CHAPTER 15

OTHER NETWORK MODELS

A brief introduction to two specialized network models, the small-world model and the exponential random graph

THE RANDOM graph and preferential attachment models of previous chapters are the most widely studied of network models, but they are not the only ones. Many other models have been proposed, either as a way of shedding light on specific observed features of networks or as tools to help in the analysis of network data. In this chapter we describe briefly two of the best-known additional types of network models, the small-world model and exponential random graphs.

15.1 THE SMALL-WORLD MODEL

One of the least well-understood features of real-world networks is transitivity, the propensity for two neighbors of a vertex also to be neighbors of one another. (See Section 7.9 for an introduction to the phenomenon of transitivity.) Neither the random graph models of Chapters 12 and 13 nor the models of network growth discussed in Chapter 14 generate networks with any significant level of transitivity, as quantified by the clustering coefficient, Eq. (7.41). The Poisson random graph of Chapter 12, for instance, has a clustering coefficient $c / (n - 1)$, where c is the mean degree of a vertex (see Eq. (12.11)). Thus the clustering coefficient vanishes as n becomes large for constant c. In practice, as discussed in Section 7.9, this often results in values of the clustering coefficient that are orders of magnitude smaller than those observed in real networks.

It is not that difficult to come up with a network model that does have a high clustering coefficient. For example, a simple triangular lattice, as shown in Fig. 15.1, has significant transitivity.

There are twice as many triangles in such a lattice as there are vertices, for a total of $2n$ triangles in a network of n vertices. At the same time there are $\binom{6}{2} = 15$ connected triples for each vertex, so, following Eq. (7.41), the clustering coefficient is

$$C = \frac{(\text{number of triangles}) \times 3}{(\text{number of connected triples})} = \frac{2n \times 3}{15n} = \frac{2}{5} = 0.4. \tag{15.1}$$

A value of 0.4 is comparable with the clustering coefficients measured for many social networks (see Section 7.9 again). Moreover, this value does not depend on the size of the network, as the value for the random graph (and many other models) does, so it remains large even as the network size diverges.

Another simple model network with high transitivity is depicted in Fig. 15.2a. Unlike the triangular lattice, this model allows the value of the clustering coefficient to be varied. In this model the vertices are arranged on a one-dimensional line, and each vertex is connected by an edge to the c vertices nearest to it, where for consistency c should be an even number. To make analytic treatment easier, we can apply periodic boundary conditions to the line, effectively bending it around into a circle, as in Fig. 15.2b.

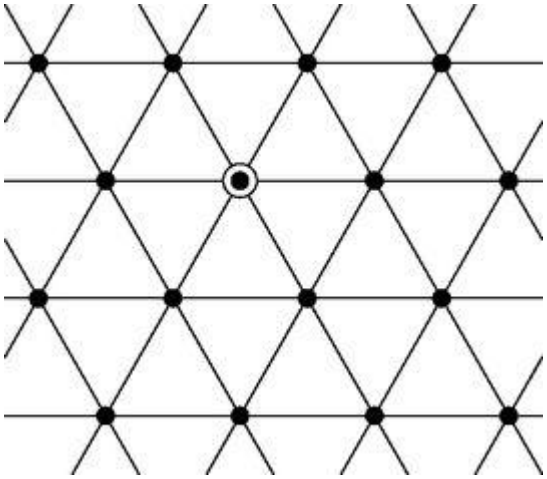


Figure 15.1: A triangular lattice. Any vertex in a triangular lattice, such as the one highlighted here, has six neighbors and hence $\binom{6}{2} = 15$ pairs of neighbors, of which six are connected by edges, giving a clustering coefficient of $\frac{6}{15} = 0.4$ for the whole network, regardless of size.

To calculate the number of triangles in such a network, we observe that a trip around any triangle must consist of two steps in the same direction around the circle—say clockwise—followed by one step back to close the triangle. The number of triangles per vertex in the whole network is then equal to the number of such triangles that start from any given point.

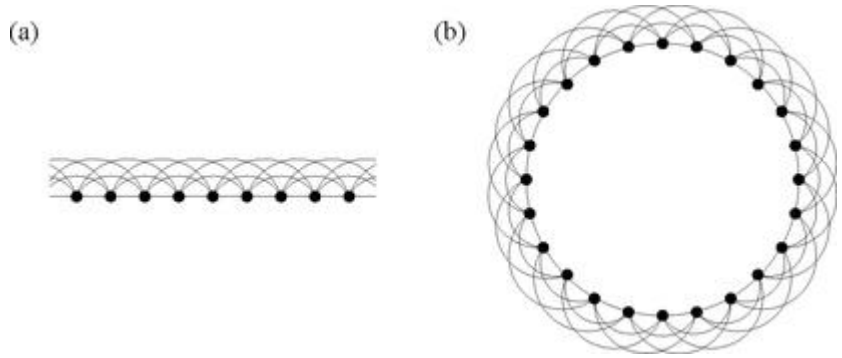
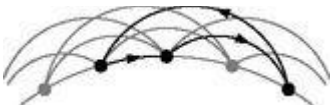


Figure 15.2: A simple one-dimensional network model. (a) Vertices are arranged on a line and each is connected to its c nearest neighbors, where $c = 6$ in this example. (b) The same network with periodic boundary conditions applied, making the line into a circle.



Traversing a “triangle” in our circle model means taking two steps forward around the circle and

one step back.

Note, however, that the third and final step in the triangle can go at most $\frac{1}{2}c$ units or lattice spacings around the circle, since this is the length of the longest link in the network. And the number of ways to choose the two steps forward is simply the number of distinct ways of choosing the target vertices for those steps from the $\frac{1}{2}c$ possibilities, which is $\binom{c/2}{2} = \frac{1}{4}c(\frac{1}{2}c-1)$. .Thus the total number of triangles is $\frac{1}{4}nc(\frac{1}{2}c-1)$

The number of connected triples centered on each vertex is just $= \binom{c}{2} = \frac{1}{2}c(c-1)$ and hence the total number of connected triples is $\frac{1}{2}nc(c-1)$.

Putting these results together, the clustering coefficient for the complete network is

$$C = \frac{\frac{1}{4}nc(\frac{1}{2}c-1)\times 3}{\frac{1}{2}nc(c-1)} = \frac{3(c-2)}{4(c-1)}.$$

(15.2)

As c is varied, this clustering coefficient ranges from zero for $c = 2$ up to a maximum of $\frac{3}{4}$ when $c \to \infty$. And, as with the triangular lattice, the value does not fall off with increasing network size, since Eq. (15.2) is independent of n.

While this simple "circle model" and the triangular lattice both give large values of the clustering coefficient, they are clearly unsatisfactory in other respects as models of networks. One obvious problem is the degree distribution. The circle model, for instance, gives all vertices the same degree c. In the language of graph theory the model generates a regular graph, which is entirely unlike most real-world networks with their broad distributions of vertex degree. This problem however could quite easily be solved by making a circle of vertices with varying degrees instead of constant ones.

A more serious problem with models of this type is that they are "large worlds"—they don't display the small-world effect characteristic of essentially every observed network in the real world and discussed previously in Sections 3.6 and 8.2. The small-world effect is the observation that the geodesic or shortest-path distance between most pairs of vertices in a network is small—typically just a few steps even in networks with billions of vertices such as the acquaintance network of the entire world population.

The shortest distance between two vertices in the circle model above is straightforward to calculate: the farthest one can move around the ring in a single step is $\frac{1}{2}c$ lattice spacings, so two vertices m lattice spacings apart are connected by a shortest path of $2m/c$ steps.[232] Averaging over the complete range of m from 0 to $\frac{1}{2}n$ then gives a mean shortest path of $n/2c$. In a network such as the acquaintance network of the world, with $n = O(10^9)$ people each acquainted with, say, $c = O(10^3)$ others, this expression yields an average shortest path length on the order of a million steps, which is wildly off the mark—a more realistic figure would be six or maybe ten, but not a million.

By contrast, the random graph studied in Chapter 12 *does* capture the small-world effect rather well (as indeed do most of the other network models discussed in previous chapters). As shown in Section 12.7, the typical shortest path between connected vertices in a random graph has length about $\ln n / \ln c$ which has a value on the order of $\frac{9}{3} = 3$ for the acquaintance network above. On the other hand, as we have said, the random graph has an unrealistically low clustering coefficient.

Thus we have two models, our simple circle model and the random graph, that between them each capture one property of real networks—high transitivity and short path lengths—but neither captures both. This leads us to ask whether it is possible to create a hybrid of the two that, like real-world networks, displays both high transitivity and short path lengths simultaneously. The *small-world model*, proposed in 1998 by Watts and Strogatz [323], does exactly this.

The small-world model, in its original form, interpolates between our circle model and the random graph by moving or *rewiring* edges from the circle to random positions. The detailed structure of the model is shown in Fig. 15.3a. Starting with a circle model of n vertices in which every vertex has degree c, we go through each of the edges in turn and with some probability p we remove that edge and replace it with one that joins two vertices chosen uniformly at random.[233] The randomly placed edges are commonly referred to as *shortcuts* because, as shown in Fig. 15.3a, they create shortcuts from one part of the circle to another.

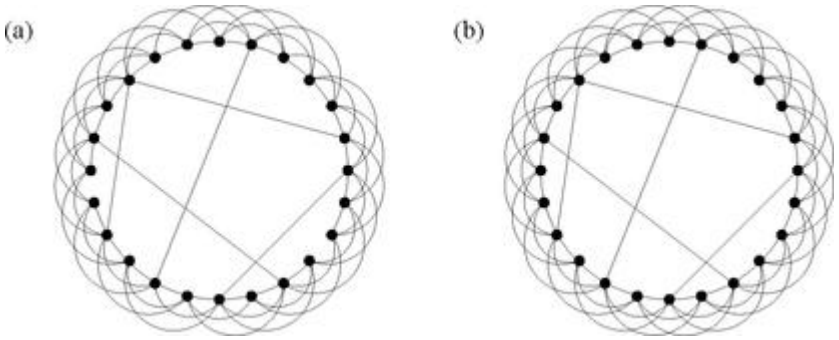


Figure 15.3: Two versions of the small-world model. (a) In the original version of the small-world model, edges are with independent probability p removed from the circle and placed between two vertices chosen uniformly at random, creating shortcuts across the circle as shown. In this example $n = 24$, $c = 6$, and $p = 0.07$, so that 5 out of 72 edges are "rewired" in this fashion. (b) In the second version of the model only the shortcuts are added and no edges are removed from the circle.

The parameter p in the small-world model controls the interpolation between the circle model and the random graph. When $p = 0$ no edges are rewired and we retain the original circle. When $p = 1$ all edges are rewired to random positions and we have a random graph. For intermediate values of p we generate networks that lie somewhere in between. Thus for $p = 0$ the small-world model shows clustering (so long as $c > 2$—see Eq. (15.2)) but no small-world effect. For $p = 1$ it does the reverse. The crucial point about the model is that as p is increased from zero the clustering is maintained up to quite large values of p while the small-world behavior, meaning short average path lengths, already appears for quite modest values of p. As a result there is a substantial range of intermediate values for which the model shows both effects simultaneously, thereby demonstrating that the two are in fact entirely compatible and not exclusive at all.

Unfortunately, it is hard to demonstrate this result rigorously because the small-world model as defined above is difficult to treat by analytic means. For this reason we will in this chapter study a slight variant of the model, which is easier to treat [254]. In this variant, shown in Fig. 15.3b, edges are added between randomly chosen vertex pairs just as before, but no edges are removed from the original circle. This leaves the circle intact, which makes our calculations much simpler. For ease of comparison with the original small-world model, the definition of the parameter p is kept the same: for every edge in the circle we add with independent probability p an additional shortcut between two vertices chosen uniformly at random.[234]

A downside of this version of the model is that it no longer becomes a random graph in the limit $p = 1$. Instead it becomes a random graph plus the original circle. This, however, turns out not to be a significant problem, since most of the interest in the model lies in the regime where p is small and in this regime the two models differ hardly at all; the only difference is the presence in the second variant of a small number of edges around the circle that would be absent in the first, having been rewired. Henceforth, we will study the variant model in which no edges are removed and we will refer to it, as others have, as the small-world model, although the reader should bear in mind that there are two slightly different models that carry this name.

15.1.1 DEGREE DISTRIBUTION

In the circle model described in the last section every vertex has the same degree c—the network is a regular graph. Once we add shortcuts to the circle to make the small-world model, the degree of a vertex is c plus the number of shortcut edges attached to it. The definition of the small-world model says that for each of the non-shortcut edges around the circle, of which there are $\frac{1}{2}nc$, we add a shortcut with probability p at a random location, so that there are $\frac{1}{2}ncp$ shortcuts on average and ncp *ends* of shortcuts. This means that cp shortcuts on average end at any particular vertex. And the specific number s of shortcuts attached to any one vertex is Poisson distributed with mean cp thus:

$$p_s = \mathrm{e}^{-cp}\frac{(cp)^s}{s!},$$

(15.3)

The total degree of a vertex is $k = s + c$. Putting $s = k - c$ into Eq. (15.3) then gives us the degree distribution of the small-world model:

$$p_k = \mathrm{e}^{-cp}\frac{(cp)^{k-c}}{(k-c)!}$$

(15.4)

for $k \geq c$ and $p_k = 0$ if $k < c$.

Figure 15.4 shows the form of this distribution for $c = 6$, $p = \frac{1}{2}$. As we can see, the distribution has an unusual peaked shape with a lower cut-off, quite unlike the degree distributions we saw for real networks in Section 8.3. In this respect, therefore, the small-world model does not mimic well the structure of networks in the real world. On the other hand, the model was never intended to mimic real-world degree distributions. What it does do well is mimic the clustering and short path lengths seen in real networks.

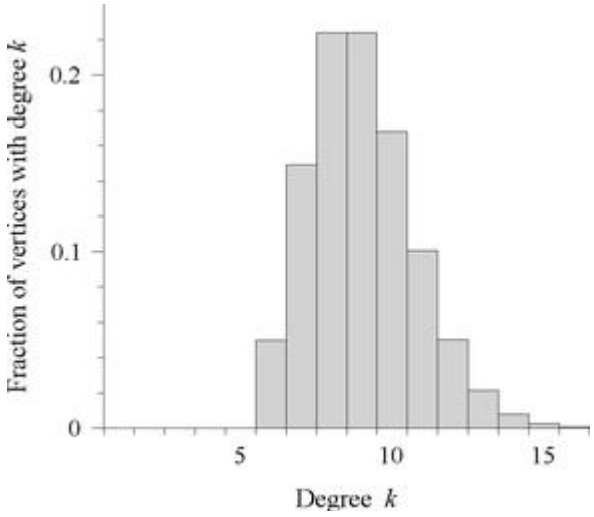


Figure 15.4: The degree distribution of the small-world model. The frequency distribution of vertex degrees in a small-world model with parameters $c = 6$ and $p = \frac{1}{2}$.

15.1.2 CLUSTERING COEFFICIENT

The clustering coefficient C is defined by Eq. (7.41), which we reproduce here:

$$C = \frac{(\text{number of triangles}) \times 3}{(\text{number of connected triples})}.$$

(15.5)

To evaluate C for the small-world model we need to calculate the numbers of triangles and connected triples in the network. Let us start with the former.

Since the underlying circle in the model is unchanged by the addition of shortcuts, every triangle in that circle, of which there are, as before, $\frac{1}{4}nc(\frac{1}{2}c-1)$, is still present. Some new triangles are also introduced by the shortcuts. For example, vertex pairs $\frac{1}{2}c+1$ to c steps apart on the circle are connected by one or more paths of length two, and if the same vertices are also connected by a shortcut those paths are turned into triangles.

The number of such paths of length two is clearly proportional to n—if we double the length of the circle we double the number of paths. The average number of shortcuts in the small-world model is, as we have said, $\frac{1}{2}ncp$ and there are $\binom{n}{2}$ places they can fall, meaning that any particular pair of vertices is connected with probability

$$\frac{\frac{1}{2}ncp}{\frac{1}{2}n(n-1)} = \frac{cp}{n-1},$$

(15.6)

or just cp / n in the limit of large n. The number of paths of length two that are completed by shortcuts to form triangles is thus proportional to $n \times cp / n = cp$, which is a constant. This means that in the limit of large network size we can safely ignore these triangles, because they will be negligible compared to the $O(n)$ triangles in the main circle.

Triangles can also be formed from two or three shortcuts, but these also turn out to be negligible in number. Thus, to leading order in n, the number of triangles in the small-world model is simply equal to the number in the circle, which is $\frac{1}{4}nc(\frac{1}{2}c-1)$

And what about the number of connected triples? Once again, all connected triples in the circle model are still present in the small-world model. As shown in Section 15.1, there are $\frac{1}{2}nc(c-1)$ such triples. There are, however, also triples created by a shortcut combining with an edge in the circle. There are $\frac{1}{2}ncp$ shortcuts and c edges that they can form a triple with at each of their two ends, for a total of $\frac{1}{2}ncp \times c \times 2 = nc^2p$ connected triples.

There are also triples created by pairs of shortcuts. If a vertex is connected to m shortcuts then there are $\binom{m}{2}$ triples made of two shortcuts centered on that vertex and, averaging over the Poisson distribution of m, with mean cp, the expected number of connected triples centered at a vertex is

$\frac{1}{2}c^2p^2$, for a total of $\frac{1}{2}nc^2p^2$ triples over all vertices.

Thus the expected total number of connected triples of all types in the whole network is $\frac{1}{2}nc(c-1) + nc^2p + \frac{1}{2}nc^2p^2$. Substituting the numbers of triangles and triples into Eq. (15.5), we then find that

$$C = \frac{\frac{1}{4}nc(\frac{1}{2}c-1)\times 3}{\frac{1}{2}nc(c-1)+nc^2p+\frac{1}{2}nc^2p^2} = \frac{3(c-2)}{4(c-1)+8cp+4cp^2}.$$

(15.7)

Note that this becomes the same as Eq. (15.2), as it should, when $p = 0$. And as p grows it becomes smaller, with a minimum value of $C = C = \frac{3}{4}(c-2)/(4c-1)$ when $p = 1$. For instance when $c = 6$, the minimum value of the clustering coefficient is $\frac{3}{23} = 0.130\ldots$ (This behavior contrasts with that of the original Watts-Strogatz version of the small-world model in which edges are removed from the circle. In that version the clustering coefficient tends to zero as $n \to \infty$ when $p = 1$, since the network becomes a random graph at $p = 1$.)

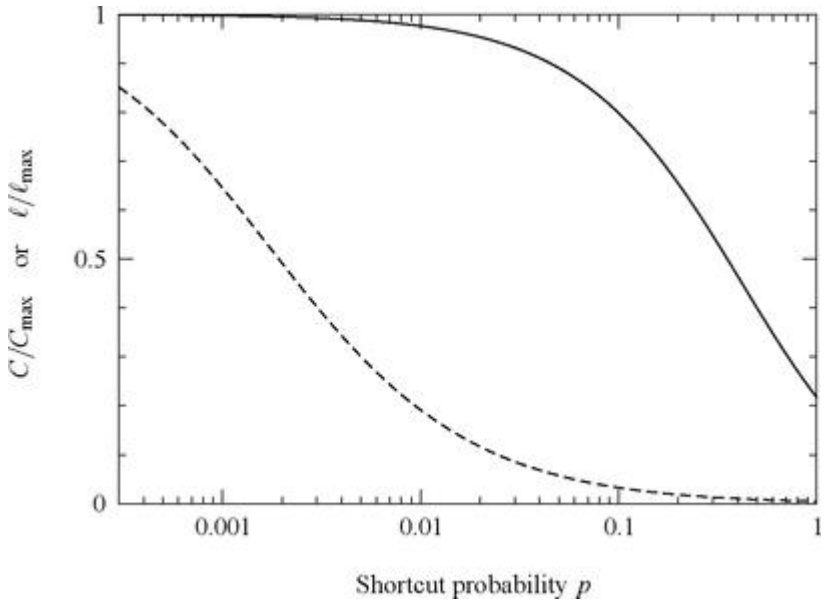


Figure 15.5: Clustering coefficient and average path length in the small-world model. The solid line shows the clustering coefficient, Eq. (15.7), for a small-world model with $c = 6$ and $n = 600$, as a fraction of its maximum value $C_{\max} = \frac{3}{4}(c-2)/(c-1) = 0.6$, plotted as a function of the parameter p. The dashed line shows the average geodesic distance between vertices for the same model as a fraction of its maximum value $\ell_{\max} = n/2c = 50$, calculated from the mean-field solution, Eq. (15.14). Note that the horizontal axis is logarithmic.

Figure 15.5 shows a plot of the clustering coefficient as a function of p for a small-world network with $c = 6$.

15.1.3 AVERAGE PATH LENGTHS

Calculating the average path length in the small-world model, i.e., the mean geodesic or shortest-path distance between pairs of vertices, is harder than calculating the degree distribution or clustering coefficient. Indeed, no exact expression for mean distance has yet been found, though some approximate expressions are known and have been found in simulations of the model to be reasonably accurate.

See Section 6.10.1 for a discussion of geodesic distances in networks.

One thing that is known about path lengths in the model is how they *scale* with the model parameters. Consider the simple case of a small-world model with $c = 2$, so that around the circle each vertex is connected only to its immediate neighbors, and consider the following dimensional argument. We define a length measure in our network by saying that the distance covered by an edge in the network is one length unit—a meter say, or a foot.[235] Then we can ask what other quantities in the model have the dimensions of length. One candidate is the distance around the whole circle, which is just n.

But there is another length in the model also, which is the mean distance between the ends of shortcuts around the circle. Suppose there are s shortcuts in our network, which means there $2s$ ends of shortcuts. (We know in fact that $s = \frac{1}{2}ncp$, but the point of this argument will be clearer if we stick with the simple notation s for the moment.) Then the average distance ξ between ends around the circle is

$$\xi = \frac{n}{2s}, \tag{15.8}$$

Once we specify the two distances n and ξ, we have specified the entire model, because once we have n the value of ξ fixes s, which fixes p, which is the only free parameter in the model given that $c = 2$.

Now consider the ratio of the length of the average shortest path in the network, which we will denote ℓ, to the length of the path around the entire circle, which is n. This ratio can, by definition, be written as a function of n and ξ, since n and ξ specify the entire model. However, it is also the ratio of two distances, meaning that it is dimensionless, and hence can be a function of only of dimensionless combinations of n and ξ. But there is only one such dimensionless combination, the ratio n / ξ. Thus it must be the case that

$$\frac{\ell}{n} = F(n/\xi) = F(2s), \tag{15.9}$$

where $f(x)$ is some function that doesn't depend on any of the parameters, a *universal function* in the language of scaling theory.

In other words, the mean geodesic distance ℓ between vertices in the small-world model with $c = 2$ is simply equal to the number of vertices n times some function of the number of shortcuts:

$$\ell = nF(2s).$$

(15.10)

And what happens for larger values of c? When we increase c the lengths of the shortest paths between vertices decrease. If we keep everything the same in our model—number of vertices, number of shortcuts—but increase c from two to four, then we will roughly halve the shortest path between any pair of vertices. This is because we now have edges connecting next-nearest neighbor vertices around the circle as well as nearest neighbors, which means that we can traverse a given distance around the circle in half as many hops as we could previously. If the path incorporates any shortcuts then that part of the distance doesn't change—the shortcuts are as long as they ever were. However, if the density of shortcuts is low then most of the hops in most paths will be around the circle rather than along shortcuts and to a good approximation we can say that the length of the paths has simply halved. Similarly, for general values of c the length of the paths is decreased by a factor of $\frac{1}{2}c$ over its value for the $c = 2$ case.

Thus, provided the density of shortcuts is low, the equation corresponding to Eq. (15.10) for general values of c is:

$$\ell = \frac{2n}{c}F(2s).$$

(15.11)

We can derive an alternative form by making use of the fact that the number of shortcuts is $s = \frac{1}{2}ncp$, which gives us $\ell = 2(n/c)f(ncp)$. In fact, conventionally we absorb the leading factor of two into the definition of f, defining a new universal function $f(x) = 2f(x)$, so that

$$\ell = \frac{n}{c}f(ncp).$$

(15.12)

This *scaling form*, first proposed by Barthélémy and Amaral [31], tells us how the average path length in the small-world model depends on the model parameters n, c, and p when the density of shortcuts is low.

The catch is that we don't know the form of the function $f(x)$. We can, however, get an idea of its shape by numerical simulation of the model. We can generate random small-world networks

and measure the mean distance ℓ between their vertices using breadth-first search (Section 10.3). Equation (15.12) tells us that if we perform such measurements for many different networks with many different values of the parameters we should find that the combination $c\ell/n$ is equal to the same function of ncp in all of them:

$$\frac{c\ell}{n} = f(ncp).$$

(15.13)

Figure 15.6 shows the results of such simulations for many different networks, and indeed we see that all of the points in the figure follow, roughly speaking, a single curve. This is the curve of $f(x)$.

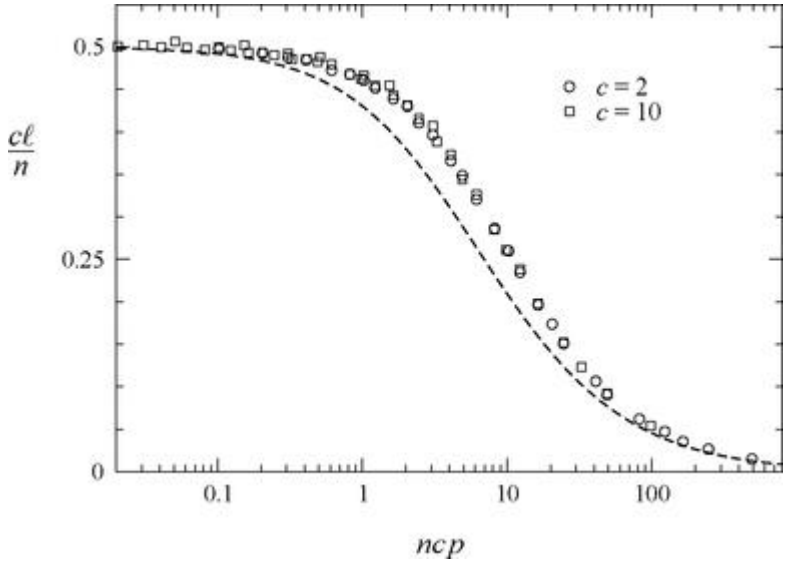


Figure 15.6: Scaling function for the small-world model. The points show numerical results for $c\ell/n$ as a function of ncp for the small-world model with a range of parameter values n = 128 to 32 768 and $p = 1 \times 10^{-6}$ to 3×10^{-2}, and two different values of c as marked. Each point is averaged over 1000 networks with the same parameter values. The points collapse, to a reasonable approximation, onto a single scaling function $f(ncp)$ in agreement with Eq. (15.13). The dashed curve is the mean-field approximation to the scaling function given in Eq. (15.14).

Another approach is to try to calculate $f(x)$ approximately in some fashion. Various approaches have been tried, including series approximations, distributional approximations, and mean-field methods. A mean-field approximation, for example, gives the result [251]

$$f(x) = \frac{2}{\sqrt{x^2+4x}} \tanh^{-1} \sqrt{\frac{x}{x+4}}.$$

(15.14)

The methods used to derive this form become exact in the limit of either very small or very large numbers of shortcuts in the network,[236] but in between around $x = 1$ they are only approximate. The form of Eq. (15.14) is shown as the dashed line in Fig. 15.6 and indeed we see that it agrees well with the numerical results at the ends of the range but less well in the middle.

This, however, is enough for us to prove that the small-world model is indeed a "small world." Consider Eq. (15.14) for large values of x. Making use of the standard identity

$$\tanh^{-1} u = \tfrac{1}{2} \ln \frac{1+u}{1-u},$$

(15.15)

we can write $f(x)$ as

$$f(x) = \frac{1}{\sqrt{x^2+4x}} \ln \frac{\sqrt{1+4/x}+1}{\sqrt{1+4/x}-1},$$

(15.16)

and then taking the limit of large x we find

$$f(x) = \frac{\ln x}{x}$$

(15.17)

for $x \gg 1$. Substituting this into Eq. (15.12), we then have

$$\ell = \frac{\ln(ncp)}{c^2 p}$$

(15.18)

for $ncp \gg 1$. Recalling that ncp is simply twice the number of shortcuts in the network, this implies that, provided the number of shortcuts in the network is significantly greater than 1, the average distance between vertices will increase logarithmically with n, i.e., very slowly, for fixed c and p. Thus the number of vertices in the network can become very large and the value of ℓ will

remain small, which is precisely the phenomenon we call the small-world effect.

Moreover, since only the *number* of shortcuts, and not the number per vertex, has to be large, the model tells us that the addition of only a small density of random shortcuts to a large network can produce small-world behavior. This helps explain why most real-world networks show the small-world effect. Most networks contain long-range connections and have at least some randomness in them—very few are perfectly regular or have only short-range connections—so we should not be surprised to see small-world behavior in almost all cases.

It is important to notice that the small-world model not only shows the small-world effect, but that it does so at the same time as displaying clustering. Since the number of shortcuts in the network is $\frac{1}{2}ncp$, we can always make it much larger than one simply by increasing the size n of the network, while keeping c and p constant. At the same time, the clustering coefficient, Eq. (15.7), is independent of n and hence retains its (non-zero) value as $n \to \infty$. In this limit, therefore, we simultaneously have non-zero clustering and the small-world effect, demonstrating conclusively that the two are not at odds with one another—it is perfectly possible to have both in the same network at the same time.

Figure 15.5 shows a plot of the approximate value of ℓ as a function of p from Eqs. (15.12) and (15.14) for a small-world model with $n = 600$ vertices and $c = 6$, along with the curve for the clustering coefficient of the same model that we plotted earlier and, as we can see, there is a substantial range of values of p in which the value of ℓ is low while the value of C is high.

Many other properties and quantities can be calculated for the small-world model, either analytically or numerically. For a short review of results concerning the model see Ref. [232].

15.2 EXPONENTIAL RANDOM GRAPHS

Many of the networks we observe in the real world exist in only one instantiation, one example that we can study. There's only one Internet, for instance, and only one World Wide Web. But is the precise structure of such a network—the precise pattern of connections in the Internet, say—the only possible structure the network could have? Common sense suggests that it is not. For a start, the Internet evolves in time, so we see different structures if we look at different times and all of them are by definition plausible structures for the network. More importantly, it's clear that, had circumstances been slightly different, the Internet could easily have evolved to have a different topology, but one that in practical terms would probably have worked about as well as the present one.

On the other hand, we can say that the structure of such an alternate Internet would probably have been "similar" to the real Internet, in some sense. That is, all reasonable choices for the structure of the Internet have some basic features in common, even if they differ in smaller details. Similar considerations also apply to other types of network, including social networks, biological networks, and information networks.

In some cases the questions we want to answer about a networked system can be tackled by studying the structure of only a single observed example—the real Internet for instance. But there are other cases where we would like to know about the entire set of possible networks that *could* represent a system. If we are studying some social process in a social network, for instance, such as opinion formation or the spread of a disease, we can measure a social network and then calculate or simulate the effects of the process of interest on that network. More often, however, we would like to know how the process behaves on social networks generally, rather than on the one particular network we have measured.

Considerations of this kind lead us to consider ensemble models of networks, an *ensemble*, in this context, meaning a set of possible networks plus a probability distribution over them. We have seen some examples of ensemble models in previous chapters, such as the random graphs of Chapters 12 and 13. In this section we introduce a beautiful and general formalism for ensemble network models called the *exponential random graph*, which includes random graphs as special cases but also extends to many other network ensembles that describe all sorts of network phenomena.[237]

Elegant though this formalism is, however, it also has some serious drawbacks. For reasons that are still not entirely understood, exponential random graphs fail as models of some common network phenomena such as transitivity (see Section 7.9). We will examine the nature of some of these failures towards the end of the chapter.

15.2.1 DEFINITION OF THE EXPONENTIAL RANDOM GRAPH

Suppose we want to create an ensemble of networks with a given set of properties, such as a given number of edges or a given value of the clustering coefficient. We can do that, as ordinary random graph models do, by fixing absolutely the values of the quantity or quantities of interest and then drawing uniformly from the set of all networks with the desired values. For instance, if we draw uniformly from the set of all graphs with a given number of edges we have the $G(n, m)$ random graph model of Section 12.1.

In many cases, however, this approach is not exactly what we want. If we observe that a social network, for example, has a given number of edges, it does not necessarily mean that every possible social network for the given community would have exactly that many edges. Had the world evolved slightly differently, the number of edges might well have turned out differently as well.

Often, therefore, a better approach is to fix the *average* value of the property or properties of interest. We might fix the average number of edges, for instance, so that some networks in our ensemble have more than the average and some have less, but over the whole ensemble we get the right average value. Moreover, we can arrange that networks with numbers of edges close to the desired value have higher probabilities in the ensemble than networks further away, so that the ensemble is dominated by networks with properties close to the desired ones. The exponential random graph provides an elegant way of achieving these goals.

Suppose, therefore, that we have some set of network measures whose numerical values we want to fix. Examples might include number of edges or mean degree of a vertex, degrees of individual vertices, number of triangles or clustering coefficient, and so forth. Let us denote these measures by $x_1, x_2, \ldots$

Recall that a simple graph is a graph with no multiedges and no self-edges—see Section 6.1.

Now consider the set $\mathscr{G}$ of all simple graphs[238] with n vertices and let us define an ensemble by giving each graph G in the set a probability $P(G)$, normalized so that

$$\sum_{G\in\mathscr{G}} P(G) = 1. \tag{15.19}$$

The mean or expectation value $\langle x_i \rangle$ of a network measure x_i within this ensemble is given by

$$\langle x_i \rangle = \sum_{G\in\mathscr{G}} P(G)\, x_i(G), \tag{15.20}$$

where $x_i(G)$ is the value of x_i measured on the graph G (e.g., number of edges in graph G, number of triangles, etc.).

Now, following the prescription outlined above, let us fix the mean value of each of our measures within our ensemble. If we do this, then Eq. (15.20) is turned around and becomes a constraint on the probability distribution over graphs:

$$\sum_{G\in\mathscr{G}} P(G)\, x_i(G) = \langle x_i \rangle, \tag{15.21}$$

where x_i is now a specified number. We have one such constraint for each network measure.

The number of measures, however, is typically quite small—maybe only one or two, maybe hundreds or even thousands, but usually nowhere near the number of graphs in our ensemble. The number of simple graphs of n vertices is $2^{n(n-1)/2}$, which becomes very large even for relatively modest values of n. This means that the constraints in Eqs. (15.19) and (15.21) do not specify the probability distribution $P(G)$ completely. Indeed, they leave an enormous amount of flexibility about the values of $P(G)$. There are many more unknowns $P(G)$ than there are constraints in our equations and hence a wide range of choices of $P(G)$ that will satisfy the constraints. How do we choose between them?

This question, of making the best choice of a probability distribution given only a relatively small number of constraints on that distribution, is one that is familiar to physicists and statisticians, having been studied for over a hundred years since the pioneering work of Willard Gibbs in the latter part of the nineteenth century. The solution is remarkably simple, although deriving it is not. It can be shown that the best choice of probability distribution is the one that maximizes the *Gibbs entropy*

$$S = -\sum_{g\in\mathscr{G}} P(G)\, \ln P(G), \tag{15.22}$$

subject to the known constraints.

One may well ask what we mean by "best choice" in this context. The maximum entropy choice is best in the sense that it makes the minimum assumptions about the distribution other than those imposed upon us by the constraints. There are choices of distribution we could make that would satisfy the constraints but would effectively make additional assumptions. For instance, some choices might make a particular graph or graphs highly probable while other graphs, only slightly different, are given far lower probabilities. These would be considered "bad" choices in the sense that they assume things about the ensemble for which we have no supporting evidence. The Gibbs entropy is precisely a measure of the amount of "assumption" that goes into a particular choice of distribution $P(G)$, or more precisely it is the amount of "antiassumption" or ignorance, and by maximizing it we minimize unjustified assumptions as much as possible. The derivation of the formula, Eq. (15.22), would take us some way away from our central topic of networks, so we will not go through it here, but the interested reader is encouraged to look for example at the books by Grandy [142] and Cover and Thomas [82].

The maximization of the entropy, subject to the constraints of Eqs. (15.19) and (15.21), can be

achieved by the method of Lagrange multipliers. The optimum is the set of values of the $P(G)$ that maximizes the quantity

$$-\sum_{G\in\mathcal{G}} P(G)\,\ln p(G) - \alpha\left[1-\sum_{G\in\mathcal{G}} P(G)\right] - \sum_i \beta_i\left[\langle x_i\rangle - \sum_{G\in\mathcal{G}} P(G)\,x_i(G)\right],$$

(15.23)

where α and β_i are Lagrange multipliers whose values will be determined shortly. Differentiating with respect to the probability $P(G)$ of a particular graph G and setting the result to zero, we then find that

$$-\ln P(G) - 1 + \alpha + \sum_i \beta_i x_i(G) = 0,$$

(15.24)

which implies

$$P(G) = \exp\left[\alpha - 1 + \sum_i \beta_i x_i(G)\right],$$

(15.25)

or

$$P(G) = \frac{e^{H(G)}}{Z},$$

(15.26)

where $Z = e^{1-\alpha}$ is called the *partition function* and

$$H(G) = \sum_i \beta_i x_i(G)$$

(15.27)

is the *graph Hamiltonian*.[239]

It remains to fix the values of Z and β_i (for all i). Z is fixed by the normalization condition, Eq. (15.19), which requires that

$$\sum_{G\in\mathscr{G}} P(G) = \frac{1}{Z}\sum_{G\in\mathscr{G}} e^{H(G)} = 1,$$

(15.28)

and hence

$$Z = \sum_{G\in\mathscr{G}} e^{H(G)}.$$

(15.29)

There is no equivalent general formula for the values of the β_i. They are calculated by substituting Eq. (15.26) into Eq. (15.21) and solving the resulting set of non-linear simultaneous equations, but the particular solution depends on the form of the Hamiltonian. We will see some examples of the process shortly.

There are some cases in which we are interested in an exponential random graph only as a class of models. That is, we are concerned not as much with the model's properties for a particular set of values $\{\beta_i\}$ as with the behavior of the model in general. In such cases we can regard the β_i as free parameters controlling the structure of the network, much as the edge probability p controls the structure of the network in a Poisson random graph.

15.2.2 EXPECTATION VALUES

Once we have determined the probability distribution $P(G)$ over graphs, we can use it to calculate estimates of quantities of interest within the ensemble. The most common objects of interest are expectation values (i.e., averages) of quantities, the expectation value of a quantity y in the ensemble being given by

$$\langle y \rangle = \sum_{G \in \mathscr{G}} P(G)\, y(G) = \frac{1}{Z} \sum_{G \in \mathscr{G}} e^{H(G)}\, y(G).$$

(15.30)

In effect, this calculation gives us a "best estimate" of the value of y. That is, given a certain set of observations or constraints on our network, embodied in Eq. (15.21), but no other information about the network structure, we can calculate a best-guess ensemble of networks subject to those constraints and then use that ensemble to calculate the expectation value of the quantity y, giving us a best guess at the value of that quantity given only the constraints. Thus the exponential random graph model enables us to answer questions of the type, "If I know certain things, A, B, and C, about a network, what is my best estimate of some other thing D?" For instance, if I know the average degree of a vertex in a network, what is my best estimate of the degree distribution? Or the clustering coefficient? The exponential random graph gives a rigorous and principled answer to questions of this kind.

An interesting special case arises when the quantity y that we want to estimate is itself one of the set of network measures x_i that we used to specify our ensemble in the first place. You might ask why we would want to do this, given that, by hypothesis, we already know the expectation values of these quantities—they are precisely the quantities that we used as inputs to our model in the first place. The answer is that we still need to fix the parameters β_i and we do this by calculating the expectation values x_i for given β_i and then varying the β_i until the x_i take the desired values.

The value of x_i within the ensemble is given by

$$\begin{aligned}\langle x_i \rangle &= \frac{1}{Z} \sum_{G \in \mathscr{G}} e^{\sum_i \beta_i x_i(G)}\, x_i(G) = \frac{1}{Z} \frac{\partial}{\partial \beta_i} \sum_{G \in \mathscr{G}} e^{\sum_i \beta_i x_i(G)} \\ &= \frac{1}{Z} \frac{\partial Z}{\partial \beta_i} = \frac{\partial \ln Z}{\partial \beta_i},\end{aligned}$$

(15.31)

where we have made use of Eq. (15.29). The quantity

$$F = \ln Z$$

(15.32)

is called the *free energy* of the ensemble and Eq. (15.31) can be written simply as

$$\langle x_i \rangle = \frac{\partial F}{\partial \beta_i}.$$

(15.33)

To calculate xi , therefore, all we need to do is calculate the partition function Z, from it evaluate the free energy, and then differentiate.

Calculating expectation values for other quantities is harder, and indeed this is one of the main practical problems with exponential random graphs: the actual calculations of quantities of interest can be very difficult and in many cases can only be performed using numerical methods. If we are clever, however, we can still use the machinery embodied in Eq. (15.33) in some cases. The trick is to introduce an extra term involving y into our Hamiltonian thus:

$$H(G) = \sum_i \beta_i x_i(G) + \mu y(G).$$

(15.34)

If we set the parameter μ to zero, then the answers we get out of our calculations will be unchanged from before and hence will still be correct. However, we can now differentiate with respect to μ (at the point $\mu = 0$) to calculate the expectation value of y:

$$\langle y \rangle = \left. \frac{\partial F}{\partial \mu} \right|_{\mu=0}.$$

(15.35)

This allows us again to calculate just the one sum, the partition function Z, and from it calculate the free energy and thus the average y . The catch is that we have to calculate Z for general (non-zero) values of μ so that we can perform the derivative—we only set μ to zero at the end of the calculation. In many cases it can be quite difficult to calculate Z in this way, which makes the exponential random graph, though elegant, technically tricky.

15.2.3 SIMPLE EXAMPLES

Probably the simplest example of an exponential random graph model is the model in which we fix the expected number of edges in an undirected network and nothing else. Following the formalism above, this gives us a graph Hamiltonian, Eq. (15.27), of $H = \beta m$, where m is the number of edges. Then individual graphs appear in the ensemble with probability

$$P(G) = \frac{e^{\beta m}}{Z},$$

(15.36)

where

$$Z = \sum_G e^{\beta m}.$$

(15.37)

Thus higher values of β in this model correspond to denser networks, those with more edges.

To make further progress with this model we need a way to perform the sum over graphs G in Eq. (15.37). The standard way to achieve this is to sum over possible values of the elements A_{ij} of the adjacency matrix. In this case we are considering undirected graphs, so we need to specify only the matrix elements above the diagonal or those below it, but not both, since the matrix is symmetric. And since we are restricting ourselves to simple graphs the only allowed values of A_{ij} are 0 and 1 if $i \neq j$ and $A_{ii} = 0$.

We can write the number of edges m in terms of the adjacency matrix thus:

$$m = \sum_{i<j} A_{ij},$$

(15.38)

and hence the partition function is

$$\begin{aligned} Z &= \sum_{\{A_{ij}\}} \exp\left(\beta \sum_{i<j} A_{ij}\right) \\ &= \sum_{\{A_{ij}\}} \prod_{i<j} e^{\beta A_{ij}} = \prod_{i<j} \sum_{A_{ij}=0,1} e^{\beta A_{ij}} = \prod_{i<j} (1 + e^{\beta}) \\ &= (1 + e^{\beta})^{\binom{n}{2}}, \end{aligned}$$

(15.39)

where the notation $\{A_{ij}\}$ indicates summation over all allowed values of the adjacency matrix. From this expression we can calculate the free energy:

$$F = \ln Z = \binom{n}{2} \ln(1 + e^{\beta}),$$

(15.40)

and thus, using Eq. (15.33), the average number of edges in the model is

$$\langle m \rangle = \frac{\partial F}{\partial \beta} = \binom{n}{2} \frac{1}{1 + e^{-\beta}}.$$

(15.41)

If we have a particular desired value that m should take, we can now achieve it by rearranging this expression to find the appropriate value for the Lagrange multiplier β thus:

$$\beta = \ln \frac{\langle m \rangle}{\binom{n}{2} - \langle m \rangle}.$$

(15.42)

We can also calculate, for example, the probability p_{vw} that there will be an edge between a particular pair of vertices v, w, which is given by the average of the corresponding element A_{vw} of the adjacency matrix. From Eq. (15.30) we have

$$p_{vw} = \langle A_{vw} \rangle = \frac{1}{Z} \sum_{\{A_{ij}\}} A_{vw} \exp\left(\beta \sum_{i<j} A_{ij}\right) = \frac{\sum_{A_{vw}=0,1} A_{vw} e^{\beta A_{vw}}}{\sum_{A_{vw}=0,1} e^{\beta A_{vw}}}$$
$$= \frac{1}{1+e^{-\beta}} = \frac{\langle m \rangle}{\binom{n}{2}}.$$

(15.43)

Thus the probability of an edge between a given pair of vertices is the same in this model for every pair. In other words, this model is just the ordinary Poisson random graph of Chapter 12 with $p = \langle m \rangle / \binom{n}{2}$. The random graph can thus be regarded as a special case of the more general exponential random graph model.

The random graph, as we saw in Chapter 12, is in many respects a poor model of real-world networks. In particular, its degree distribution is Poissonian and hence very different from the highly right-skewed degree distributions in most observed networks. It is natural to ask, therefore, whether we can make an exponential random graph model that has a more realistic degree distribution. There are a number of ways of doing this, but one of the simplest is to create a model in which we specify the expected degree of each vertex within the ensemble. That is, we create an exponential random graph model with the graph Hamiltonian

$$H = \sum_i \beta_i k_i,$$

(15.44)

where k_i is the degree of vertex i. Note that we do not also need a term that fixes the average number of edges in this model, since fixing the average degree of each vertex already fixes the average number of edges (see Eq. (6.20)).

We can write the degrees in terms of the adjacency matrix as

$$k_i = \sum_j A_{ij},$$

(15.45)

and hence write the Hamiltonian as

$$H = \sum_{ij} \beta_i A_{ij}$$
$$= \sum_{i<j} \beta_i A_{ij} + \sum_{i>j} \beta_i A_{ij} = \sum_{i<j} \beta_i A_{ij} + \sum_{i<j} \beta_j A_{ji}$$
$$= \sum_{i<j} (\beta_i + \beta_j) A_{ij},$$

(15.46)

where in the second line we have interchanged the dummy variables i and j and in the third line we have made use of $A_{ji} = A_{ij}$. We have also again assumed that there are no self-edges, so that $A_{ii} = 0$ for all i.

Now we can write the partition function as

$$Z = \sum_{\{A_{ij}\}} \exp\Bigl(\sum_{i<j}(\beta_i+\beta_j)A_{ij}\Bigr) = \prod_{i<j}\sum_{A_{ij}=0,1} \mathrm{e}^{(\beta_i+\beta_j)A_{ij}}$$
$$= \prod_{i<j}\bigl[1+\mathrm{e}^{\beta_i+\beta_j}\bigr],$$

(15.47)

and the probability of an edge between vertices u and v is

$$p_{vw} = \langle A_{vw}\rangle = \frac{1}{Z}\sum_{\{A_{ij}\}} A_{vw}\exp\Bigl(\sum_{i<j}(\beta_i+\beta_j)A_{ij}\Bigr)$$
$$= \frac{\sum_{A_{vw}=0,1} A_{vw}\,\mathrm{e}^{(\beta_v+\beta_w)A_{vw}}}{\sum_{A_{vw}=0,1}\mathrm{e}^{(\beta_v+\beta_w)A_{vw}}}$$
$$= \frac{1}{1+\mathrm{e}^{-(\beta_v+\beta_w)}}.$$

(15.48)

Thus edges in this model now have different probabilities. Of particular interest is the case of a sparse network, one in which the probability of any individual edge is small, $p_{vw} \ll 1$. (As we have seen throughout this book, most real-world networks are very sparse.) To achieve this, we need $\mathrm{e}^{-(\beta v+\beta w)} \gg 1$ in Eq. (15.48), which means that

$$p_{vw} \simeq \mathrm{e}^{\beta_v}\mathrm{e}^{\beta_w}.$$

(15.49)

In other words, in a sparse network the probability of an edge is simply a product of two terms, one for each of the vertices at either end of the edge. Moreover, it turns out that these terms are simply related to the expected degrees of the vertices. The expected degree of vertex v, for instance, is just the sum of the expected number p_{vw} of edges between it and every other vertex:

$$\langle k_v \rangle = \sum_w p_{vw} = e^{\beta_v} \sum_w e^{\beta_w},$$

(15.50)

so that

$$e^{\beta_v} = C \langle k_v \rangle,$$

(15.51)

where $C = 1/\sum_w e^{\beta w}$.

Thus $p_{vw} = C^2 \quad kv \quad kw \quad$ in this model, and since we require that $\sum_{vw} p_{vw} = \sum_v \quad kv \quad = 2 \quad m$ (see Eqs. (6.19) and (6.20)), it′s then straightforward to show that

$$p_{vw} = \frac{\langle k_v \rangle \langle k_w \rangle}{2\langle m \rangle}.$$

(15.52)

Once again, this is a model we have seen before. It is the random graph model that we studied in Section 13.2.2 in which we specify the expected degrees of vertices (rather than their exact degrees, as in the more common configuration model).

We can also create exponential random graph models of directed networks. For instance, we can make a model in which the constrained quantities are the expected values of the in- and out-degrees of a directed network by using a Hamiltonian of the form

$$H = \sum_i \beta_i^{\text{in}} k_i^{\text{in}} + \sum_j \beta_j^{\text{out}} k_j^{\text{out}}.$$

(15.53)

Writing $k_i^{\text{in}} = \sum_{j(\neq i)} A_{ij}$ and $k_j^{\text{out}} = \sum_{i(\neq j)} A_{ij}$, we have

$$H = \sum_{i \neq j} (\beta_i^{\text{in}} + \beta_j^{\text{out}}) A_{ij}.$$

(15.54)

The ensemble is now a distribution over (simple) directed graphs, which means that the adjacency matrix is in general asymmetric and each element A_{ij} can take its own value. Thus the partition function is

$$\begin{aligned} Z &= \sum_{\{A_{ij}\}} \exp\Bigl[\sum_{i\neq j}(\beta_i^{\text{in}} + \beta_j^{\text{out}})A_{ij}\Bigr] = \prod_{i\neq j}\sum_{A_{ij}=0,1} \mathrm{e}^{(\beta_i^{\text{in}}+\beta_j^{\text{out}})A_{ij}} \\ &= \prod_{i\neq j}\Bigl[1 + \mathrm{e}^{\beta_i^{\text{in}}+\beta_j^{\text{out}}}\Bigr], \end{aligned}$$

(15.55)

and the probability of an edge from w to v is

$$\begin{aligned} p_{vw} = \langle A_{vw}\rangle &= \frac{1}{Z}\sum_{\{A_{ij}\}} A_{vw}\exp\Bigl(\sum_{i\neq j}(\beta_i^{\text{in}} + \beta_j^{\text{out}})A_{ij}\Bigr) \\ &= \frac{\sum_{A_{vw}=0,1} A_{vw}\,\mathrm{e}^{(\beta_v^{\text{in}}+\beta_w^{\text{out}})A_{vw}}}{\sum_{A_{vw}=0,1}\mathrm{e}^{(\beta_v^{\text{in}}+\beta_w^{\text{out}})A_{vw}}} \\ &= \frac{1}{1+\mathrm{e}^{-(\beta_v^{\text{in}}+\beta_w^{\text{out}})}}. \end{aligned}$$

(15.56)

In the case of a sparse network this becomes

$$p_{vw} \simeq \mathrm{e}^{\beta_v^{\text{in}}}\mathrm{e}^{\beta_w^{\text{out}}} = \frac{\langle k_v^{\text{in}}\rangle\langle k_w^{\text{out}}\rangle}{\langle m\rangle},$$

(15.57)

by an argument similar to the one leading to Eq. (15.52). This expression is similar to that for the corresponding quantity in the directed version of the configuration model (see page 475), and indeed the model above is the equivalent for the directed case of the random graph in which we specify the expected degrees of the vertices rather than the exact degrees.

15.2.4 RECIPROCITY MODEL

We now turn to some more complex examples of exponential random graphs, ones that are not equivalent to models we have already seen. The first example we look at is the "reciprocity model" proposed by Holland and Leinhardt [157].

As discussed in Section 7.10, many directed networks exhibit the phenomenon of reciprocity, whereby edges between vertices tend to be reciprocated. If I say that you are my friend, for example, then it is likely that you will also say that I am your friend. We can create an exponential random graph model of reciprocity by fixing the expected number of reciprocated edges in the network. The number of reciprocated edges, m_r is given by $m_r = \sum_{i \neq j} A_{ij} A_{ji}$, so we need to introduce a term proportional to this into our graph Hamiltonian. We can also introduce other terms, such as terms to fix the expected degrees of vertices as in the previous section. Here let us look the simple case where we fix only the number of edges as we did with the Poisson random graph. The number of edges in a simple directed network is given by $m = \sum_{i \neq j} A_{ij}$ and hence our Hamiltonian takes the form

$$\begin{aligned} H &= \beta \sum_{i \neq j} A_{ij} + \gamma \sum_{i \neq j} A_{ij} A_{ji} \\ &= \sum_{i<j} \bigl[\beta (A_{ij} + A_{ji}) + 2\gamma A_{ij} A_{ji}\bigr], \end{aligned}$$

(15.58)

where β and γ are free parameters that can be varied to create the desired numbers of edges and reciprocated edges. This is actually a simplified version of the model proposed by Holland and Leinhardt, but it will serve our purpose nicely, and it is easy to solve.

The partition function for this model is

$$\begin{aligned} Z &= \sum_{\{A_{ij}\}} \exp\Biggl(\sum_{i<j} \bigl[\beta (A_{ij} + A_{ji}) + 2\gamma A_{ij} A_{ji}\bigr]\Biggr) \\ &= \prod_{i<j} \sum_{A_{ij}=0,1} \sum_{A_{ji}=0,1} \mathrm{e}^{\beta (A_{ij} + A_{ji}) + 2\gamma A_{ij} A_{ji}} = \prod_{i<j} \Bigl[1 + 2\mathrm{e}^{\beta} + \mathrm{e}^{2(\beta+\gamma)}\Bigr] \\ &= \Bigl[1 + 2\mathrm{e}^{\beta} + \mathrm{e}^{2(\beta+\gamma)}\Bigr]^{\binom{n}{2}}. \end{aligned}$$

(15.59)

The free energy for the network is then

$$F = \binom{n}{2} \ln\bigl(1 + 2\mathrm{e}^{\beta} + \mathrm{e}^{2(\beta+\gamma)}\bigr),$$

(15.60)

and, applying Eq. (15.33), we find that the expected numbers of edges and reciprocated edges are

$$\langle m \rangle = \frac{\partial F}{\partial \beta} = n(n-1)\frac{e^{\beta} + e^{2(\beta+\gamma)}}{1 + 2e^{\beta} + e^{2(\beta+\gamma)}},$$
$$\langle m_r \rangle = \frac{\partial F}{\partial \gamma} = n(n-1)\frac{e^{2(\beta+\gamma)}}{1 + 2e^{\beta} + e^{2(\beta+\gamma)}}.$$

(15.61)

In Section 7.10 we defined the reciprocity r of a directed network to be the fraction of edges that are reciprocated, which in our model is given by the ratio

$$r = \frac{\langle m_r \rangle}{\langle m \rangle} = \frac{1}{1 + e^{-(\beta+2\gamma)}}.$$

(15.62)

Thus we can control both the number of edges and the level of reciprocity in the network by suitable choices of β and γ.

15.2.5 TWO-STAR MODEL

After ordinary random graphs, probably the simplest undirected exponential random graph is the so-called *two-star model*. In this model one specifies the expected number m of edges in the network and the expected number m_2 of *two-stars*, meaning a vertex connected by edges to two others (which we called a "connected triple" in other circumstances—see Eq. (7.41) on page 200). Varying the number of two-stars allows us to control the extent to which edges in the network "stick together," meaning they share common vertices. If we fix only the number of edges in a network, then those edges may stick together or they may not, but if we also give the network a lot of two-stars, then the edges have to stick together to make the required number of two-stars. Thus the two-star model allows us to control the "clumpiness" of the network, the extent to which the edges gather together in clumps or are distributed more randomly.

A two-star is a vertex connected by edges to two other vertices.

The number of two-stars in a network is

$$m_2 = \sum_i \sum_{j(\neq i)} \sum_{k(\neq i,j)} A_{ij}A_{ik} = \tfrac{1}{2}\sum_{i\neq j} A_{ij} \sum_{k(\neq i,j)} (A_{ik} + A_{jk}),$$

(15.63)

and the number of edges is, as before, $m = \sum_{i<j} A_{ij} = \frac{1}{2}\sum_{i\neq j} A_{ij}$. Thus the Hamiltonian is

$$\begin{aligned} H &= \tfrac{1}{2}\beta \sum_{i\neq j} A_{ij} + \tfrac{1}{2}\gamma \sum_{i\neq j} A_{ij} \sum_{k(\neq i,j)} (A_{ik} + A_{jk}) \\ &= \tfrac{1}{2}\sum_{i\neq j} A_{ij}\Bigl[\beta + \gamma \sum_{k(\neq i,j)} (A_{ik} + A_{jk})\Bigr], \end{aligned}$$

(15.64)

where β and γ are our two parameters.

We encountered mean-field theory briefly earlier in the chapter, in our study of the small-world model, though we did not elaborate on it there. See Eq. (15.14) and the associated discussion.

We can solve this model using *mean-field theory*, a technique borrowed from statistical physics. We note that the term $\sum_{k(\neq i,j)} A_{ik}$ is simply the number of edges attached to vertex i, excluding any edge between i and j. All vertex pairs are equivalent in this model—vertices have no individual properties to distinguish them—so the mean probability A_{ij} of an edge between any pair is the same. If we denote this probability by p then the expected value of the term above is just

$$\left\langle \sum_{k(\neq i,j)} A_{ik} \right\rangle = \sum_{k(\neq i,j)} \langle A_{ik} \rangle = \sum_{k(\neq i,j)} p = (n-2)p.$$

(15.65)

But, assuming that the network is large, this is, to a good approximation, just np, which is the mean degree of a vertex.

The mean-field approach consists of replacing the actual term in the Hamiltonian with the expected value np. We also make the same replacement for the term $\sum_{k(\neq i,j)} A_{jk}$. These replacements are a good approximation so long as $np \gg 1$ since for large values of np the statistical variation from vertex to vertex around the expected value becomes negligible. If the value of p is kept fixed as we make our network larger then np will always be large in the limit $n \to \infty$. Thus, in the limit of large network size, this mean-field approximation is a good one.

In this large-n regime, making the replacement described above, we have

$$H = \tfrac{1}{2}(\beta + 2\gamma np) \sum_{i \neq j} A_{ij} = (\beta + 2\gamma np)m,$$

(15.66)

where m is the number of edges as before.

Now, however, this is the same as the Hamiltonian for the ordinary Poisson random graph in Section 15.2.3, except for the replacement $\beta \to \beta + 2\gamma np$, so we can immediately write down the partition function and other quantities using the results of that section. In particular, Eq. (15.41) tells us that the average number of edges in the network will be

$$\langle m \rangle = \binom{n}{2} \frac{1}{1 + e^{-(\beta + 2\gamma np)}}.$$

(15.67)

But the average number of edges is related to the mean probability of an edge by $\langle m \rangle = \binom{n}{2} p$ and hence

$$p = \frac{\langle m \rangle}{\binom{n}{2}} = \frac{1}{1 + e^{-(\beta + 2\gamma n p)}} = \tfrac{1}{2}\left[\tanh\left(\tfrac{1}{2}\beta + \gamma n p\right) + 1\right].$$

(15.68)

This gives us a self-consistent equation that we can solve to find p as a function of the parameters β and γ, and once we have p we can solve for other properties of the network by treating it as a normal Poisson random graph.

For convenience in solving for p, let us define $B = \frac{1}{2}\beta$ and $C = \frac{1}{2}\gamma n$ so that Eq. (15.68) becomes

$$p = \tfrac{1}{2}[\tanh(B + 2Cp) + 1].$$

(15.69)

There is no known closed-form solution for this equation in general, but we can visualize the solution easily enough using a graphical method. If we make plots of the lines $y = p$ and $y = \frac{1}{2}[\tanh(B + 2Cp) + 1]$ as functions of p on the same axes, they will intersect at the solution (or solutions) of Eq. (15.69). Three such plots are shown in Fig. 15.7 for different choices of the parameters.

Consider first panel (a), which shows the curve of $y = \frac{1}{2}[\tanh(B + 2Cp) + 1]$ for $C = \frac{1}{2}$ and three different values of B (solid lines). Varying B merely shifts the entire curve horizontally without changing its overall shape. For each curve there is a single point of intersection with the line $y = p$, indicated by a small circle. As B is varied this intersection point moves smoothly between high and low values of p. Thus in this regime we can tune the density of the network to any desired value by varying the parameter B (or equivalently the parameter $\beta = 2B$).

Now take a look at the last panel in Fig. 15.7, panel (c), which shows curves for $C = \frac{3}{2}$ and again three difference values of B. Again varying B shifts the curve horizontally, but now there is an important difference. Because of the higher value of C, the shape of the curve has changed. It is steeper in the middle than it was previously and as a result it is now possible at suitable values of B for the curve to intersect with the line $y = p$ not just in one place but in three different places. In this regime there are three different possible solutions for p for the same values of the parameters. In fact it turns out that the middle solution is unphysical and only the two outer solutions are realized in practice. These two, however, correspond to very different networks. One has very high density with many edges while the other is very sparse with few edges. Yet both solutions are real. If one were to simulate the two-star model on a computer, generating networks at random according to the model prescription, one would in this regime sometimes find a high-density network and sometimes a low-density one for the same parameter values, and one would not be able to predict in advance which would occur.

This peculiar behavior is called *spontaneous symmetry breaking*. It is a behavior well known to physicists, who study it in condensed matter physics, where it gives rise to the phenomenon of ferromagnetism, and in particle physics, where it gives rise to the phenomenon of particle mass. In network models, however, it is primarily an annoyance, and sometimes a grave weakness. A model that can produce two radically different classes of network for the same values of the model parameters is, at the least, troubling. But worse, for values of C as in Fig. 15.7c there some values

of p that are simply impossible to reach.

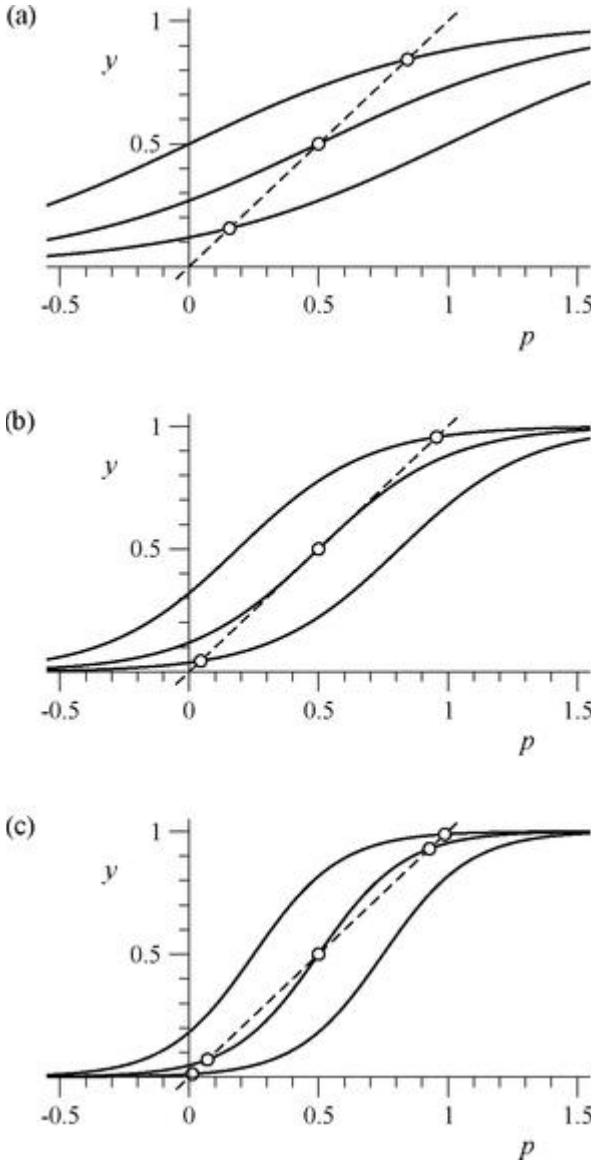


Figure 15.7: Graphical solutions of the properties of the two-star model. Curves for $y = \frac{1}{2}[\tanh(B + 2Cp) + 1]$ for varying values of B and (a) $C = \frac{1}{2}$, (b) $C = 1$, and (c) $C = \frac{3}{2}$. The points where the curves intersect the line $y = p$ (dotted line in each panel) are solutions of Eq. (15.69).

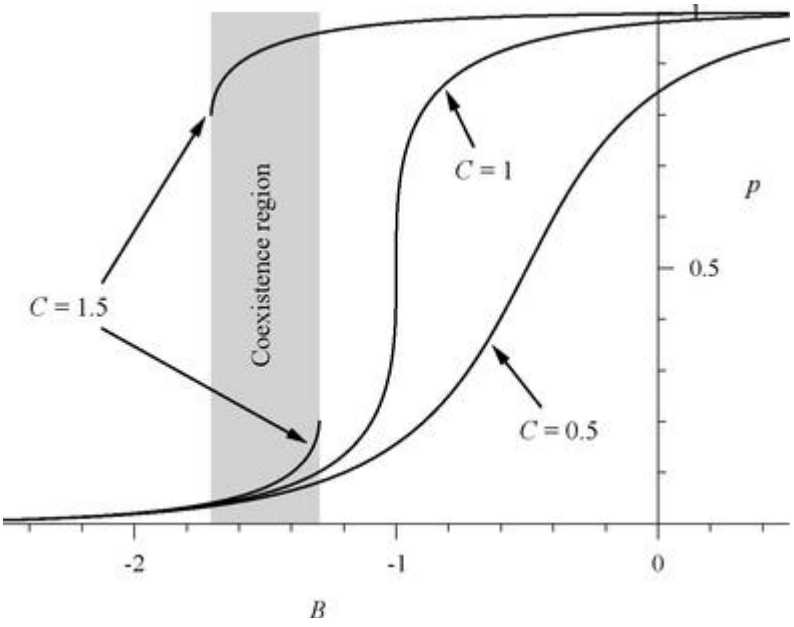


Figure 15.8: Edge probability in the two-star model. Plot of solutions of Eq. (15.69) for the edge probability p as a function of B for the same three values of C as were used in the three panels of Fig. 15.7. Note that there are two possible solutions within the coexistence region for the case $C = \frac{3}{2}$, and more importantly that for this case there is no value of B that gives any intermediate value of p. For $C = \frac{3}{2}$ the only possible values of p lie either above about 0.8 or below about 0.2.

Figure 15.8 shows the values of the solutions for p for the cases depicted in Fig. 15.7 as a function of B and, as we have said, p is a smooth function of B for the $C = \frac{1}{2}$ case, so that any value of p is reachable. For the case of $C = \frac{3}{2}$, however, there are only very high and very low values of p. There is no value of B that produces intermediate values of p and hence no way in this model to generate graphs with such intermediate values if $C = \frac{3}{2}$. If we wanted to generate a graph with $p = \frac{1}{2}$, for instance, there is simply no way to do it in the two-star model when $C = \frac{3}{2}$.

This is a fundamental problem with the two-star model and with many other exponential random graphs. We will see in the following section an example of an exponential random graph where this kind of behavior renders the model essentially useless as a model of a network.

Panel (b) of Fig. 15.7 shows the borderline case that falls between panels (a) and (c). When the parameter C is such that the curve of $y = \frac{1}{2}[\tanh(B + 2Cp) + 1]$ has gradient exactly one at its steepest point then we are right on the boundary between the two different types of behavior. In the present case, this happens at $C = 1$. If C is increased any further beyond this point, spontaneous symmetry breaking occurs. Below it, there is no symmetry breaking. In the physics jargon this transition is called a *continuous phase transition* and the point at which it occurs is called a *critical point*.[241]

Note that, even when the value of C is greater than 1 and we are above the critical point, spontaneous symmetry breaking still only occurs within a certain range of values of B, as Fig. 15.7c shows. If B is either too small or too large then there is only one solution to Eq. (15.69) (the two outer curves in Fig. 15.7c). The portion of parameter space where there are two solutions is called the *coexistence region*. The boundaries of the coexistence region correspond to the values of B such that the curve is tangent to the line $y = p$, as shown in Fig. 15.9. Put another way, we are on the boundary when the point at which the curve has gradient one falls on the line $y = p$. The

gradient of $y = \frac{1}{2}[\tanh(B + 2Cp) + 1]$ is given by

$$\frac{dy}{dp} = C\,\mathrm{sech}^2(B + 2Cp),$$

(15.70)

and setting this equal to one and making use of $\mathrm{sech}^2 x = 1 - \tanh^2 x$, we have

$$1 - \tanh^2(B + 2Cp) = \frac{1}{C}.$$

(15.71)

But p is also a solution of Eq. (15.69), so $\tanh(B + 2Cp) = 2p - 1$ and Eq. (15.71) becomes $1 - (2p - 1)^2 = 1/C$, or

$$p^2 - p + \frac{1}{4C} = 0,$$

(15.72)

which has solutions

$$p = \tfrac{1}{2}\left[1 \pm \sqrt{1 - 1/C}\right].$$

(15.73)

Rearranging Eq. (15.69) for B and substituting for p we then find that

$$\begin{aligned} B &= \tanh^{-1}(2p - 1) - 2Cp \\ &= \pm\tanh^{-1}\sqrt{1 - 1/C} - C\left[1 \pm \sqrt{1 - 1/C}\right], \end{aligned}$$

(15.74)

where we either take both the plus signs or both the minus signs.

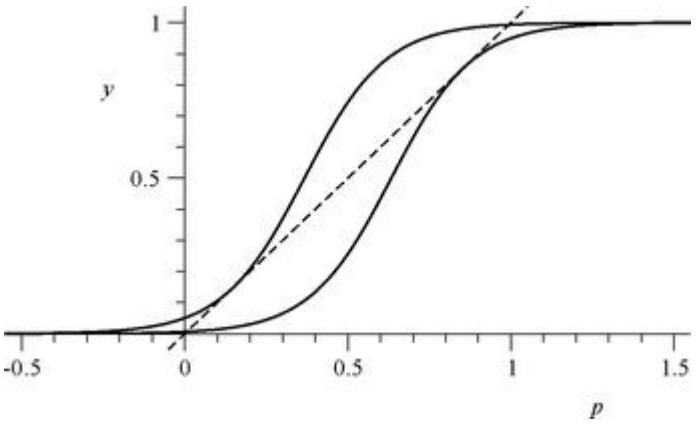


Figure 15.9: The boundaries of the coexistence region in the two-star model. The ends of the coexistence region for a given value of C correspond to those values of B that place the curve $y = \frac{1}{2}[\tanh(B + 2CP) + 1]$ precisely tangent to the line $y = p$.

Figure 15.10 shows a plot of this result in the form of a *phase diagram* of the two-star model showing the different regimes or "phases" of the model as a function of its two parameters. The two lines corresponding to the solutions in Eq. (15.74) form the boundaries of the coexistence region. Inside this region there are values of p than cannot be reached for any choice of parameters. Outside it, we can generate networks with any value of p.

15.2.6 STRAUSS′ S MODEL OF TRANSITIVE NETWORKS

As our last example in this chapter, we look at another exponential random graph model that shows spontaneous symmetry breaking, the transitive network model of Strauss [306]. Where the two-star model is something of a toy model—useful for demonstrating the mathematics, but not especially important in practice—the model of this section is one of some importance, and the fact that it shows pathological behavior with the variation of its parameters is a puzzle and a significant hindrance to progress, one that has not, at least at the time of writing, been fully resolved.

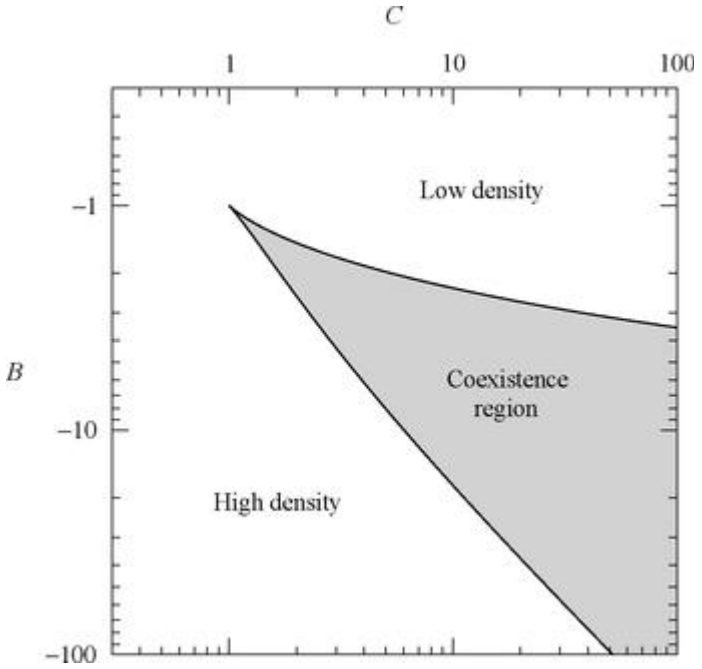


Figure 15.10: Phase diagram of the two-star model. The phases of the two-star model as a function of the parameters B and C. Density generally increases as B becomes more negative and for $C > 1$ there is a coexistence region at intermediate values of B in which spontaneous symmetry breaking occurs. Notice that the scales are logarithmic and that $B < 0$. (There are no other phases for positive B or negative C, so these values are not shown.) Adapted from Park and Newman [260]. Original figure Copyright 2009 American Physical Society. Reproduced with permission.

Strauss's model is a model of a simple undirected network that shows clustering or transitivity, the propensity for triangles to form in the network, which, as discussed in Section 7.9, is a common phenomenon, particularly in social networks. In this model one specifies the expected number of edges m in the network and also the expected number of triangles m_3. The number of triangles can be expressed in terms of the elements of the adjacency matrix as

$$m_3 = \tfrac{1}{3}\sum_{ijk} A_{ij}A_{jk}A_{ki},$$

(15.75)

where the factor of $\frac{1}{3}$ accounts for the fact that each triangle in the network appears three times in the sum. The number of edges is just $m = \frac{1}{2}\sum_{ij} A_{ij}$. (For simplicity of notation we have included the diagonal terms in these sums. They are zero since the network is simple, so it makes no difference whether we include them or not.) Thus the graph Hamiltonian is

$$H = \tfrac{1}{2}\beta \sum_{ij} A_{ij} + \tfrac{1}{3}\gamma \sum_{ijk} A_{ij}A_{jk}A_{ki}.$$

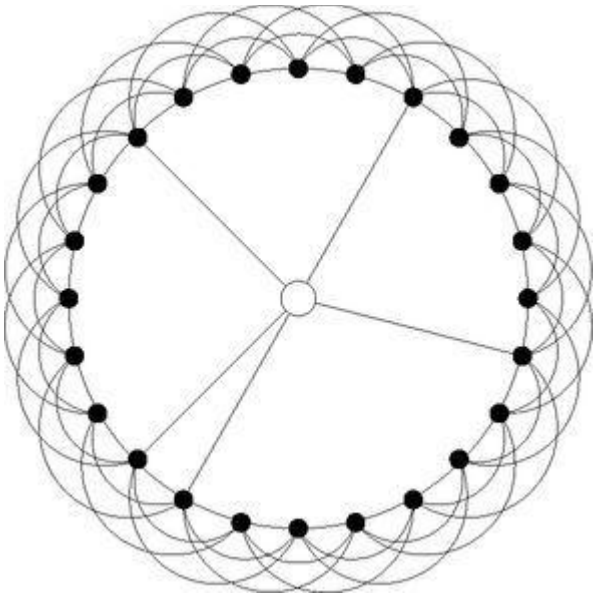
(15.76)

This model, like the two-star model, can be solved exactly in the limit of large network size using a mean-field technique. The details of the calculation are more complicated than for the two-star model. As well as replacing sums of the form $\sum_k A_{ik}$ by their average value, we also make a similar replacement for sums of the form $\sum_k A_{jk}A_{ki}$, and the values of these two quantities are expressed self-consistently in terms of each other. We will not go into the details of the calculation here—the interested reader is invited to consult Ref. [261]. The end result, however, is similar to that for the two-star model: there is a phase transition in the model beyond which the system develops a coexistence region where there are two distinct solutions to the equations, both of which are realized in simulations of the model. One solution corresponds to a network of high density and the other a network of low density but, as in the two-star case, there is in this regime no choice of model parameters that will give networks of medium density and as a result there is a wide range of networks that simply cannot be generated by this model. If one were to observe a network in the real world whose properties fell within this unattainable range, then Strauss's model could not be used to mimic its properties.

This is a fundamental problem with Strauss's model and with many similar exponential random graphs. The entire point of a model such as this one is to create model networks with properties similar to those seen in real networks. Moreover, this model in particular and exponential random graphs in general seem at first sight to be a very logical approach to the creation of such networks: from a statistical point of view the construction of the model using a maximum entropy ensemble is natural and should, one might imagine, give sensible answers. The fact that it does not is a disturbing finding that is still not properly understood. That there are ranges of network properties that simply cannot be created using the model, while at the same time real-world networks can and do display properties in these ranges, indicates that there is a fundamental flaw or gap in our reasoning, or perhaps in our understand of the nature of networks themselves. Strauss himself was already aware of these issues when he proposed his model in the 1980s, and the fact that they are still unresolved indicates that there are some difficult issues here.

PROBLEMS

15.1 Consider the following variation on the small-world model. Again we have a ring of n vertices in which each is connected to its c nearest neighbors, where c is even. And again a shortcut is added to the network with probability p for each edge around the ring, but now instead of connecting random vertex pairs, each shortcut connects a random vertex to the same single hub vertex in the center of the network:



This model could be, for example, a model of a (one-dimensional) world connected together by a bus or train (the central vertex) whose stops are represented by the shortcuts.

Show that the mean distance between two vertices in this network in the limit of large n is $\ell = 2(c^2p + 1)/c^2p$ (which is a constant, independent of n).

15.2 One of the difficulties with the original small-world model depicted in Fig. 15.3a is that vertices can become disconnected from the rest of the network by the rewiring process. For instance, a single vertex can become disconnected if all of its incident edges around the ring are rewired and it has no shortcuts.

a. Show that the probability of this happening to any given vertex is $[\, pe^{-p} \,]^c$.
b. Hence, how large must the network be before we expect that one vertex will be disconnected, if $c = 6$ and $p = 0.01$?

15.3 Consider an undirected exponential random graph model in which the Hamiltonian takes the form $H = \sum_{i<j} \Theta_{ij} A_{ij}$, where the Θ_{ij} are parameters we control.

a. Derive an expression for the free energy.