_2}. \end{aligned}$$

(13.97)

Thus, as expected we have a solution $u = 1$, but we also have another non-trivial solution which *might* imply that we have a giant component.

If $q_2 < q_0$ then this non-trivial solution gives $u > 1$. Since u is a probability it cannot be greater than 1, so in this case we definitely do not have a giant component. On the other hand, if $q_2 > q_0$ we have a viable non-trivial solution $u < 1$ equal to

$$u = \frac{q_0}{q_2} = \frac{p_1}{3p_3},$$

(13.98)

where we have extracted values of q_0 and q_2 from Eq. (13.95). We can also write the condition $q_2 > q_0$ in terms of the p_k as

$$p_3 > \tfrac{1}{3}p_1.$$

(13.99)

In other words, there can be a giant component if the number of vertices of degree three exceeds one third the number of degree one. This is a remarkable result. It says that the number of vertices of degree zero and degree two don't matter at all (except to the extent that their absence makes room for more vertices of the other degrees). As we will see, this is actually a general result—the values of p_0 and p_2 never make any difference to the presence or absence of a giant component. On the other hand, the size of the giant component for the current example is given by Eq. (13.90) to be

$$S = 1 - g_0(u) = 1 - p_0 - \frac{p_1^2}{3p_3} - \frac{p_1^2 p_2}{9p_3^2} - \frac{p_1^3}{27p_3^2}.$$

(13.100)

Thus the size of the giant component does depend on p_0 and p_2, even though its presence or absence does not.

We have not, however, yet proved that a giant component actually does exist. In the regime where we have two solutions for u, one with $u = 1$ (no giant component) and one with $u < 1$ (there is a giant component) it is unclear which of these solutions we should believe. In Section 13.6, however, we showed that there is a giant component in the network when the degree sequence

satisfies a specific condition, Eq. (13.76). In the next section, we show that in fact this condition is always satisfied whenever a non-trivial solution $u < 1$ exists, and hence that there is always a giant component when we have such a solution.

13.8.2 GRAPHICAL SOLUTIONS AND THE EXISTENCE OF THE GIANT COMPONENT

The example given in the last section is unusual in that we can solve the fixed-point equation (13.91) exactly for the crucial parameter u. In most other cases exact solutions are not possible, but we can nonetheless get a good idea of the behavior of u by graphical means. The derivatives of $g_1(z)$ are proportional to the probabilities ρ_s and hence are all non-negative. That means that for $z \geq 0$, $g_1(z)$ is in general positive, an increasing function of its argument, and upward concave. It also takes the value 1 when $z = 1$. Thus it must look qualitatively like one of the curves in Fig. 13.6. The solution of the fixed-point equation $u = g_1(u)$ is then given by the intercept of the curve $y = g_1(u)$ with the line $y = u$ (the dotted line in the figure).

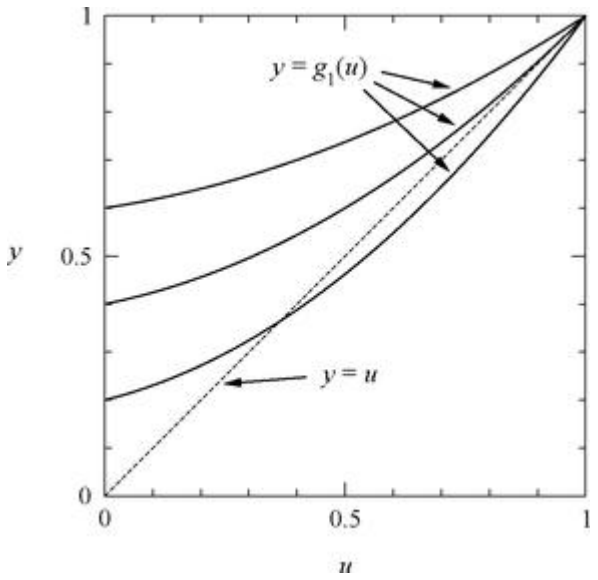


Figure 13.6: Graphical solution of Eq. (13.91). The solution of the equation $u = g_1(u)$ is given by the point at which the curve $y = g_1(u)$ intercepts the line $y = u$.

As we already know, there is always a trivial solution at $u = 1$ (top right in the figure). But now we can see that there can be just one other solution with $u < 1$ and only if the curve takes the right form. In particular, we have a non-trivial solution at $u < 1$ if the slope $g_1'(1)$ of the curve at $u = 1$ is greater than the slope of the dotted line. That is, if

$$g_1'(1) > 1. \tag{13.101}$$

Using Eq. (13.49) for $g_1(z)$, we have

$$g_1'(1) = \sum_{k=0}^{\infty} k q_k = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} k(k+1) p_k = \frac{1}{\langle k \rangle} \sum_{k=0}^{\infty} (k-1) k p_k$$
$$= \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}.$$

(13.102)

Thus our condition for the solution at $u < 1$ is

$$\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} > 1.$$

(13.103)

or equivalently,

$$\langle k^2 \rangle - 2\langle k \rangle > 0,$$

(13.104)

But this is none other than the condition for the existence of a giant component, Eq. (13.76). In other words, the conditions for the existence of a giant component and the existence of the non-trivial solution to Eq. (13.91) are exactly the same and and hence, as promised, there is always a giant component whenever a solution $u < 1$ exists for Eq. (13.91).

Writing $\langle k \rangle = n^{-1} \sum_i k_i$ and $\langle k^2 \rangle = n^{-1} \sum_i k_i^2$, we can also write Eq. (13.104) as

$$\sum_i k_i (k_i - 2) > 0.$$

(13.105)

But note now that, as before, vertices of degree zero and degree two make no contribution to the sum, since terms in which $k_i = 0$ or $k_i = 2$ vanish. Thus we can add as many vertices of degree zero or two to the network as we like (or take them away) and it will make no difference to the existence or not of a giant component. We noted a special case of this phenomenon in Section

13.8.1.

13.9 SIZE DISTRIBUTION FOR SMALL COMPONENTS

Having looked in some detail at the behavior of the giant component in the configuration model, let us return once more to the small components. In Eqs. (13.85) and (13.86) we have—in theory at least—the generating functions that give the entire distribution of sizes of the small components. Unfortunately, it is in most cases impossible to solve these equations exactly, but we can still extract plenty of useful information from them. For example, we can calculate the mean size of the component to which a randomly chosen vertex belongs, which is given by the equivalent of Eq. (12.29) thus:

$$\langle s \rangle = \frac{\sum_s s\pi_s}{\sum_s \pi_s} = \frac{h_0'(1)}{1-S} = \frac{h_0'(1)}{g_0(u)},$$

(13.106)

where we have used Eq. (13.90) in the final equality. Differentiating Eq. (13.85) we get

$$\begin{aligned} h_0'(z) &= g_0(h_1(z)) + z g_0'(h_1(z)) h_1'(z) \\ &= g_0(h_1(z)) + z g_0'(1) g_1(h_1(z)) h_1'(z) \\ &= \frac{h_0(z)}{z} + g_0'(1) h_1(z) h_1'(z), \end{aligned}$$

(13.107)

where we have used Eq. (13.51) in the second equality and Eqs. (13.85) and (13.86) in the third. Setting z = 1 we then get

$$h_0'(1) = h_0(1) + g_0'(1) h_1(1) h_1'(1) = 1 - S + g_0'(1) h_1'(1) u,$$

(13.108)

where we have used Eqs. (13.87) and (13.89). To calculate $h_1'(1)$ we differentiate Eq. (13.86) thus:

$$\begin{aligned} h_1'(z) &= g_1(h_1(z)) + z g_1'(h_1(z)) h_1'(z) \\ &= \frac{h_1(z)}{z} + z g_1'(h_1(z)) h_1'(z), \end{aligned}$$

(13.109)

or, rearranging,

$$h_1'(z) = \frac{h_1(z)/z}{1 - z g_1'(h_1(z))}.$$

(13.110)

Setting $z = 1$ in this expression gives

$$h_1'(1) = \frac{u}{1 - g_1'(u)}.$$

(13.111)

Combining Eqs. (13.106), (13.108), and (13.111), we then find that

$$\langle s \rangle = 1 + \frac{g_0'(1) u^2}{g_0(u)[1 - g_1'(u)]}.$$

(13.112)

Using values of S and u from Eqs. (13.90) and (13.91) we can then calculate s from this equation.

A simple case occurs when we are in the region where there is no giant component. In this region we have $S = 0$ and $u = 1$ by definition and hence

$$\langle s \rangle = 1 + \frac{g_0'(1)}{1 - g_1'(1)}.$$

(13.113)

Thus the average size of the component to which a vertex belongs diverges precisely at the point where $g_1'(1) = 1$, the point at which the curve in Fig. 13.6 is exactly tangent to the dotted line (the

middle curve in the figure). This is, of course, also the point at which the giant component first appears.

Thus the picture we have is similar to that shown in Fig. 12.4 for the Poisson random graph, in which the typical size of the component to which a vertex belongs grows larger and larger until we reach the point, or *phase transition*, where the giant component appears, at which it diverges. Beyond this point the small components shrink in size again, although the overall mean component size, including the giant component, is infinite.

Equation (13.113) can also be expressed in a couple of other forms that may be useful in some circumstances. From Eq. (13.69) we know that $g_1'(1) = (\langle k^2 \rangle - \langle k \rangle)/\langle k \rangle$ and, putting $g_0'(1) = \langle k \rangle$ also, we find that

$$\langle s \rangle = 1 + \frac{\langle k \rangle^2}{2\langle k \rangle - \langle k^2 \rangle}.$$

(13.114)

This expression can be evaluated easily given only a knowledge of the degree sequence and avoids the need to calculate any generating functions. Using the notation introduced earlier in which c_1 and c_2 are the mean number of first and second neighbors of a vertex, with c_2 given by Eq. (13.70), we can also write (13.114) in the form

$$\langle s \rangle = 1 + \frac{c_1^2}{c_1 - c_2},$$

(13.115)

so that the average size of the component a vertex belongs to is dictated entirely by the mean numbers of first and second neighbors.

13.9.1 AVERAGE SIZE OF A SMALL COMPONENT

As with the Poisson random graph, we must be careful about our claims in the previous section. We have calculated the average size s of the component to which a randomly chosen vertex belongs but this is not the same thing as the average size of a component, since more vertices belong to larger components, which biases the value of s. If we want the true average size R of the small components, we must use Eq. (12.36), which we reproduce here for convenience:

$$R = \frac{1-S}{\sum_s \pi_s/s}.$$

(13.116)

The sum can be calculated as before using the equivalent of Eq. (12.37):

$$\sum_{s=1}^{\infty} \frac{\pi_s}{s} = \int_0^1 \frac{h_0(z)}{z}\,\mathrm{d}z.$$

(13.117)

Taking $h_0(z)/z$ from Eq. (13.107), we get

$$\begin{aligned}\sum_{s=1}^{\infty} \frac{\pi_s}{s} &= \int_0^1 \frac{\mathrm{d}h_0}{\mathrm{d}z}\,\mathrm{d}z - g_0'(1)\int_0^1 h_1(z)\frac{\mathrm{d}h_1}{\mathrm{d}z}\,\mathrm{d}z \\ &= \int_0^{1-S} \mathrm{d}h_0 - \langle k\rangle \int_0^u h_1\,\mathrm{d}h_1 \\ &= 1 - S - \tfrac{1}{2}\langle k\rangle u^2.\end{aligned}$$

(13.118)

Then

$$R = \frac{2}{2 - \langle k\rangle u^2/(1-S)}.$$

(13.119)

Note that the value of this average at the transition point where $S = 0$ and $u = 1$ is just $2/(2 - k)$, which is normally perfectly finite.[198] Thus the average component size does not normally diverge at the transition (unlike s).

13.9.2 COMPLETE DISTRIBUTION OF SMALL COMPONENT SIZES

One of the most surprising results concerning the configuration model is that it is possible to derive an expression not just for the average size of the component to which a vertex belongs, but for the exact probability that it belongs to a component of any specific size—the probability that it belongs to a component of size ten, or a hundred, or a million. The derivation of this result is similar to the derivation given in Section 12.6.3 for the corresponding quantity for the Poisson random graph.

Since a component cannot have size zero, the generating function for the probabilities π_s has the form

$$h_0(z) = \sum_{s=1}^{\infty} \pi_s z^s,$$

(13.120)

with the sum starting at 1. Dividing by z and differentiating $s-1$ times, we then find that

$$\pi_s = \frac{1}{(s-1)!}\left[\frac{\mathrm{d}^{s-1}}{\mathrm{d}z^{s-1}}\left(\frac{h_0(z)}{z}\right)\right]_{z=0},$$

(13.121)

(which is just a minor variation on the standard formula, Eq. (13.2)). Using Eq. (13.85), this can also be written

$$\begin{aligned}\pi_s &= \frac{1}{(s-1)!}\left[\frac{\mathrm{d}^{s-1}}{\mathrm{d}z^{s-1}} g_0(h_1(z))\right]_{z=0} \\ &= \frac{1}{(s-1)!}\left[\frac{\mathrm{d}^{s-2}}{\mathrm{d}z^{s-2}}\left[g_0'(h_1(z))h_1'(z)\right]\right]_{z=0}.\end{aligned}$$

(13.122)

Now we make use of the Cauchy formula for the n derivative of a function, which says that

$$\left.\frac{\mathrm{d}^n f}{\mathrm{d}z^n}\right|_{z=z_0} = \frac{n!}{2\pi \mathrm{i}} \oint \frac{f(z)}{(z-z_0)^{n+1}}\,\mathrm{d}z,$$

(13.123)

where the integral is around a contour that encloses z_0 in the complex plane but encloses no poles in $f(z)$. Applying this formula to Eq. (13.122) with $z_0 = 0$ we get

$$\pi_s = \frac{1}{2\pi \mathrm{i}(s-1)} \oint \frac{g_0'(h_1(z))}{z^{s-1}} \frac{\mathrm{d}h_1}{\mathrm{d}z}\,\mathrm{d}z.$$

(13.124)

For our contour, we choose an infinitesimal circle around the origin.

Changing the integration variable to h_1, we can also write this as

$$\pi_s = \frac{1}{2\pi \mathrm{i}(s-1)} \oint \frac{g_0'(h_1)}{z^{s-1}}\,\mathrm{d}h_1.$$

(13.125)

Here we are regarding z now as a function of h_1, rather than the other way around. Furthermore, since $h_1(z)$ goes to zero as $z \to 0$, the contour in h_1 surrounds the origin too. (The proof is the same as for Eq. (12.46).)

Now we make use of Eq. (13.86) to eliminate z and write

$$\begin{aligned}\pi_s &= \frac{1}{2\pi \mathrm{i}(s-1)} \oint \frac{[g_1(h_1)]^{s-1} g_0'(h_1)}{h_1^{s-1}}\,\mathrm{d}h_1 \\ &= \frac{g_0'(1)}{2\pi \mathrm{i}(s-1)} \oint \frac{[g_1(h_1)]^{s}}{h_1^{s-1}}\,\mathrm{d}h_1,\end{aligned}$$

(13.126)

where we have made use of Eq. (13.51) in the second line. Given that the contour surrounds the origin, this integral is now in the form of Eq. (13.123) again, and hence

$$\pi_s = \frac{\langle k \rangle}{(s-1)!} \left[\frac{\mathrm{d}^{s-2}}{\mathrm{d}z^{s-2}} [g_1(z)]^s \right]_{z=0},$$

(13.127)

where we have written $g_0'(1) = \langle k \rangle$.

The only exception to this formula is for the case $s = 1$, for which Eq. (13.124) gives 0/0 and is therefore clearly incorrect. However, since the only way to belong to a component of size 1 is to have no connections to any other vertices, the probability π_1 is trivially equal to the probability of having degree zero:

$$\pi_1 = p_0.$$

(13.128)

Equations (13.127) and (13.128) give the probability that a randomly chosen vertex belongs to a component of size s in terms of the degree distribution. In principle if we know p_k we can calculate π_s. It is not always easy to perform the derivatives in practice and in some cases we may not even know the generating function $g_1(z)$ in closed form, but at least in some cases the calculations are possible. As an example, consider a network with the exponential degree distribution

$$p_k = (1 - \mathrm{e}^{-\lambda})\mathrm{e}^{-\lambda k},$$

(13.129)

with exponential parameter $\lambda > 0$. From Eqs. (13.9) and (13.51) the generating functions $g_0(z)$ and $g_1(z)$ are given by

$$g_0(z) = \frac{\mathrm{e}^{\lambda} - 1}{\mathrm{e}^{\lambda} - z}, \qquad g_1(z) = \left(\frac{\mathrm{e}^{\lambda} - 1}{\mathrm{e}^{\lambda} - z} \right)^2.$$

(13.130)

Then it is not hard to show that

$$\frac{\mathrm{d}^n}{\mathrm{d}z^n} [g_1(z)]^s = \frac{(2s - 1 + n)!}{(2s-1)!} \frac{[g_1(z)]^s}{(\mathrm{e}^{\lambda} - z)^n},$$

(13.131)

and hence

$$\pi_s = \frac{(3s-3)!}{(s-1)!(2s-1)!} e^{-\lambda(s-1)} \left(1 - e^{-\lambda}\right)^{2s-1}.$$

(13.132)

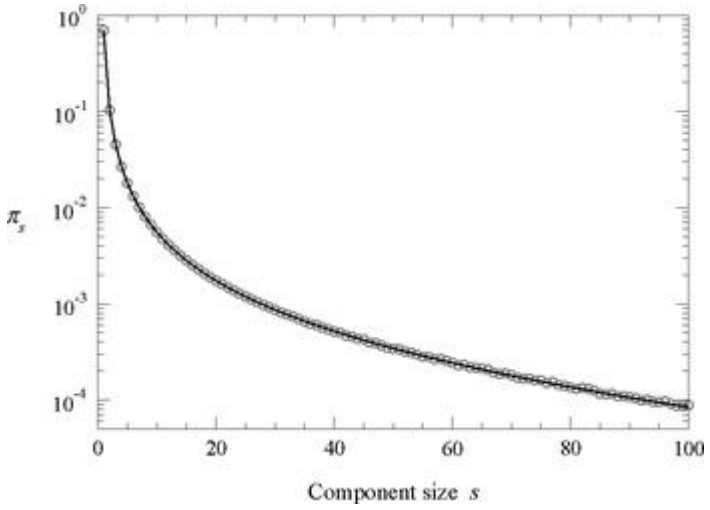


Figure 13.7: The distribution of component sizes in a configuration model. The probability π_s that a vertex belongs to a component of size s for the configuration model with an exponential degree distribution of the form (13.129) for $\lambda = 1.2$. The solid lines represent the exact formula, Eq. (13.132), for the $n \to \infty$ limit and the points are measurements of π_s averaged over 100 computer-generated networks with $n = 10^7$ vertices each.

Figure 13.7 shows a comparison of this formula with the results of numerical simulations for $\lambda = 1.2$ and, as we can see, the agreement between formula and simulations is good—our calculations seem to describe the simulated random graph well even though the graph is necessarily finite in size while the calculations are performed in the limit of large n.

13.10 POWER-LAW DEGREE DISTRIBUTIONS

As we saw in Section 8.4, a number of networks have degree distributions that approximately obey a power law. As an example of the application of the machinery developed in this chapter, let us look at the properties of a random graph with a power-law degree distribution.

Suppose we have a network with a "pure" power-law degree distribution of the form

$$p_k = \begin{cases} 0 & \text{for } k = 0, \\ k^{-\alpha}/\zeta(\alpha) & \text{for } k \geq 1. \end{cases}$$

(13.133)

(See Eq. (13.13).) Here $\alpha > 0$ is a constant exponent and $\zeta(\alpha)$ is the Riemann zeta function:

$$\zeta(\alpha) = \sum_{k=1}^{\infty} k^{-\alpha}.$$

(13.134)

Using the results of the previous sections we can, for instance, say whether there is a giant component in this network or not. Equation (13.76) tells us that there will be a giant component if and only if

$$\langle k^2 \rangle - 2\langle k \rangle > 0.$$

(13.135)

In the present case

$$\langle k \rangle = \sum_{k=0}^{\infty} k p_k = \frac{1}{\zeta(\alpha)} \sum_{k=1}^{\infty} k^{-\alpha+1} = \frac{\zeta(\alpha-1)}{\zeta(\alpha)},$$

(13.136)

and

$$\langle k^2 \rangle = \sum_{k=0}^{\infty} k^2 p_k = \frac{1}{\zeta(\alpha)} \sum_{k=1}^{\infty} k^{-\alpha+2} = \frac{\zeta(\alpha-2)}{\zeta(\alpha)}.$$

(13.137)

Thus there is a giant component if

$$\zeta(\alpha-2) > 2\zeta(\alpha-1).$$

(13.138)

Figure 13.8 shows this inequality in graphical form. The two curves in the figure show the values of $\zeta(\alpha - 2)$ and $2\zeta(\alpha - 1)$ as functions of α and, as we can see, the inequality (13.138) is satisfied only for sufficiently low values of α, below the dotted line in the figure. In fact a numerical solution of the equation $\zeta(\alpha - 2) = 2\zeta(\alpha - 1)$ indicates that the network will have a giant component only for $\alpha < 3.4788$ ..., a result first given by Aiello *et al.* [9] in 2000.

In practice this result is of only limited utility because it applies only for the pure power law. In general, other distributions with power-law tails but different behavior for low k will have different thresholds at which the giant component appears. There is however a general result we can derive that applies to all distributions with power-law tails. In Section 8.4.2 we noted that the second moment $\langle k^2 \rangle$ diverges for any distribution with a power-law tail with exponent $\alpha \leq 3$, while the first moment $\langle k \rangle$ remains finite so long as $\alpha > 2$. This means that Eq. (13.135) is always satisfied for any configuration model with a power-law tail to its degree distribution so long as α lies in the range $2 < \alpha \leq 3$, and hence there will always be a giant component no matter what else the distribution does. For $\alpha > 3$, on the other hand, there may or may not be a giant component, depending on the precise functional form of the degree distribution. (For $\alpha \leq 2$ it turns out that there is always a giant component, although more work is needed to demonstrate this.) Note that, as discussed in Section 8.4, most observed values of α for real-world networks lie in the range $2 < \alpha \leq 3$ and hence we tentatively expect such networks to have a giant component, although we must also bear in mind that the configuration model is a simplified model of a network and is not necessarily a good representation of any specific real-world network.

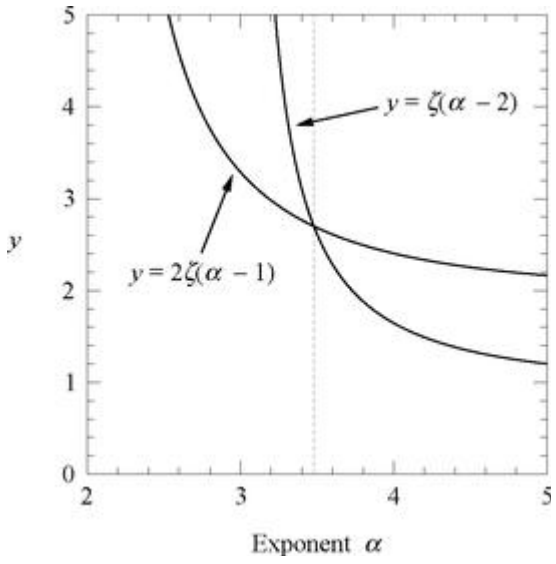


Figure 13.8: Graphical solution of Eq. (13.138). The configuration model with a pure power-law degree distribution (Eq. (13.133)) has a giant component if $\zeta(\alpha - 2) > 2\zeta(\alpha - 1)$. This happens for values of α below the crossing point of the two curves.

Returning to the pure power law let us calculate the size S of the giant component, when there is one. The fundamental generating functions $g_0(z)$ and $g_1(z)$ for the power-law distribution are given by Eqs. (13.55) and (13.56), which we repeat here for convenience:

$$g_0(z) = \frac{\mathrm{Li}_\alpha(z)}{\zeta(\alpha)}, \qquad g_1(z) = \frac{\mathrm{Li}_{\alpha-1}(z)}{z\zeta(\alpha-1)}.$$

(13.139)

Here $\zeta(\alpha)$ is the Riemann zeta function again and $\mathrm{Li}_\alpha(z)$ is the polylogarithm

$$\mathrm{Li}_\alpha(z) = \sum_{k=1}^{\infty} k^{-\alpha} z^k.$$

(13.140)

(See Eq. (13.15).) Now the crucial equation (13.91) for the probability $u = h_1(1)$ reads

$$u = \frac{\mathrm{Li}_{\alpha-1}(u)}{u\zeta(\alpha-1)} = \frac{\sum_{k=1}^{\infty} k^{-\alpha+1} u^k}{u\zeta(\alpha-1)} = \frac{\sum_{k=0}^{\infty} (k+1)^{-\alpha+1} u^k}{\zeta(\alpha-1)},$$

(13.141)

where we have used the explicit definition of the polylogarithm for clarity.

In general there is no closed-form solution for this equation, but we do notice some interesting points. In particular, note that the sum in the numerator is strictly positive for $u \geq 0$, which means that if $\zeta(\alpha - 1)$ diverges we will get a solution $u = 0$. And indeed $\zeta(\alpha - 1)$ does diverge. It diverges at $\alpha = 2$ and all values below, as one can readily verify from the definition, Eq. (13.134).[199] Thus for $\alpha \leq 2$ we have $u = 0$ and Eq. (13.90) then tells us that the giant component has size $S = 1 - g_0(0) = 1 - p_0$. However, for our particular choice of degree distribution, Eq. (13.133), there are no vertices with degree zero, and hence $p_0 = 0$ and $S = 1$. That is, the giant component fills the entire network and there are no small components at all!

Technically, this statement is not quite correct. There is always some chance that, for instance, a vertex of degree 1 will connect to another vertex of degree 1, forming a small component. What we have shown is that the probability that a randomly chosen vertex belongs to a small component is zero in the limit of large n, i.e., that what small components there are fill a fraction of the network that vanishes as $n \to \infty$. In the language used by mathematicians, a randomly chosen vertex "almost surely" belongs to the giant component, meaning it is technically possible to observe another outcome, but the probability is vanishingly small.

Thus our picture of the pure power-law configuration model is one in which there is a giant component for values of $\alpha < 3.4788$ ... and that giant component fills essentially the entire network when $\alpha \leq 2$. In the region between $\alpha = 2$ and $\alpha = 3.4788$ there is a giant component but it does not fill the whole network and some portion of the network consists of small component. If $\alpha > 3.4788...$ there are only small components. As a confirmation of this picture, Fig. 13.9 shows the size of the giant component extracted from a numerical solution of Eq. (13.141).[200] As we can see it fits nicely with the picture described above.

We could in principle take our calculations further, calculating, for instance, the mean size of the small components in the region $\alpha > 2$ using Eq. (13.112), or the entire distribution of their sizes using Eq. (13.127).

13.11 DIRECTED RANDOM GRAPHS

In this chapter we have studied random graph models that go a step beyond the Poisson random graph of Chapter 12 by allowing us to choose the degree distribution of our model network. This introduces an additional level of realism to the model that makes it substantially more informative. It is, however, only a first step. There are many other features we can add to the model to make it more realistic still. We can for instance create random graph models of networks with assortative (or disassortative) mixing [237], bipartite structure [253], or clustering [247]. All of these models are still exactly solvable in the limit of large system size, although the solutions are more complicated than for the models we have seen in this chapter. For instance, in the case of the random graph with assortative mixing the fundamental generating function $g_1(z)$ becomes a vector, the corresponding equation (13.86) for the distribution of component sizes becomes a vector equation, and the condition for the existence of a giant component, Eq. (13.76), becomes a condition on the determinant of a matrix.

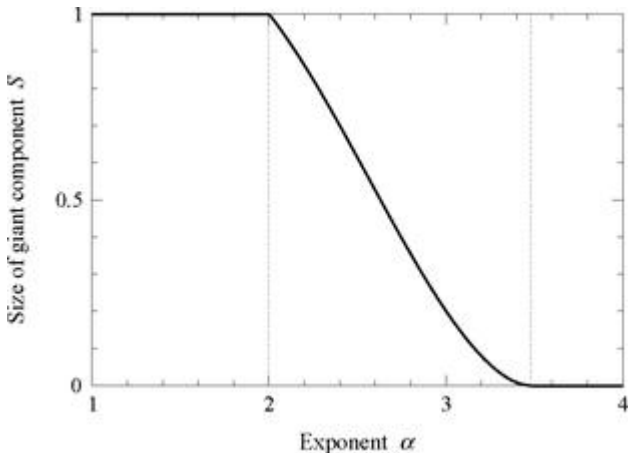


Figure 13.9: Size of the giant component for the configuration model with a power-law degree distribution. This plot shows the fraction of the network filled by the giant component as a function of the exponent α of the power law, calculated by numerical solution of Eqs. (13.91) and (13.141). The dotted lines mark the value $\alpha = 2$ below which the giant component has size 1 and the value $\alpha = 3.4788$ above which there is no giant component.

We will not go into detail on all of the many random graph models that have been proposed and studied, but in this section we take a look at one case, that of the directed random graph, as an example of the types of calculation that are possible.

13.11.1 GENERATING FUNCTIONS FOR DIRECTED GRAPHS

As discussed in Section 6.4, many networks, including the World Wide Web, metabolic networks, food webs, and others, are directed. The configuration model can be generalized to directed networks in a straightforward fashion, although the generalization displays some new behaviors not seen in the undirected case. Our presentation follows that of Refs. [100] and [253].

To create a directed equivalent of the configuration model, we must specify a double degree sequence, consisting of an in-degree j_i and an out-degree k_i for each vertex i. We can think of these as specifying the numbers of ingoing and outgoing stubs of edges at each vertex. Then we create a network by repeatedly choosing pairs of stubs—one ingoing and one outgoing—uniformly at random and connecting them to make directed edges, until no unused stubs remain. The result is a matching of the stubs drawn uniformly at random from the set of all possible matchings, just as in the configuration model, and the model itself is defined to be the ensemble of such directed networks in which each matching appears with equal probability. (The only small catch is that we must make sure that the total number of ingoing and outgoing stubs is the same, so that none are left over at the end of the process. We will assume this to be the case in the following developments.)

The probability that a particular outgoing stub at vertex w attaches to one of the j_v ingoing stubs at vertex v is

$$\frac{j_v}{\sum_i j_i} = \frac{j_v}{m},$$

(13.142)

where m is the total number of edges and we have made use of Eq. (6.26). Since the total number of outgoing stubs at w is k_ω, the total expected number of directed edges from vertex w to vertex v is then $j_v k_w/m$, which is also the probability of an edge from w to v in the limit of large network size, provided the network is sparse. This is similar to the corresponding result, Eq. (13.32), for the undirected configuration model, but not identical—notice that there is no factor of two now in the denominator.

As in the undirected case we can, if we prefer, work with the degree distribution, rather than the degree sequence. As discussed in Section 8.3, the most correct way to describe the degree distribution of a directed network is by a joint distribution: we define p_{jk} to be the fraction of vertices in the network that have in-degree j and out-degree k. This allows for the possibility that the in–and out-degrees of vertices are correlated. For instance, it would allow us to represent a network in which the in–and out-degrees of each vertex were exactly equal to one another.[201] (This is rather an extreme example, but it demonstrates the point.)

The joint degree distribution can be captured in generating function form by defining a *double generating function* $g_{00}(x,y)$ thus:

$$g_{00}(x,y) = \sum_{j,k=0}^{\infty} p_{jk} x^j y^k.$$

(13.143)

(The two subscript zeros are the equivalent for the double generating function of the subscript zero in our previous generating function $g_0(z)$ for the undirected network.) As in the undirected case, the generating function $g_{00}(x, y)$ captures all the information contained in the degree distribution. Given the generating function we can reconstruct the degree distribution by differentiating:

$$p_{jk} = \frac{1}{j!\,k!} \frac{\partial^j \partial^k g_{00}}{\partial x^j \partial y^k}\bigg|_{x,y=0}.$$

(13.144)

This is the equivalent for the directed case of Eq. (13.2) in the undirected case.

Just as in the undirected case the generating function satisfies certain conditions. First, since the degree distribution must be normalized according to $\sum_{jk} p_{jk} = 1$, the generating function satisfies

$$g_{00}(1,1) = 1.$$

(13.145)

Second, the average in–and out-degrees are given by

$$\langle j \rangle = \sum_{jk=0}^{\infty} j p_{jk} = \frac{\partial g_{00}}{\partial x}\bigg|_{x,y=1},$$

(13.146)

$$\langle k \rangle = \sum_{jk=0}^{\infty} k p_{jk} = \frac{\partial g_{00}}{\partial y}\bigg|_{x,y=1}.$$

(13.147)

In a directed graph, however, the average in–and out-degrees are equal—see Eq. (6.27)—so j =

k and

$$\left.\frac{\partial g_{00}}{\partial x}\right|_{x,y=1} = \left.\frac{\partial g_{00}}{\partial y}\right|_{x,y=1}.$$

(13.148)

For convenience we will denote the average in-degree and out-degree by c in the equations that follow. Thus $j = k = c$.

We can also write down generating functions for the excess degree distribution of vertices reached by following an edge in the network. There are two different ways of following a directed edge—either forward or backward. Consider first the forward case. If we follow an edge forward to the vertex it points to, then the probability of reaching a particular vertex will be proportional to the number of edges pointing to that vertex, i.e., to its in-degree. Thus the joint degree distribution of such a vertex is proportional not to p_{jk} but to jp_{jk}. As before, we will be interested primarily in the number of edges entering and leaving a vertex other than the one we arrived along. If j and k denote these numbers then the total in-degree is $j + 1$ and the total out-degree is just k, so the distribution we want is proportional to $(j + 1)\, p_{j+1,k}$ or, correctly normalized, $(j + 1)p_{j+1,k}/c$. The double generating function for this excess degree distribution is then

$$g_{10}(x,y) = \frac{\sum_{jk}(j+1)p_{j+1,k}x^j y^k}{\sum_{jk}(j+1)p_{j+1,k}} = \frac{\sum_{jk} jp_{jk}x^{j-1}y^k}{\sum_{jk} jp_{jk}} = \frac{1}{c}\frac{\partial g_{00}}{\partial x}.$$

(13.149)

The backward case is similar. The appropriate excess degree distribution for the vertex from which an edge originates is $(k + 1)p_{j,k+1}/c$ and has generating function

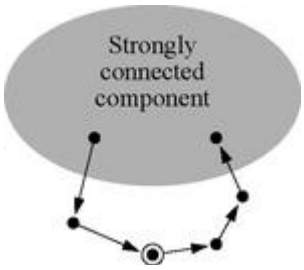
$$g_{01}(x,y) = \frac{\sum_{jk}(k+1)p_{j,k+1}x^j y^k}{\sum_{jk}(k+1)p_{j,k+1}} = \frac{1}{c}\frac{\partial g_{00}}{\partial y}.$$

(13.150)

13.11.2 GIANT COMPONENTS

A directed graph has various different types of component, as discussed in Sections 6.11.1 and 8.1.1, including strongly and weakly connected components, in-components, and out-components. (Take a look at the "bow tie" diagram, Fig. 8.2 on page 240, for a reminder of the definitions of the components.) In general there can be both small and giant components of each of these types. Let us look at the giant components.

A strongly connected component is a set of vertices in which every vertex is reachable by a directed path from every other in the set. To put that a different way, for a vertex to belong to a strongly connected component at least one of its outgoing edges must lead to another vertex from which there is a path to the strongly connected component, and at least one of its ingoing edges must lead from a vertex to which there is path from the strongly connected component (see figure).



A vertex belongs to a strongly connected component if it has a directed path to the component and another from the component.

Let v be the probability that the vertex to which a randomly chosen edge in our graph leads has no directed path to the giant strongly connected component. For this to happen, it must be that none of the other outgoing edges from that vertex themselves have such a path. If the vertex has out-degree k, this happens with probability v^k. But j and k are distributed according to the excess degree distribution $(j+1)p_{j+1,k}/c$ and hence, averaging over both, we find that

$$v = \frac{1}{c}\sum_{j,k=0}^{\infty}(j+1)p_{j+1,k}v^k = g_{10}(1,v).$$

(13.151)

Similarly, consider the vertex from which a randomly chosen edge originates and let u be the probability that there is no path from the giant strongly connected component to that vertex. Then u is the solution to

$$u = g_{01}(u,1).$$

(13.152)

Now consider a vertex with in-degree j and out-degree k. The probability that there is no path to the vertex from the giant strongly connected component via any of the vertex's j ingoing edges is u^j and the probability that there *is* such a path is $1 - u^j$. Similarly the probability that there is a path from the vertex to the giant strongly connected component is $1 - v^k$. And the probability that there are both—and hence that the vertex itself belongs to the giant strongly connected component—is the product of these two, or $(1 - u^j)(1 - v^k)$. Averaging this expression over the joint distribution of j and k we then find that the average probability S_s that a vertex lies in the giant strongly connected component, which is also the size of the giant strongly connected component measured as a fraction of the network size, is

$$\begin{aligned} S_s &= \sum_{j,k=0}^{\infty} p_{jk}(1-u^j)(1-v^k) \\ &= \sum_{j,k=0}^{\infty} p_{jk} - \sum_{j,k=0}^{\infty} p_{jk}u^j - \sum_{j,k=0}^{\infty} p_{jk}v^k + \sum_{j,k=0}^{\infty} p_{jk}u^j v^k \\ &= 1 - g_{00}(u,1) - g_{00}(1,v) + g_{00}(u,v), \end{aligned}$$

(13.153)

with u and v given by Eqs. (13.151) and (13.152).

As discussed in Section 6.11, each strongly connected component in a network also has an in-component and an out-component associated with it—the sets of vertices from which it can be reached, and which can be reached from it. The in–and out-components of the giant strongly connected component are usually called the giant in–and out-components. By their definition, both are supersets of the giant strongly connected component itself, and we can calculate the size of both for our directed random graph. In fact, we have performed most of the calculation already.

A vertex with out-degree k fails to belong to the giant in-component only if none of its outgoing edges leads to a vertex that has a path to the strongly connected component. This happens with probability v^k, where v is as above. Averaging over j and k, we find the probability S_i that the vertex does belong to the giant in-component (which is also the size of the giant in-component) to be

$$S_i = 1 - \sum_{j,k=0}^{\infty} p_{jk}v^k = 1 - g_{00}(1,v).$$

(13.154)

Similarly the size of the giant out-component is given by

$$S_o = 1 - g_{00}(u,1).$$

(13.155)

Using these results we can also write an expression for the combined size of the giant strongly connected component and its in–and out-components—the entire "bow tie" in Fig. 8.2. Since the giant in–and out-components both include the giant strongly connected component as a subset, their sum is equal to the size of the whole bow tie except that it counts the strongly connected part twice. Subtracting S_s to allow for this overcounting we then find the size of the bow tie to be and the fraction of the network not in the bow tie is just $g_{00}(u, v)$. (We could have derived this result by more direct means, just by noting that a vertex not in the bow tie has a path neither to nor from the giant strongly connected component.)

$$S_i + S_o - S_s = 1 - g_{00}(u, v),$$

(13.156)

And what about the giant weakly connected component? A weakly connected component in a directed graph is a normal graph component of connected vertices in which we ignore the directions of all the edges. At first glance one might imagine that the size of the giant weakly connected component was just equal to the combined size $S_i + S_o - S_s$ of the in-, out-, and strongly connected components calculated above. This, however, is not correct because the definition of the giant weakly connected component includes some vertices that are not in the in-, out-, or strongly connected components. An example would be any vertex that is reachable from the giant in-component but that does not itself have a path to the strongly connected component and hence is not in the giant in-component. Thus the size of the giant weakly connected component is, in general, larger than $S_i + S_o - S_s$. Nonetheless, we can still calculate the size of the giant weakly connected component by an argument quite similar to the ones we have already seen.

A vertex belongs to the giant weakly connected component if any of its edges, ingoing or outgoing, are connected to a vertex in that component. Let u now be the probability that a vertex is not connected to the giant weakly connected component via the vertex at the other end of one of its ingoing edges and let v be the equivalent probability for an outgoing edge. Then the probability that a vertex with in-degree j and out-degree k is not in the giant weakly connected component is $u^j v^k$ and the probability that it is in the giant weakly connected component is $1 - u^j v^k$. Averaging over the joint distribution p_{jk} of the two degrees we then find that the size S_ω of the giant weakly connected component is

$$S_w = \sum_{jk} p_{jk} - \sum_{jk} p_{jk} u^j v^k = 1 - g_{00}(u, v).$$

(13.157)

We can derive the value of u by noting that the vertex at the end of an ingoing edge is not in the giant weakly connected component with probability $u^j v^k$ again, but with j and k being the numbers of edges excluding the edge we followed to reach the vertex. These numbers are distributed according to the appropriate excess degree distribution and, performing the average, we find that

$$u = g_{01}(u, v).$$

(13.158)

Similarly we can show that

$$v = g_{10}(u, v),$$

(13.159)

and Eqs. (13.157) to (13.159) between them give us our solution for the size of the giant weakly connected component.

13.11.3 THE APPEARANCE OF THE GIANT COMPONENTS

As in the undirected random graph, there may or may not be giant components in the direct random graph, depending on the degree distribution. We can derive conditions for the existence of the giant components using the machinery developed above. The calculation is easiest for the giant in–and out-components. Their size is given by Eqs. (13.154) and (13.155). Given that $g_{00}(1, 1) = 1$ (Eq. (13.145)), these equations give a non-zero size only if u or v is less than 1. Looking at Eq. (13.151) for the value of v we see a similar situation to that depicted in Fig. 13.6: we can have a solution with $v < 1$, and hence a giant in-component, only if

$$\left.\frac{\partial g_{10}}{\partial y}\right|_{x,y=1} > 1,$$

(13.160)

or equivalently if

$$\left.\frac{\partial^2 g_{00}}{\partial x \partial y}\right|_{x,y=1} > c,$$

(13.161)

where we have made use of Eqs. (13.148) and (13.149). Similarly we can have a giant out-component only if

$$\left.\frac{\partial g_{01}}{\partial x}\right|_{x,y=1} > 1,$$

(13.162)

or equivalently,

$$\left.\frac{\partial^2 g_{00}}{\partial x \partial y}\right|_{x,y=1} > c.$$

(13.163)

Interestingly, Eqs. (13.161) and (13.163) are identical, meaning that the conditions for the giant in–and out-components to appear are the same. If there is a phase transition at which one appears, the other also appears at the exact same moment.[202]

We can express (13.161) directly in terms of the degree distribution if we want. Substituting from Eq. (13.143) we find that

$$\left.\frac{\partial^2 g_{00}}{\partial x \partial y}\right|_{x,y=1} = \sum_{j,k=0}^{\infty} jkp_{jk}$$

(13.164)

and hence the giant in–and out-components appear if

$$\sum_{j,k=0}^{\infty} jkp_{jk} > c.$$

(13.165)

If we prefer we can write $c = \sum_j jp_{jk} = \sum_j kp_{jk}$ to give the alternative form

$$\sum_{j,k=0}^{\infty} (2jk - j - k)p_{jk} > 0.$$

(13.166)

This result is the equivalent for a directed network of Eq. (13.76) for the undirected case.

The calculation for the giant strongly connected component is similar. From Eq. (13.153) we see that $S_s = 0$ unless at least one of u and v is non-zero, so the condition for the existence of a giant strongly connected component is the same as for the in–and out-components, Eq. (13.166). In other words, the giant in-, out-, and strongly connected components all appear or disappear simultaneously.

The giant weakly connected component, however, is different. It is possible for there to be a giant weakly connected component in a network but no giant strongly connected component and hence the condition for the existence of a giant weakly connected component must be different from that for the other giant components. For instance, a network in which all vertices have either only ingoing edges or only outgoing edges can have a giant weakly connected component but trivially has no strongly connected component of size greater than one, since there are only paths to or from each vertex, but not both. Weakly connected components, however, are generally of less interest than strongly connected ones, and the calculation of the condition for the existence of the

giant weakly connected component is non-trivial, so we leave it as an exercise for the motivated reader and move on to other things.

13.11.4 SMALL COMPONENTS

We can also calculate the distribution of small components in a directed random graph. In fact the distribution of small strongly connected components is trivial: there aren′t any. Or more properly the probability that a randomly chosen vertex belongs to a strongly connected component of size greater than one other than the giant strongly connected component is zero in the limit of large network size. To see this, recall that the small components in the undirected configuration model take the form of trees (see Section 13.7). If we consider a small strongly connected component in a directed network and ignore the directions of its edges, then the same argument we used before indicates that the resulting subgraph will also be a tree. But a tree has no loops in it, which leads to a contradiction because a strongly connected component must have loops—the paths in either direction between any pair of vertices form a loop. Thus, we conclude, there cannot be any small strongly connected components of size greater than one in the network.

In fact, this is not precisely true. In a random network there is always some chance that, for example, two vertices will each have a directed edge to the other, forming a strongly connected component of two vertices. In the limit of large n, however, the probability that a randomly chosen vertex belongs to such a component tends to zero. A detailed calculation shows that on average there is only a constant number of short loops in the network and their density vanishes as $1/n$ in the limit of large network size.

There can however be small in–and out-components. In a directed network, each strongly connected component has its own in–and out-components. In the present model, as we have said, we have no small strongly connected components, other than single vertices, so the component structure consists of the giant in–and out-components and then a large number of small in–and out-components for single vertices. Let us ask what the probability is that a randomly chosen vertex has a small out-component of size s, i.e., that there are s vertices including itself that can be reached by directed paths starting from the vertex. We can calculate the distribution of sizes by the same method we used for the undirected case. We define a generating function $h_1(y)$ for the distribution of the size of the out-component of a vertex reached by following an edge in the forward direction, which then satisfies an equation of the form of Eq. (13.86), except that the generating function g_1 for the excess degree distribution is replaced by the corresponding generating function for the directed network, Eq. (13.149), giving

$$h_1(y) = y g_{10}(1, h_1(y)).$$

(13.167)

And the generating function $h_0(y)$ for the size of the out-component to which a randomly chosen vertex belongs is then

$$h_0(y) = y g_{00}(1, h_1(y)).$$

(13.168)

We can write similar equations for in-components too and, armed with these equations, we can find the average size of the in–or out-component to which a vertex belongs, or even find the entire distribution of component sizes using the equivalent of Eq. (13.127).

PROBLEMS

13.1 Consider the binomial probability distribution $p_k = \binom{n}{k} p^k (1-p)^{n-k}$.

a. Show that the distribution has probability generating function $g(z) = (pz + 1 - p)^n$.
b. Find the first and second moments of the distribution from Eq. (13.25) and hence show that the variance of the distribution is $\sigma^2 = np(1-p)$.
c. Show that the sum of two numbers drawn independently from the same binomial distribution is distributed according to $\binom{2n}{k} p^k (1-p)^{2n-k}$.

13.2 Consider a configuration model in which every vertex has the same degree k.

a. What is the degree distribution p_k? What are the generating functions g_0 and g_1 for the degree distribution and the excess degree distribution?
b. Show that the giant component fills the whole network for all $k \geq 3$.
c. What happens when $k = 1$?
d. When $k = 2$ show that in the limit of large n the probability π_s that a vertex belongs to a component of size s is given by $\pi_s = 1/[2\sqrt{n(n-s)}]$.

13.3 Consider the configuration model with exponential degree distribution $p_k = (1 - e^{-\lambda})e^{-\lambda k}$ with $\lambda > 0$, so that the generating functions $g_0(z)$ and $g_1(z)$ are given by Eq. (13.130).

a. Show that the probability u of Eq. (13.91) satisfies the cubic equation

$$u^3 - 2e^{\lambda}u^2 + e^{2\lambda}u - (e^{\lambda} - 1)^2 = 0.$$

b. Noting that $u = 1$ is always a trivial solution of this equation, show that the non-trivial solution corresponding to the existence of a giant component satisfies the quadratic equation $u^2 - (2e^{\lambda} - 1)u + (e^{\lambda} - 1)^2 = 0$, and hence that the size of the giant component, if there is one, is

$$S = \tfrac{3}{2} - \sqrt{e^{\lambda} - \tfrac{3}{4}}.$$

c. Show that the giant component exists only if $\lambda < \ln 3$.

13.4 Equation (13.74) tells us on average how many vertices are a distance d away from a given vertex.

a. Assuming that this expression works for all values of d (which is only a rough approximation to the truth), at what value of d is this average number of vertices equal to the number n in the whole network?

b. Hence derive a rough expression for the diameter of the network in terms of c_1 and c_2, and so argue that configuration model networks display the small-world effect in the sense that typical geodesic distances between vertices are O(log n).

13.5 Consider a network model in which edges are placed independently between each pair of vertices i, j with probability $p_{ij} = Kf_if_j$, where K is a constant and f_i is a number assigned to vertex i. Show that the expected degree c_i of vertex i within the model is proportional to f_i, and hence that the only possible choice of probability with this form is $p_{ij} = c_ic_j/2m$, as in the model of Section 13.2.2.

13.6 As described in Section 13.2, the configuration model can be thought of as the ensemble of all possible matchings of edge stubs, where vertex i has k_i stubs. Show that for a given degree sequence the number Ω of matchings is

$$\Omega = \frac{(2m)!}{2^m m!},$$

which is independent of the degree sequence.

13.7 Consider the example model discussed in Section 13.8.1, a configuration model with vertices of degree three and less only and generating functions given by Eqs. (13.94) and (13.95).

a. In the regime in which there is no giant component, show that the average size of the component to which a randomly chosen vertex belongs is

$$\langle s \rangle = 1 + \frac{(p_1 + 2p_2 + 3p_3)^2}{p_1 - 3p_3}.$$

b. In the same regime find the probability that such a vertex belongs to components of size 1, 2, and 3.

13.8 Consider a directed random graph of the kind discussed in Section 13.11.

a. If the in–and out-degrees of vertices are uncorrelated, i.e., if the joint in/out-degree distribution p_{jk} is a product of separate functions of j and k, show that a giant strongly connected component exists in the graph if and only if $c(c - 1) > 0$, where c is the mean degree, either in or out.
b. In real directed graphs the degrees are usually correlated (or anti-correlated). The correlation can be quantified by the covariance ρ of in–and out-degrees. Show that in the presence of correlations, the condition above for the existence of a giant strongly connected component generalizes to $c(c - 1) + \rho > 0$.
c. In the World Wide Web the in–and out-degrees of the vertices have a measured covariance of about $\rho = 180$. The mean degree is around $c = 4.6$. On the basis of these numbers, do we expect the Web to have a giant strongly connected component?