How Small is your Social Life?
Exactly Solvable Small-World Networks

Douglas Ashton

MPhys Project Report
Department of Physics
University of Oxford

Supervisor: Prof. Neil Johnson

Submitted May, 2004

Abstract

This report looks at simple networks that are designed to model the many so called "small-world" networks found in the real world. We review what small-world networks are, and some of the models designed to mimic them. A particular model, originally proposed by Dorogovtsev and Mendes, that looks at systems where the shortcuts through the network are created by a central hub is extended creating several new results. First we look at what happens when a second hub is introduced and then many identical hubs. We find the effect of having an associated cost for paths through the central hub and when that cost depends on how connected the hub is we are able to find the minimum possible path and how it scales with network size.

Contents

1 Introduction

1.1 It's a Small World After All

On a day to day basis the world seems like a very big place. There are roughly 6.4 billion people and most of us only know about 300 [1][2]. Allegedly someone wins the lottery every week, we even watch them on the television grinning away saying something like "I never thought I'd win" but somehow that doesn't make you believe it will ever be you. However, we are regularly faced with events that make the world seem somewhat smaller: You might have a friend that went to school with the prime minister, or perhaps last weeks lottery winner turns out to know your mum's milk man.

In 1967 an experiment was performed by psychologist Stanley Milgram [3] where letters were sent to a random group of people in Nebraska and Kansas in the United States with the instructions that the letter was to be forwarded to a stockbroker who lived in Boston. No address was given. They had to do this by sending the letter to someone they were personally acquainted with and who they considered might be socially "closer" to a Boston stockbroker.

The letters mostly found their target. This might seem impressive on its own, but what was more impressive was that typically they made it in about six steps. This result became famous and spawned the phrase "six degrees of separation". The population of the United States is a small fraction of the world population, but the effect seems to be similar if we'd started in Nebraska or Italy. Whether it is six or ten, the idea that any two people on the planet can be linked through a small chain of acquaintances does rather shake ones' opinion of how we all fit together.

Social networks are not the only place where we see this kind of effect. Similar "small-world" behaviour has been seen in biological systems, the internet, the world wide web, power grids and even Hollywood actors. These and many others are discussed in detail in the review articles by Newman[4], and Dorogovtsev and Mendes[5]. If we can understand the structure of the complex networks that lie all around us then we might be able to use this to our advantage. We can attempt to answer questions such as how do diseases spread? How vulnerable to an attack is the internet?

There have been many attempts to create mathematical models that exhibit small-world behaviour. In 1998 Watts and Strogatz [6] presented a very simple model that was able to produce the main characteristic properties of a small-world network. Apart from the path length between nodes on the network being small, small-world networks are also highly clustered. A large proportion of the people you know, know each other. Watts and Strogatz's (W-S) model has clustering built in at the start. The second property of a small-world network, the short path length, comes about through a small number of random connections across the network.

Despite its simplicity, in the W-S model an explicit form for the average path length across the network could not be found. In 2000 Dorogovtsev and Mendes [8] introduced a model that had a central hub and all shortcuts across the network went through that. At first glance it is not quite as realistic as the W-S model but it is able to exactly solve for the average path length.

1.2 Outline of the Report

What comes to mind immediately when looking at the Dorogovtsev-Mendes (D-M) model is what would happen if there were two central hubs? Indeed Dorogovtsev and Mendes asked this themselves although it still remains to be answered.

Another interesting question that has arisen recently is what might work against the small-world effect? The W-S model shows that just a few random links across the network is enough to bring the average path length down to a small size. In many real-world networks the problem this creates is that those very same links that helped to create the small-world effect get congested. If the congestion is strong enough it might destroy the small-world effect completely. The D-M model looks particularly vulnerable to this kind of congestion. If the central hub is not able to keep up with demand then its usefulness might seize up all together.

We will start by reviewing the models discussed above. In particular we will go through the model setup by Dorogovtsev and Mendes in some detail as we will then be presenting several new extensions to this model. These extensions focus on two main lines of interest:

1. Instead of a single hub what happens when we allow two hubs? We answer this for two different hubs and also for N identical hubs.

2. With so much traffic going through the central hub what happens if there is a cost imposed? We will show that when the cost depends on how connected the hub is there is an optimum way to wire up the network to create the shortest paths through. We also show that the network no longer behaves like a small-world network at this point.

All the results after section (2) are completely original.

2 Background

2.1 Some definition of terms

The problem of complex networks has been approached from many different fields and so there are many different ways of referring to the same thing. In this section we will present a selection of the most important definitions.

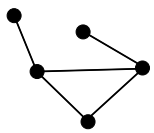


Figure 1: Example of clustering. This network has $C = \frac{3\times1}{7} = \frac{3}{7}$

In mathematical literature a network is often referred to as a 'graph'. Graphs are made up of 'vertices' (often referred to as 'sites' or 'nodes') that are connected together by 'edges' (often referred to as 'connections' or 'links'). For example, in the world wide web the vertices would be the webpages and the edges would be the hyperlinks. In this report we will usually use the terms 'node' and 'connection', although we may use the mathematical terms occasionally.

There are also some general properties of networks that it is useful to define now:

1. **Degree**: The degree of a vertex is defined as the number of edges that are attached to it. If the edges are directed then the vertex can have a separate 'in' degree and 'out' degree which are the number of incoming and outgoing edges.

2. **Clustering (or Transitivity)**: If two people are friends then there is a good chance that a friend of the first person is also a friend of the second. On a graph this would create a closed triangle with three edges connecting three vertices. Some quantifiable measure of how clustered a network is would be useful and so the clustering of a network is defined as:

$$C = \frac{3 \times \text{number of triangles in the network}}{\text{number of connected triples}}$$

   where a "connected triple" is a single vertex with edges running to two other vertices (see figure 1). There are also definitions for local clustering at a vertex[4] which we will not go through.

3. **Geodesic Path**: The shortest path between two vertices. In this report the geodesic distance is the length ℓ. We will also be interested in the mean geodesic distance.

These properties are useful because they allow us to compare our models to real world systems and see if we are on the right track. Data from many different real world systems has been compiled over recent years so there is a lot to work with.

2.2 Poisson Random Graphs

The random graph model consists of n vertices where each of the $\frac{1}{2}n(n-1)$ pairs are connected by an edge with probability p. The mean degree of any vertex is given by $z = p(n-1)$, in the limit of large n the mean number of neighbours a distance d away from the vertex is z^d. So for this to include the entire network we require $z^\ell \simeq n$. This gives us one of the main features of a small world network - that is[1]:

$$\ell \propto \log n$$

As far as the other main feature - namely clustering - goes, the random graph falls down. In the high n limit there is virtually no clustering at all. This was addressed by Watts and Strogatz.

[1] For rigourous proof see reference [11]

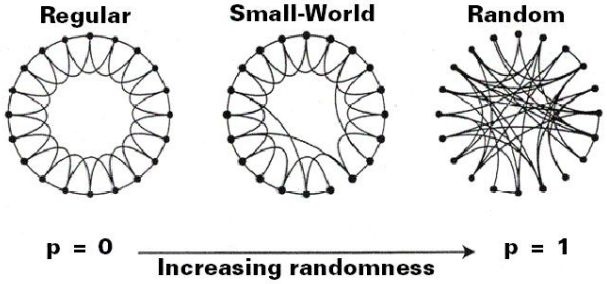


Figure 2: Strogatz and Watts' small world model. Each connection is rewired from the ordered state with a probability p. When only a few links are rewired we still have high clustering but the average shortest path has decreased rapidly.

2.3 Watts-Strogatz Model

Any model that is designed to reflect the small-world networks that we see around us must include clustering. Almost every network in the real world studied has a clustering coefficient of order a tenth or a hundredth[2]. To build this in, Watts and Strogatz created the following model [6]: Arrange n nodes around a circle and connect each node with its k nearest neighbours. This will create nk connections. With a certain probability p, each connection is then rewired to another node on the network chosen uniformly at random. The network this creates is illustrated in figure (2).

For $p \to 0$, we have a totally ordered network and the mean path length across it is $\bar{\ell} \to n/4k$. In the other limit, $p \to 1$, we get a totally random network and obtain the same result as in section (2.2) that $\bar{\ell} \to \log n / \log k$. The clustering coefficient (which comes from all the nodes being linked to k nearest neighbours) remains high for $p \to 0$ but quickly diminishes as $p \to 1$.

On the face of it we haven't solved the problem at all. We either have high clustering or a short path length, but never both. However, Watts and Strogatz were able to show that there exists a large region where the clustering coefficient remains high and the average shortest path has dropped dramatically. This can be understood when the effect of one rewiring is considered: Creating a long distance connection (or shortcut) has a highly nonlinear effect on the average shortest path. The removal of that link at most has a linear effect on the clustering coefficient. For this reason after p has been turned up from zero only a small amount we see small world behaviour emerging.

The exact form for the average shortest path in the small-world region cannot be calculated in this model. We do have the limits, and numerical analysis shows that $\bar{\ell}$ scales logarithmically with n.

2.4 The Dorogovtsev-Mendes Model

A model that allows us to exactly solve for the average shortest path, $\bar{\ell}$, is one proposed by Mendes and Dorogovtsev [8]. The bulk of this report will be looking at extensions of this model so it is worth going through it in some detail. Some variables that were defined in previous sections such as k and z are redefined here and should not be confused. The model goes as follows: Let n nodes be arranged around a circle. Each node is connected to its

[2] see table II on p10 of ref [4]

nearest neighbours by a link of unit length. A central hub is added, and with a probability p, any node can be attached to it by a link of length $\frac{1}{2}$. We then consider two cases. The first case is as shown in figure (3a) where the links around the circle are directed. The second case is shown in figure (3b) and the links are undirected. In both cases links to the hub are undirected.

We are looking for the probability, $P(\ell)$, that the shortest path between two randomly selected nodes is of length ℓ. To get this we first find the probability, $P(\ell, k)$, that the shortest path is ℓ, *given* they are separated around the ring by length k: $\sum_{\ell=1}^{k} P(\ell, k) = 1$. We can then sum over all possible values of k to get the full distribution.

2.4.1 Directed Case

First we will consider $P(\ell, k)$ for the directed model of figure (3a), the reader can verify the following expressions for small ℓ,k.

$$\begin{aligned} P(1,1) &= 1 \\ \\ P(1,2) &= p^2 \\ P(2,2) &= 1-p^2 \\ \\ P(1,3) &= p^2 \\ P(2,3) &= 2p^2(1-p) \\ P(3,3) &= 1-p^2-2p^2(1-p) \end{aligned} \tag{2.1}$$

The general form is:

$$P(\ell < k, k) = \ell p^2 (1-p)^{\ell-1} \tag{2.2}$$

$$P(\ell = k, k) = 1 - p^2 \sum_{i=0}^{k-1} i(1-p)^{i-1} \tag{2.3}$$

Performing the summation over i we get:

$$P(\ell = k, k) = (1 + (\ell - 1)p)(1-p)^{\ell-1} \tag{2.4}$$

We now know the probability that two nodes separated around the circle by length k will have a shortest distance ℓ between them. To get the probability distribution for any pair regardless of k, we sum over all possible values of k. Each distribution $P(\ell, k)$ is normalised so we get for $P(\ell)$:

$$P(\ell) = \frac{1}{n-1} \sum_{k=1}^{n-1} P(\ell, k) = \frac{1}{n-1} \sum_{k=\ell}^{n-1} P(\ell, k) \tag{2.5}$$

Inserting equations (2.2) and (2.4) we get finally

$$P(\ell) = \frac{1}{n-1} \big[1 + (\ell - 1)p + \ell(n - 1 - \ell)p^2\big](1-p)^{\ell-1} \tag{2.6}$$

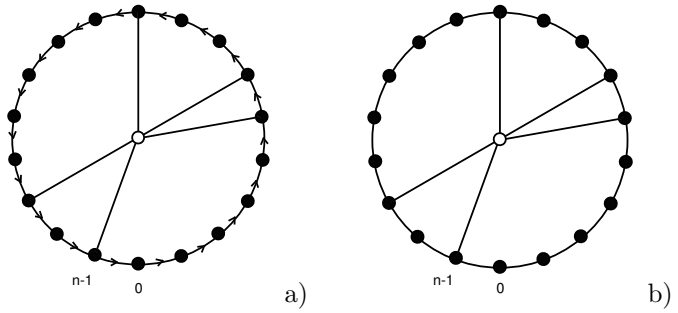


Figure 3: The Dorogovtsev-Mendes models. Links to the hub are length $\frac{1}{2}$ and undirected in both cases. In a) links around the circle are directed and in b) they are undirected.

If $p \to 0$, $P(\ell) \to 1/(n-1)$. If $p \to 1$ then $P(\ell) \to \delta_{\ell,1}$, where $\delta_{i,k}$ is the Kronecker symbol. This is what we would expect. The average value for the shortest path across the network is given by:

$$\bar{\ell} = \sum_{\ell=1}^{n-1} \ell P(\ell) \tag{2.7}$$

Substituting equation (2.6) above we get

$$\bar{\ell} = \frac{1}{n-1}\left[\frac{2-p}{p}n - \frac{3}{p^2} + \frac{2}{p} + \frac{(1-p)^n}{p}\left(n-2+\frac{3}{p}\right)\right] \tag{2.8}$$

As $p \to 0$, $\bar{\ell} \to n/2$ and $\bar{\ell}(p \to 1) \to 1$.

To obtain a scaled description of the crossover region we introduce the variables $\rho \equiv pn$ and $z \equiv \ell/n$. In the limit $n \to \infty$ and $p \to 0$ these variables remain fixed. In this limit we can use equation (2.6) to obtain:

$$nP(\ell) \equiv Q(z,\rho) = [1 + \rho z + \rho^2 z(1-z)]e^{-\rho z} \tag{2.9}$$

where $(1-\rho/n)^{nz} = e^{nz\ln(1-\rho/n)} \approx e^{-\rho z}$. This function is plotted for several different values for ρ in figure (4). We can also find the limiting form for $\bar{z}$ from equation (2.8)

$$\bar{z} = \frac{1}{\rho^2}\left[2\rho - 3 + (\rho+3)e^{-\rho}\right] \tag{2.10}$$

The last result is interesting because it allows us to see the cross over between the limiting values of $\bar{z}$ at $\rho = 0$ and $\rho = n$. The system undergoes a sharp transition after only a small number of connections are added (see figure (5)).

2.4.2 Undirected case

The directed links of the previous section could be seen to be a little artificial. So Dorogovtsev and Mendes [8] also considered the model in figure (3b) with undirected links. The expressions are more complicated now because there are more possible paths with the same length. We can straight away say that:

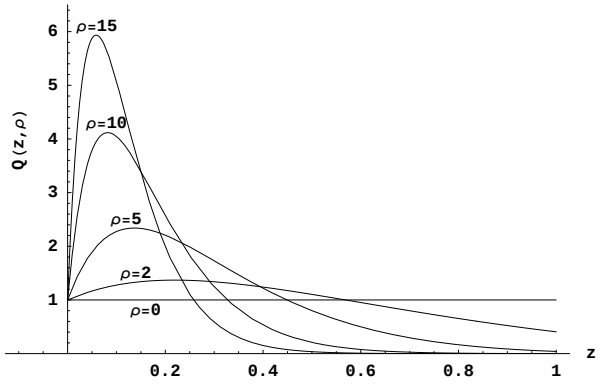


Figure 4: The scaled distribution $Q(z,\rho) \equiv nP(\ell,p)$ for the model with directed links, where $z \equiv \ell/n$ and $\rho \equiv pn$. Curves for $\rho = 0, 2, 5, 10, 15$ are drawn.

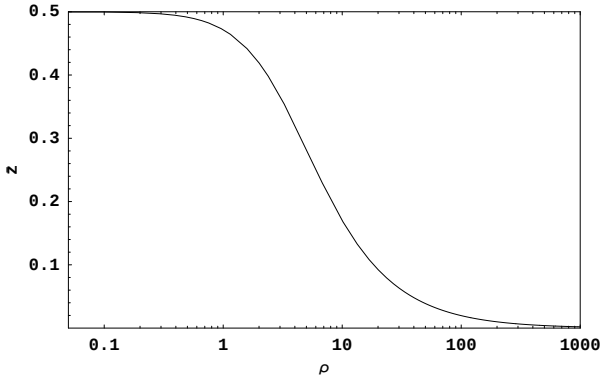


Figure 5: $\bar{z}$ ($\equiv \bar{\ell}/n$) vs ρ ($\equiv pn$) for the model with directed links.

$$\begin{aligned} P(1,1) &= 1 \\ P(\ell=1,k) &= p^2 \end{aligned}$$

For $\ell > 1$ consider figure (6) that shows two sections of the circle around our randomly selected pair. One can see that for every possible path there are $2\ell - 4$ nodes that have to be disconnected from the hub. After this there are two distinct paths to follow. Firstly, if the target or the starting hub is directly connected to the hub then there is only one path that has probability $(1-p)^{2\ell-4}p(p+p-p^2)$ (fig 6b). In the case that neither are directly connected then we get $(1-p)^{2\ell-4}(p+p-p^2)^2$, and there are 2ℓ of these (fig 6a). Putting together we get for the undirected case:

$$P(\ell=1,k) = p^2 \tag{2.11}$$

$$P(2 \le \ell < k, k) = p^2(1-p)^{2\ell-4}(2-p)(2\ell-2-\ell p) \tag{2.12}$$

$$P(\ell=k,k) = (1-p)^{2k-4}\big[1+(2k-4)p-(k-1)p^2\big] = 1 - \sum_{\ell=1}^{k-1} P(\ell,k) \tag{2.13}$$

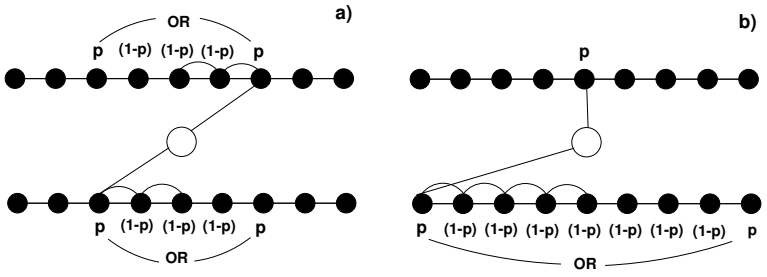


Figure 6: The two distinct possible routes between two nodes where the shortest path, ℓ, is less than their physical separation k using undirected links. Summing over all possible paths allows us to derive equations (2.12) and (2.13).

To get the full distribution we sum over all possible values of k once more to get for odd n:

$$P(\ell) = \frac{2}{n-1} \sum_{k=1}^{(n-1)/2} P(\ell, k) \tag{2.14}$$

Using equations (2.11), (2.12) and (2.13) we obtain

$$P(\ell = 1) = \frac{2}{n-1}\left(1 + \frac{n-3}{2}p^2\right) \tag{2.15}$$

$$P(\ell \geq 2) = \big[2 + 4(\ell-2)p + 2(\ell-1)(2n-4\ell-3)p^2 - 2(2\ell-1)(n-2\ell-1)p^3 + \ell(n-2\ell-1)p^4\big]\frac{(1-p)^{2\ell-4}}{n-1} \tag{2.16}$$

This last result is a lot more complicated than for the directed case. Interestingly when you take the scaling limit ($n \to \infty, p \to 0$) we find that $Q_{undir}(z, \rho) = 2Q_{dir}(2z, \rho)$. The models are only different by a scaling factor of two: $z \to 2z$, with z now running from 0 to 1/2.

2.4.3 Summary of D-M model

The advantage of this model is that we are able to find the exact form of $P(\ell)$ and thus $\bar{\ell}$. We see that it exhibits the same behaviour as the W-S model in that only a few added links is enough to reduce the shortest path greatly. It is not quite the same as the W-S model because where as here $\bar{\ell} \sim 1/\rho$, in the W-S model we get $\bar{\ell} \sim \log\rho/\rho$. As Dorogovtsev and Mendes point out, the model does show an often occurring situation in the real world whereby far connections usually occur through some common centre.

3 Adding a Second Hub

A possible extension of the Dorogovtsev-Mendes (D-M) model is to consider the effect of adding another central hub with a probability q of attachment (see figure 7). This is attractive because systems with more than one centre are perhaps more realistic, or at least more common. It also brings with it a much more complicated problem as there are now many more ways to traverse the network. One can categorise them in the following three ways:

1. Move around the circle, take a shortcut through one hub and then arrive at the destination after another movement around the circle.
2. Going into one hub, moving along a short bridge section on the circle, going back into the second hub and then on to the destination.
3. Use neither hub and just go straight around the circle.

Clearly the number of possible paths has grown considerably, the 'bridge' section that was mentioned is the main difficulty and causes any analytical solution in the vein of section (2.4) to be very difficult. A slightly simpler model might be to say that there is an infinite cost associated with using both hubs, it is easy to imagine a system where one has to choose between central paths. These are the networks that we consider in the following sections.

3.1 Two Hubs with Directed Links

Considering figure 7a now. Recalling some results from the single hub case. For two nodes separated by k we have:

$$P_s(\ell < k, p) = \ell p^2 (1-p)^{\ell-1} \tag{3.1}$$

$$P_s(\ell = k, p) = 1 - \sum_{i=1}^{k-1} P(\ell < k, p) \tag{3.2}$$

where P_s indicates the distribution from the single hub D-M model. k is the separation between the nodes around the circle and ℓ is the geodesic path length between them. For two hubs the probability that the shortest path will be ℓ (given $\ell < k$) will be:

$$P(\ell < k) = P_s(p)\Big[1 - \sum_{i=1}^{\ell-1} P_s(q)\Big] + P_s(q)\Big[(1 - \sum_{i=1}^{\ell-1} P_s(p)\Big] - P_s(p)P_s(q) \tag{3.3}$$

where $P_s(p)$ is understood to be $P_s(\ell < k, p)$ from equation (3.1). The first term is the probability that a geodesic path runs through the p-hub and *not* the q-hub. The last term is the probability that there are geodesic paths through the p-hub and the q-hub.

Considering the summation in the first term of (3.3) we can write this explicitly as:

$$\begin{aligned} 1 - \sum_{i=1}^{\ell-1} P_s(q) &= 1 - q^2 \sum_{i=1}^{\ell-1} i(1-q)^{i-1} \\ &= (1 - q + \ell q)(1-q)^{\ell-1} \end{aligned} \tag{3.4}$$

Inserting 3.4 into 3.3 we get an expression which can be written:

$$P(\ell < k) = (g_1 \ell + g_2 \ell^2)((1-p)(1-q))^{\ell-1} \tag{3.5}$$

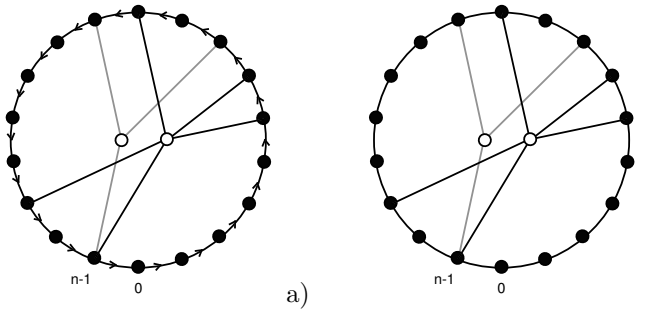


Figure 7: The two networks under consideration. Links to the hub are length $\frac{1}{2}$ and undirected in both cases. In a) links around the circle are directed and in b) they are undirected.

where

$$g_1 = p^2(1-q) + q^2(1-p) \tag{3.6}$$
$$g_2 = pq(p+q-pq) \tag{3.7}$$

The full distribution is now taken by summing over all possible k. Note that $P(\ell > k, k) = 0$:

$$P(\ell) = \frac{1}{n-1}\sum_{k=1}^{n-1} P(\ell,k) = \frac{1}{n-1}\sum_{k=\ell}^{n-1} P(\ell,k) \tag{3.8}$$
$$= \frac{1}{n-1}\left[1 - \sum_{i=1}^{\ell-1} P(i<\ell) + (n-1-\ell)P(\ell<k)\right] \tag{3.9}$$

This kind of summation appears a lot and so we replace each term with the set of functions

$$f_m(a,n) = \sum_{i=1}^{n-1} i^m a^{i-1}$$

These will be used a lot in this report and are discussed in detail in appendix A. We can now put everything into equation (3.9) to get the final result:

$$P(\ell) = \frac{1}{n-1}\left[1 - g_1 f_1(a,\ell) - g_2 f_2(a,\ell) + (n-1-\ell)(g_1\ell + g_2\ell^2)a^{\ell-1}\right] \tag{3.10}$$

where $a = (1-p)(1-q)$ and g_1, g_2 are given in equations 3.6 and 3.7.

$P(\ell)$ has been plotted in figure 8 using the scaled variables $\rho_p \equiv np$, $Q(z,\rho_p,\rho_q) \equiv nP(\ell,p,q$ and $z \equiv \ell/n$.

3.2 Two Hubs with Undirected Links

Now we consider the case where the links are undirected and can go in either direction. Recall the result from the single hub case in section (2.4.2) which for a given separation k gives:

$$P_s(\ell = 1, k) = p^2 \tag{3.11}$$
$$P_s(2 \le \ell < k, k) = p^2(1-p)^{2\ell-4}(2-p)(2\ell - 2 - \ell p) \tag{3.12}$$

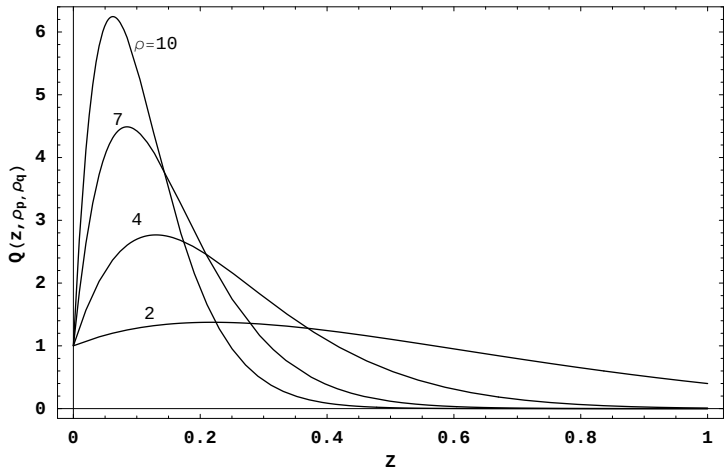


Figure 8: Plotting the scaled functions $Q(z, \rho_p, \rho_q) \equiv nP(\ell, p, q)$ with $p = q$. The curves from top to bottom are $\rho_p = \rho_q = 10,7,4,2$ respectively.

Using the same argument as for the directed case we can write down:

$$P(2 \leq \ell < k) = P_s(p)\Big[1 - \sum_{i=1}^{\ell-1} P(q)\Big] + P_s(q)\Big[(1 - \sum_{i=1}^{\ell-1} P(p)\Big] - P_s(p)P_s(q) \tag{3.13}$$

Considering the summation in the first term. To make it smaller on the page we'll use the substitution:

$$\begin{aligned} a_q &= (1-q)^2 \\ 1 - a_q &= q(2-q) \end{aligned}$$

Thus giving:

$$\begin{aligned} \sum_{i=1}^{\ell-1} P(q) &= q^2 + q^2(2-q)\sum_{i=2}^{\ell-1}(i(2-q)-2)(1-q)^{2i-4} \\ &= q^2 + \frac{(1-a_q)^2}{a_q}\sum_{i=2}^{\ell-1} i a_q^{i-1} - \frac{2q(1-a_q)}{a_q}\sum_{i=2}^{\ell-1} a_q^{i-1} \end{aligned} \tag{3.14}$$

The summations are the $f'_m(a, n)$ functions and are listed in the appendix. Substituting in the summations and rearranging gives:

$$\sum_{i=1}^{\ell-1} P(q) = 1 - \Big(q(2-q)\ell + (1-q)^2 - 2q\Big)a_q^{\ell-2} \tag{3.15}$$

When we substitute into equation 3.13 we get:

$$\begin{aligned} P(2 \leq \ell < k) &= (g_{0pq} + g_{1pq}\ell + g_{2pq}\ell^2)(a_p a_q)^{\ell-2} \\ &\quad +(g_{0qp} + g_{1qp}\ell + g_{2qp}\ell^2)(a_p a_q)^{\ell-2} \\ &\quad -(h_0 + h_1\ell + h_2\ell^2)(a_p a_q)^{\ell-2} \end{aligned}$$

Where

$$\begin{aligned} g_{0pq} &= -2p^2(2-p)((1-q)^2-2q) \\ g_{1pq} &= p^2(2-p)\big((2-p)((1-q)^2-2q)-2q(2-q)\big) \\ g_{2pq} &= p^2(2-p)^2q(2-q) \\ h_0 &= 4p^2q^2(2-p)(2-q) \\ h_1 &= -2p^2q^2(2-p)(2-q)(4-(p+q)) \\ h_2 &= p^2q^2(2-p)^2(2-q)^2 \end{aligned} \tag{3.16}$$

Gathering together all the terms gives finally:

$$P(2 \le \ell < k) = (g'_0 + g'_1\ell + g'_2\ell^2)(a_p a_q)^{\ell-2} \tag{3.17}$$

Where

$$g'_i = g_{ipq} + g_{iqp} - h_i \tag{3.18}$$

We now need to sum over all possible values of k to get the final distribution $P(\ell)$. Taking an odd numbered network:

$$P(\ell \ge 2) = \frac{2}{n-1}\sum_{k=1}^{\frac{n-1}{2}} P(\ell,k) = \frac{2}{n-1}\sum_{k=\ell}^{\frac{n-1}{2}} P(\ell,k) \tag{3.19}$$

$$P(\ell \ge 2) = \frac{2}{n-1}\left[1 - \sum_{i=1}^{\ell-1} P(i<\ell) + (\frac{n-1}{2} - \ell)P(\ell < k)\right] \tag{3.20}$$

Finally the probability that two randomly chosen nodes will be separated by ℓ is given by:

$$\begin{aligned} P(\ell=1) &= \frac{2}{n-1}\big(1 + \frac{n-3}{2}(p^2+q^2-p^2q^2)\big) \\ P(\ell \ge 2) &= \frac{2}{n-1}\Bigg[1 - (p^2+q^2-p^2q^2) \\ &\quad -\frac{1}{a_p a_q}\Big(g'_0 f'_0(a_p a_q,\ell) + g'_1 f'_1(a_p a_q,\ell) + g'_2 f'_2(a_p a_q,\ell)\Big) \\ &\quad +(\frac{n-1}{2} - \ell)(g'_0 + g'_1\ell + g'_2\ell^2)(a_p a_q)^{\ell-2}\Bigg] \end{aligned} \tag{3.21}$$

In the scaling limit of $n \to \infty; p, q \to 0$, we once again get the result that $Q_{undir}(z, \rho_p, \rho_q) = 2Q_{dir}(2z, \rho_p, \rho_q)$. So adding another hub, and constraining to using only one, doesn't change the scaling relationship between the two models.

These results were tested using a computer simulated version of the model. The method used is described in detail in appendix B. Geodesic paths between random pairs were found on 10^5 different networks of size $n = 1000$, each sampling 10^3 random pairs giving a total of 10^8 samples. The statistical error was thus very small and no deviation could be seen from the results derived.

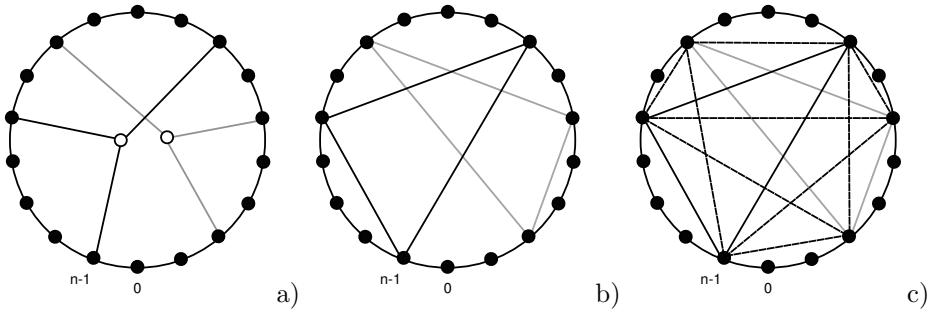


Figure 9: a) The two hub network under consideration. b) The equivalent network with the connections drawn direct. c) The hubs are connected and all the new cross connections added.

3.3 Comparing Two Hubs with One Big Hub

One question that the reader may be asking is how similar is having two hubs with probabilities of p and q to having one well connected hub with probability of $p+q$? In the cases we've been looking at where only one hub can be used at a time the answer is not at all. This is best understood by looking at figure (9). Because the links to the hub are defined as being $\frac{1}{2}$, then it is equivalent to think of all the nodes that are attached to the hub having unit links straight across the circle to every other attached node. The number of connections across is on average then $\frac{(np)(np-1)}{2}$, which is just the number of pairs. Considering the case of two hubs, with the cost of using both hubs being infinite, we get the number of pairs out to be:

$$= \frac{(np)(np-1)+(nq)(nq-1)}{2} \tag{3.22}$$

if we do the same for a single hub with probability $p+q$ and subtract equation (3.22) we get the difference to be

$$\Delta_{connections} = n^2pq = \rho_p\rho_q \tag{3.23}$$

This is usually a significant difference to the total number of connections and so the two networks are very different.

When we wave the cost of using both hubs a very different effect happens. The bridges act to join the two hubs together so that in some sense we have created a single hub network. The difference is that the bridge will come at some cost and there's no way of knowing exactly what that will be as each network will have a different smallest bridge. Hence we get distributions that are similar, but slightly shifted in favour of higher path length, from the the single hub case. Examples of how they compare are shown in figure (10).

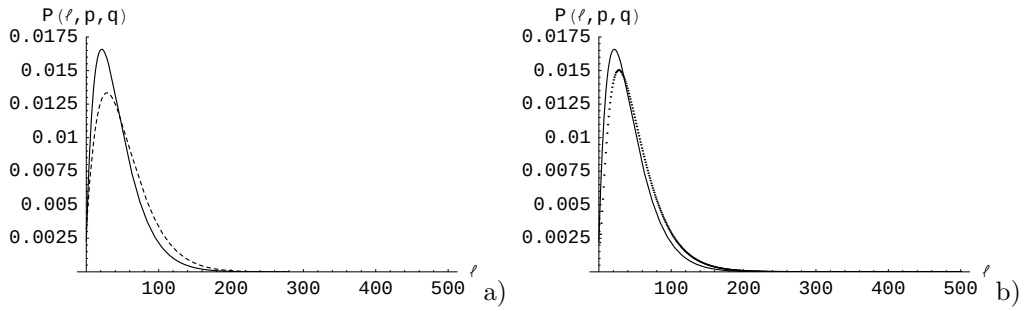


Figure 10: The solid line is that of a single hub network with $n = 1000$ and $p = 0.02$ The other lines are both with two hubs $p = q = 0.01$, a) derived result using no bridge sections, b) data taken from a computer simulation which allowed both hubs to be used. These networks use undirected links, one can see that they look the same as the directed networks except ℓ is scaled so it only goes up to $n/2$.

4 Cost for Travelling Through the Hub

In the networks we have been studying up to now almost all traffic has been flowing through one or two central hubs. Although in many systems this a reasonable picture it is easy to imagine networks where the central hubs are only able to deal with a certain volume of traffic at any one time, and this may cause congestion. For example, if the central hub was a router transporting packets of information around a computer network and the demand was high, then packets would have to wait their turn before they are sent on to their destination. In section (3.3) we saw that allowing traffic to go through both hubs was a little bit like having one hub with an associated cost of travelling through it (the bridge section). Although this is not an exact picture of what is happening it is another motivation to investigate the effect of adding a cost to the hub as a possible approximation to the two hub case.

We will now find the probability distributions for networks where any path that goes through the central hub incurs a penalty of length c for directed and undirected links.

4.1 Single Hub with a Cost

4.1.1 Directed Case

This derivation is a very simple extension of the original single hub model from section (2.4.1). There will be a discrete change in the distribution at the point where the length $\ell = c+1$, until this point there is no point in using the hub. The probability that the shortest path is ℓ, is just the probability that the two nodes are separated by length ℓ around the circle:

$$P(\ell, \ell \leq c) = \frac{1}{n-1} \tag{4.1}$$

So now we consider $\ell > c$ for a fixed k. We can use equations (2.2) and (2.3) adjusting

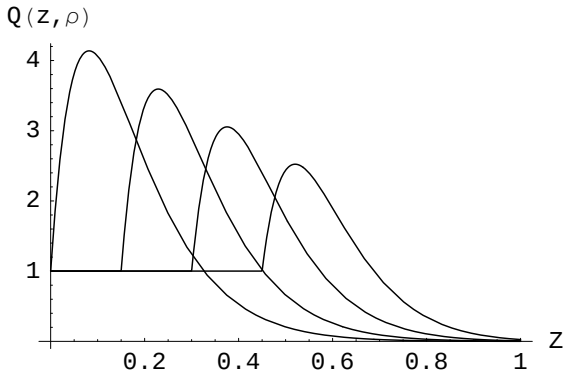


Figure 11: The effect of adding a cost to the single hub case. This is for a fixed value of $\rho = 10$ and a scaled cost of $\gamma = c/n = 0, 0.15, 0.3, 0.45$. The highest peak is for $\gamma = 0$ and then they drop with rising γ.

for the extra cost to get:

$$\begin{aligned} P(\ell < k) &= (\ell - c)p^2(1-p)^{\ell-c-1} && (4.2) \\ P(\ell = k) &= 1 - \sum_{i=c+1}^{\ell-1} P(i < k) \\ &= 1 - p^2 \sum_{i-c=1}^{(\ell-c)-1} (i-c)(1-p)^{(i-c)-1} \\ &= (1 + (\ell - c - 1)p)(1-p)^{\ell-c} && (4.3) \end{aligned}$$

The distribution is still given by summing over all values of k:

$$P(\ell, \ell > c) = \frac{1}{n-1} \sum_{k=\ell}^{n-1} P(\ell, k)$$

Subbing in (4.2) and (4.3):

$$\begin{aligned} P(\ell, \ell \leq c) &= \frac{1}{n-1} && (4.4) \\ P(\ell, \ell > c) &= \frac{1}{n-1}\left[1 + (\ell - c - 1)p + (n - 1 - \ell)(\ell - c)p^2\right](1-p)^{\ell-c-1} && (4.5) \end{aligned}$$

We take the scaling limit to be $p \to 0$ as $n \to \infty$ with the scaling variables $z \equiv \ell/n$, $\gamma \equiv c/n$, $\rho \equiv np$. Some scaled versions of $Q(\rho, z, \gamma) \equiv nP(p, \ell, c)$ are plotted in figure (11).

4.1.2 Undirected Case

Once again this is a fairly simple extension of the original model. Unfortunately it's not quite as simple as replacing ℓ with $\ell - c$, because only the $c < \ell < k$ part of the distribution is shifted, firstly we can simply say that because the hub is useless for $\ell \leq c$:

$$P(\ell, \ell \leq c) = \frac{2}{n-1} \tag{4.6}$$

For $\ell > c$ and a fixed k, we can modify equations (2.11), (2.12) and (2.13) to get:

$$\begin{aligned} P(\ell - c = 1, k) &= p^2 && (4.7) \\ P(2 \leq \ell < k, k) &= p^2(1-p)^{2(\ell-c)-4}(2-p)((\ell-c)(2-p)-2) && (4.8) \\ P(\ell = k, k) &= 1 - p^2 - \sum_{i-c=2}^{\ell-c-1} P(2 \leq \ell < k, k) && (4.9) \end{aligned}$$

Now we need to explicitly calculate $P(\ell = k, k)$, once again we'll be using the variable:

$$\begin{aligned} a &= (1-p)^2 \\ 1-a &= p(2-p) \end{aligned}$$

$$\begin{aligned} P(\ell = k, k) &= 1 - p^2 - p^2(2-p) \sum_{i-c=2}^{\ell-c-1} ((\ell-c)(2-p)-2)a^{\ell-c-2} \\ &= 1 - p^2 - \frac{(1-a)^2}{a} \sum_{i-c=2}^{\ell-c-1} (i-c)a^{i-c-1} - 2\frac{p(1-a)}{a} \sum_{i-c=2}^{\ell-c-1} a^{i-c-1} \end{aligned}$$

These summations are the $f'_m(a, n)$ functions given in the appendix. Substituting in and rearranging gives:

$$P(\ell = k, k) = \left[1 - 4p + p^2 + p(2-p)(\ell - c)\right] a^{\ell-c-2} \tag{4.10}$$

The distribution is now obtained by summing over all possible values of k. It will only be non-zero for $k > \ell$ so we can write down:

$$P(\ell, \ell > c) = \frac{2}{n-1} \sum_{k=\ell}^{\frac{n-1}{2}} P(\ell, k) \tag{4.11}$$

which we can fill in from what we've already calculated as:

$$\begin{aligned} P(\ell, \ell > c) &= \frac{1}{n-1} \Big[2 - 8p + 2p^2 + 2p(2-p)(\ell - c) \\ &\quad + (n - 2\ell - 1)p^2(2-p)((\ell-c)(2-p)-2) \Big] (1-p)^{2(\ell-c)-4} \end{aligned} \tag{4.12}$$

In the scaling limit $n \to \infty$ and $p, q \to 0$ we find that the undirected case is related to the directed case by $nP(\ell, c) \equiv Q_{undir}(z, \gamma) = 2Q_{dir}(2z, 2\gamma)$ where γ is the scaled cost and is defined as $\gamma \equiv c/n$.

We've now derived the adjusted distribution for a hub which has a penalty c for travelling through it. On its own this is not a major extension of the D-M model, however in section (5) we will show how adding a cost $c(p, n)$ can introduce some interesting effects. For completeness we shall now derive the distribution for two hubs with two costs.

4.2 Two Hubs with Costs

For all the subsequent models in this section I'm going to assume that there is a cost associated with each hub of value c_p, c_q where $c_p \geq c_q$ without losing any generality. The cost for using both hubs is assumed to be infinite.

4.2.1 Directed Case

By now the reader will have noticed a pattern as to how we are approaching these problems. The basic steps are:

1. Find the probability the shortest path is ℓ through one particular hub for a fixed separation k.
2. $P(\ell < k) = P(\ell < k, p) + P(\ell < k, q) - P_s(\ell < k, p)P_s(\ell < k, q)$ where the last term is able to use the single hub cases because the probability that the shortest path is ℓ through both hubs is just that. $P(\ell = k) = 1 - \sum_{\ell}^{k-1} P(l < k)$
3. Total distribution is then obtained by summing $P(\ell, k)$ over all k and normalising.

Consider $\ell \geq c_p \geq c_q$. First we want the probability the shortest path is ℓ through the p hub.

$$P(\ell < k, p) = P_s(\ell, p)\left[1 - \sum_{i-c_q=1}^{\ell-c_q-1} P_s(i, q)\right] \tag{4.13}$$

Inserting equation 4.2 and performing the sum gives us:

$$\begin{aligned} P(\ell < k, p) &= (1-p)^{-c_p}(1-q)^{-c_q}p^2\big[(\ell - c_p) + (\ell - c_p)(\ell - c_q - 1)q\big](a_p a_q)^{\ell-1} \\ &= (g_{0pq} + g_{1pq}\ell + g_{2pq}\ell^2)(a_p a_q)^{\ell-1} \end{aligned} \tag{4.14}$$

where

$$\begin{aligned} a_p &= 1 - p \\ g_{0pq} &= (1-p)^{-c_p}(1-q)^{-c_q}p^2 c_p((c_q+1)q - 1) && (4.15) \\ g_{1pq} &= (1-p)^{-c_p}(1-q)^{-c_q}p^2(1 - (c_p + c_q + 1)q) && (4.16) \\ g_{2pq} &= (1-p)^{-c_p}(1-q)^{-c_q}p^2 q && (4.17) \end{aligned}$$

we also need:

$$\begin{aligned} P_s(\ell < k, p)P_s(\ell < k, q) &= \frac{p^2q^2}{(1-p)^{-c_p}(1-q)^{-c_q}}(\ell^2 - \ell(c_p + c_q) + c_p c_q)(a_p a_q)^{\ell-1} && (4.18) \\ &= (h_0 + h_1\ell + h_2\ell^2)(a_p a_q)^{\ell-1} && (4.19) \end{aligned}$$

where

$$\begin{aligned} h_0 &= (1-p)^{-c_p}(1-q)^{-c_q}p^2q^2c_p c_q && (4.20) \\ h_1 &= -(1-p)^{-c_p}(1-q)^{-c_q}p^2q^2(c_p + c_q) && (4.21) \\ h_2 &= (1-p)^{-c_p}(1-q)^{-c_q}p^2q^2 && (4.22) \end{aligned}$$

we can now write down the full probability:

$$P(\ell < k) = (g'_0 + g'_1\ell + g'_2\ell^2)(a_p a_q)^{\ell-1} \tag{4.23}$$

where

$$\begin{aligned} g'_i &= g_{ipq} + g_{iqp} - h_i && (4.24) \\ & && (4.25) \end{aligned}$$

This has been very similar to the two hub case up until now. In fact if we set $c_p = c_q = 0$ we retrieve the polynomials that were derived in section 4.1.1. We now require $P(\ell = k, \ell > c_p)$, to calculate this correctly we have to be very careful about limits:

$$P(\ell = k, \ell > c_p) \quad = \quad 1 - \sum_{i=c_q+1}^{\ell-1} P(i < \ell) \tag{4.26}$$

From $i = c_q + 1 \to c_p$, $P(i < \ell)$ is given by equation 4.2 and from $i = c_p + 1 \to \ell - 1$ it is given by equation (4.23). We thus get:

$$P(\ell = k, \ell > c_p) = 1 - \sum_{i=c_q+1}^{c_p} P_s(i < c_p, q) - \sum_{i=c_p+1}^{\ell-1} P(i < k) \tag{4.27}$$

It is convenient here to introduce the new summation functions

$$\tilde{f}_m(a, n_1, n_2) = f_m(a, n_1) - f_m(a, n_2) \tag{4.28}$$

Using this definition and substituting equations 4.2 and 4.23 into 4.27 we get:

$$\begin{aligned} P(\ell = k, \ell > c_p) = 1 - \frac{q^2}{(1-q)^{c_q}} \big[\tilde{f}_1(a_q, c_p + 1, c_q + 1) - c_q \tilde{f}_0(a_q, c_p + 1, c_q + 1)\big] \\ - \big[g'_0 \tilde{f}_0(a_p a_q, \ell, c_p + 1) + g'_1 \tilde{f}_1(a_q a_p, \ell, c_p + 1) + g'_2 \tilde{f}_2(a_p a_q, \ell, c_p + 1)\big] \end{aligned} \tag{4.29}$$

Now we have everything. The final distribution is obtained by summing over all possibly values of k and is:

$$P(\ell, \ell \leq c_p) \quad = \quad \frac{2}{n-1} \tag{4.30}$$

$$P(\ell, c_q < \ell \leq c_p) \quad = \quad \frac{1}{n-1}\big[1 + (\ell - c_q - 1)q + (n - 1 - \ell)(\ell - c_q)q^2\big](1-q)^{\ell - c_q - 1} \tag{4.31}$$

$$\begin{aligned} P(\ell, c_p < \ell) \quad = \quad & \frac{1}{n-1}\Bigg[1 - \frac{q^2}{(1-q)^{c_q}}\big[\tilde{f}_1(a_q, c_p + 1, c_q + 1) - c_q \tilde{f}_0(a_q, c_p + 1, c_q + 1)\big] \\ & - \big[g'_0 \tilde{f}_0(a_p a_q, \ell, c_p + 1) + g'_1 \tilde{f}_1(a_q a_p, \ell, c_p + 1) + g'_2 \tilde{f}_2(a_p a_q, \ell, c_p + 1)\big] \\ & + (n - 1 - \ell)(g'_0 + g'_1 \ell + g'_2 \ell^2)(a_p a_q)^{\ell - 1}\Bigg] \end{aligned} \tag{4.32}$$

An example of this distribution is plotted in figure 12. Interestingly if the value of ρ_q increases above ρ_p the distribution tends to the single hub case extremely quickly - the p hub is barely used. If the p hub has a high degree and a high cost then the distribution behaves as though it's not there until $\ell > c_p$ where it quickly falls to zero. The costs have to be a reasonable fraction of the total network size to make a visible impact on the distribution.

All the derivations so far have shown that in the limit of high n the networks with directed links behave in the same way as with undirected links apart from a scaling of variable $z \to 2z$ and $\gamma \to 2\gamma$. The undirected version of the two hubs with costs network was derived and is no different so the result has been excluded.

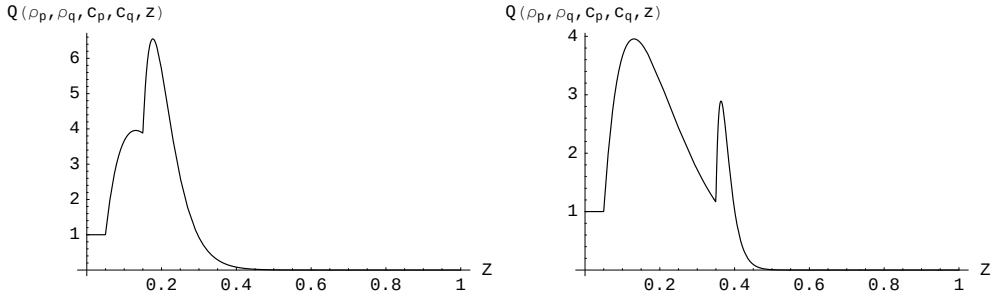


Figure 12: Examples of the distribution when the two hubs have associated costs for travelling through them. a) Here $\rho_p = 20$, $\rho_q = 10$ and $c_p = 0.15$, $c_q = 0.05$. b) Here $\rho_p = 50$, $\rho_q = 10$ and $c_p = 0.35$, $c_q = 0.05$

5 Congestion Charging

5.1 Charging Per Connection

So far we've worked out distributions for networks that have an associated cost for using the hub. This cost was labelled c and assumed to be a constant. Is it reasonable to consider a constant cost? It would seem more reasonable that the cost might be some function of how connected the hub is. It is easy imagine how this kind of cost may arise. If the hub was perhaps only able to deal with one trip through it at a time then the cost would grow linearly with the number of connections. On average, for a single hub network, the number of connections to the hub is $\rho \equiv np$. On the other hand it might be more important how many pairs are connected directly across the network, in this case cost would grow as ρ^2.

If the cost is growing with the number of connections then it should be the case that there will be some optimal value for the number of connections that creates the shortest average path. To study this we recall the probability distribution for a single hub with a cost and directed links from section (4.1.1).

$$P(\ell, \ell \leq c) = \frac{1}{n-1} \tag{5.1}$$

$$P(\ell, \ell > c) = \frac{1}{n-1}\big[1 + (\ell - c - 1)p + (n - 1 - \ell)(\ell - c)p^2\big](1-p)^{\ell - c - 1} \tag{5.2}$$

To get the average path length across the network, $\bar{\ell}$, we need to sum over this distribution:

$$\bar{\ell} = \sum_{\ell=1}^{n-1} \ell P(\ell) \tag{5.3}$$

The sum happens in two parts. It requires summing up to n and taking away the sum up to c so we require the functions $\tilde{f_m}(a, n_2, n_1)$ once again. We then get:

$$\bar{\ell} = \frac{(1-p)^{-c}}{n-1}\left[(1 - p - cp)\tilde{f}_1 + p\big(1 + p(n - 1 + c)\big)\tilde{f}_2 - p^2\tilde{f}_3\right] + \frac{c(c-1)}{2(n-1)} \tag{5.4}$$

where it is assumed that $\tilde{f_m} = \tilde{f_m}((1-p), n, c)$. Expanding out gives:

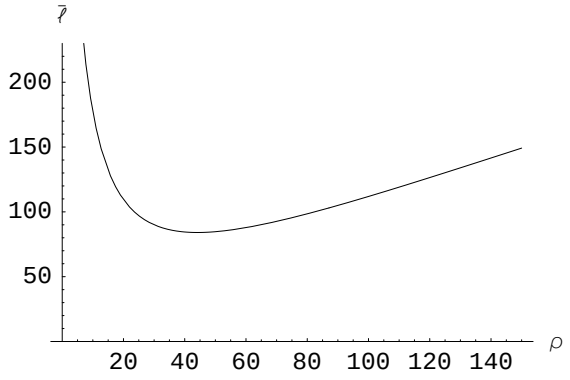


Figure 13: with a cost per connection, $k = 1$, and $n = 1000$ this figure shows that there exists an optimal value for the number of connections that minimises $\bar{\ell}$

$$\bar{\ell} = \frac{(1-p)^{n-c}\big(3 + p(n-2-c)\big) + p\big(2 - 2c + 2n - (c-1)(c-n)p\big) - 3}{p^2(n-1)} + \frac{c(c-1)}{2(n-1)} \quad (5.5)$$

This function can be plotted and is shown in figure (13) with a cost of 1 per connection (i.e $c = np = \rho$), it clearly shows that there is an optimal amount of connections - in this case about 44. This number will vary for different values for n. If the network is very large it might be worth it to have a higher cost as in general the nodes are much further from one another.

5.2 Minimizing $\bar{\ell}$

What we really want is an equation that tells us how many connections we should make for a given cost and network size. To get that we should differentiate equation (5.5) and solve for p. This is a rather tricky differential equation. It can however be solved numerically and we can thus plot how many connections we should use for a given price of connection, or even see what happens as we vary the network size at a fixed price (figure 14). In the following figures we are using:

$$c = k\rho = knp$$

To gain some insight into the workings of this we can also make some approximations. In the limit of large n, or more importantly $n - c$, the exponential term in equation (5.5), $(1-p)^{n-c} \to e^{-\rho}$. Provided the cost per connection is not too high the region containing the minimum will be at a reasonably high ρ. We saw that for $k = 1$ the minimum occurs at 44 which is well above what we need. So carefully leaving out the exponential term we get:

$$\bar{\ell} \approx \frac{p\big(2 - 2c + 2n - (c-1)(c-n)p\big) - 3}{p^2(n-1)} + \frac{c(c-1)}{2(n-1)} \quad (5.6)$$

This can be differentiated to find the minimum value of $\bar{\ell}$. Inserting the cost as $c = knp$ we get:

$$\frac{12 - 4(1+n)p - 2p^4k^2n^2 + p^3(kn + 2n^2k)}{p^3} = 0 \quad (5.7)$$

We can write this out as a fourth order polynomial in p and n, or if we like we can switch to the ρ variable which directly corresponds to the average number of connections to the hub.

$$12 - 4(1+n)p + p^3(kn + 2n^2k) - 2p^4k^2n^2 = 0 \quad (5.8)$$

$$12n^2 - 4n(1+n)\rho + \rho^3k(1+2n) - 2\rho^4k^2 = 0 \quad (5.9)$$

We can find the limit by assuming that when n is large so is ρ. We don't yet know how it will scale with n but it seems very reasonable that it will increase like n^x where $0 < x \leq 1$. It is difficult to guess how ρ will scale just by looking at equation (5.9), however one can make an educated guess by looking at the numerical solutions. These suggest that choosing $x \approx 1/2$ would be a good starting point, using this to select the higher order terms we obtain:

$$\rho \approx \sqrt{\frac{2n}{k}} \quad (5.10)$$

For a sufficiently large network there is a simple relationship between how many connections you should put in to create the shortest average path around the network. Figure (14) was created from numerically solving equation (5.5) for minimum $\bar{\ell}$, it shows that these approximations are good in the high n limit. It seems to hold surprisingly well even down to around $n \sim 10$ although it is particularly good starting around 10^3.

The other possible way of 'pricing' the hub would be to look at how many pairs there are connected via the hub. Roughly speaking this increases as ρ^2 so putting $c = k(np)^2$ into equation (5.6), differentiating and setting to zero we get an analogous result to equation (5.8):

$$6 - 2p(1+n) + n^2p^3(p + 2np - 2)k - 2n^4p^6k^2 = 0 \quad (5.11)$$

Once again there is some choice for guessing the form of how ρ scales with n, but numerical solutions suggest it is around $1/3$, so keeping the higher order terms with this in mind we obtain:

$$\rho \approx \sqrt[3]{\frac{n}{k}} \quad (5.12)$$

this agrees well with numerical solutions. One may be skeptical how we chose which terms were highest order and which we could throw away. The numerical solutions only give an indication as to what we should keep. The results from equations (5.10) and (5.12) can be taken and compared more closely with the full numerical solutions of equation (5.5) and one sees that they are indeed the correct scaling form. We can now put the form for the minimising value of ρ back into equation (5.6) and see what happens to $\bar{\ell}$. We find that when cost is proportional to the number of connections the lowest value of $\bar{\ell}$ one can hope for is:

$$\bar{\ell} \approx \sqrt{8kn} \quad (5.13)$$

Putting $n = 1000$ and $k = 1$ this gives $\bar{\ell} = 89.4$, one can see this agrees well with the exact result shown in figure (13).

If we consider the cost where $c = k(np)^2$ we get:

$$\bar{\ell} \approx \sqrt[3]{27kn^2} \quad (5.14)$$

This always gives a slight over estimate as there are some slightly lower order terms that contribute, however it provides the correct scaling relationship for $\bar{\ell}$ in the high n limit.

These last results are very interesting; they show that the average shortest path across the network grows like $n^{\frac{1}{2}}$ when we impose a cost per connection to the hub, or $n^{\frac{2}{3}}$ when we put

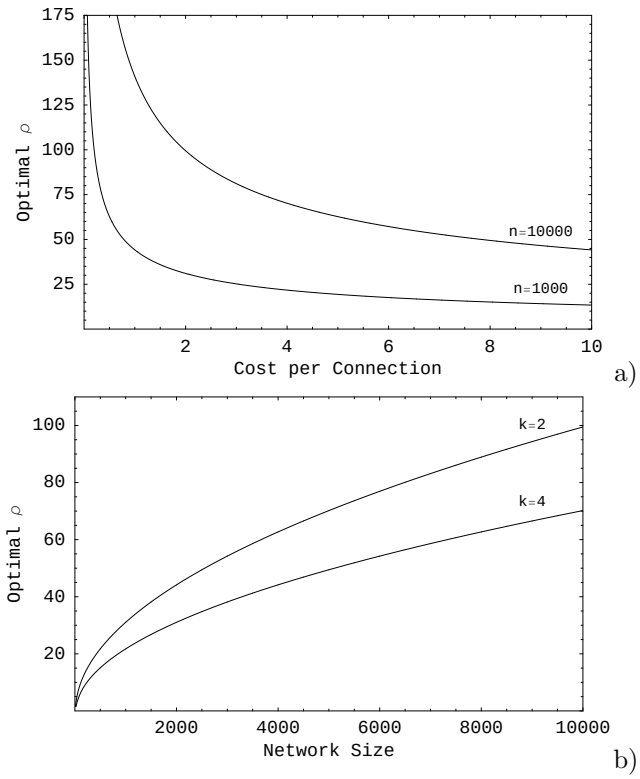


Figure 14: Numerically solving equation (5.5) for minimum $\bar{\ell}$. a) Optimum number of connections for different costs per connection. Curves for $n = 1000$ and $n = 10000$ are drawn b)Fixing the cost per connection,$k = 2$ and $k = 4$, how does the optimal number of connections vary according to network size.

a cost on the number of direct connections between nodes made via the hub. In both cases the small-world nature of the network has been destroyed. To be a small-world network the average shortest path scales no faster than $\log n$. Of course in a practical situation where the network isn't too large the average path can still be very small, the point is that if you then doubled your network size the average shortest path would increase by about 50%, where as in a small world network you would *add* on at most $\log 2$ links [3].

5.3 Result for Undirected Links

Once again we've only treated the directional case. The non-directional results can be easily obtained by using the variable change $P_{undir}(\ell, c) \approx 2P_{dir}(2\ell, 2c)$.

[3] where we do not specify the base of the logarithm

We get $\bar{\ell}$ in the usual way:

$$\bar{\ell} = \sum_{\ell=1}^{\frac{n-1}{2}} \ell P_{undir}(\ell, c) \tag{5.15}$$

$$= \sum_{\ell=1}^{\frac{n-1}{2}} 2\ell P_{dir}(2\ell, 2c) \tag{5.16}$$

Defining $\ell' = 2\ell$ and $c' = 2c$ we can make this look just like the directed case. ℓ' is a dummy variable but c' is not. When we make the change of variable we must remember that we are now increasing ℓ' in jumps of two so should account for the extra terms in the sum with a factor of a half:

$$\bar{\ell} = \frac{1}{2} \sum_{\ell'=1}^{n-1} \ell' P_{dir}(\ell', c') \tag{5.17}$$

$$\Rightarrow \bar{\ell}_{undir}(c) = \frac{1}{2} \bar{\ell}_{dir}(2c) \tag{5.18}$$

This relationship allows us to now write down the results for the undirected case by adjusting equations (5.10) and (5.13):

$$\bar{\ell} \approx \sqrt{4kn} \tag{5.19}$$

$$\rho \approx \sqrt{\frac{n}{k}} \tag{5.20}$$

for the minimum average geodesic path and optimum connections with a cost $c = knp$ and non-directional links.

5.4 Comparing with other Cost Models

In a recent paper Stanley et.al.[10] added costs to the random links in the W-S model. To simulate the fact that in the real world not all connections are of the same quality they added a distribution of costs to the different connections. They then looked at two situations: So called 'strong disorder' is when the shortest path is dominated by the cost of the most expensive link in the chain. An internet user who is using a slow dial-up connection will never get faster than 56kbps no matter what is attached to their ISP. 'Weak disorder' is where the path is found by summing over all the costs on the way and no one link dominates.

Their simulations found that whilst in the weak disorder limit the small-world effect can prevail, in the strong disorder limit it was destroyed and that the average shortest path scaled as $\bar{\ell} \propto n^{1/3}$. Their model is significantly more complicated than the exactly solvable one that we have presented , however, the fact that all the shortcuts go through the central hub means the hub could perhaps be viewed as being the 'limiting factor' and so putting the network in the strong disorder category.

Any comparison however should be cautious. The cost arose in our model from congestion caused by almost all the paths going through the central hub. In Stanley et.al's model the cost is seen as links just naturally having a different quality. If one of their links finds that it is taking a large volume of traffic then it will perform just as well.

6 Many Identical Hubs

The two hub networks that we studied in section (3) become a lot simpler if we set $p = q$ and create two identical hubs. They won't be connected to the same nodes but on average they will be connected to the same number. In fact the problem becomes so much simpler that it allows us to consider what happens if we have N identical hubs. There is still an infinite cost for using both hubs.

In section (3.1) we wrote down the probability intuitively as the probability the shortest path between two nodes is length ℓ through the p-hub *and* not shorter than ℓ through the q-hub, *or*, the shortest path is ℓ through the q-hub *and* not shorter than ℓ through the p-hub. In general, $P(A \text{ or } B) = P(A) + P(B) - P(A)P(B)$, however, the last term in our case can be simplified because the only situation that satisfies this condition is the shortest path being ℓ through both hubs simultaneously.

More precisely:

$$P(\ell < k) \quad = \quad P_s(p,\ell)\big[1 - \sum_{i=1}^{\ell-1} P_s(q,i)\big] + P_s(q,\ell)\big[1 - \sum_{i=1}^{\ell-1} P_s(p,i)\big] - P_s(p,\ell)P_s(q,\ell) \tag{6.1}$$

where $P_s(p,\ell)$ is the probability that the separation is ℓ using a path through the p-hub. If we replace the terms in the square brackets with $R(p,\ell) = 1 - \sum_{i=1}^{\ell-1} P_s(p,i)$ and set $p = q$ we get:

$$P(\ell < k) = 2P_s(p,\ell)R(p,\ell) - P_s(p,\ell)^2 \tag{6.2}$$

From here on it will be assumed that $P_s \equiv P_s(p,\ell)$ and $R \equiv R(p,\ell)$. Adding in another hub that also has probability, p, one can verify that we now get:

$$P(\ell < k) = 3P_sR^2 - 3P_s^2R + P_s^3 \tag{6.3}$$

and for a arbitrary number of hubs each with probability p we can write down the general form:

$$P(\ell < k) = R^N - (R - P_s)^N \tag{6.4}$$

To generalise further we have to split off into the specific cases, directed or undirected.

6.1 Directed Links

This is as usual the simplest case. The function $P(\ell < k)$ is the same for every value of ℓ. We can thus write down $P(\ell = k)$ as:

$$P(\ell = k) \quad = \quad 1 - \sum_{i=1}^{k-1} R^N - (R - P_s)^N \tag{6.5}$$

R and P_s are both first order in ℓ so we can rewrite equation (6.5) as:

$$P(\ell = k) = 1 - \sum_{i=0}^{N} g_i(p) f_i(a,\ell) \tag{6.6}$$

where the functions $g_i(p)$ and $f_i(a,\ell)$ have the same meaning as in earlier sections and are yet to be determined. The final distribution is now easily obtained by summing over all values of k as usual and gives us:

$$P(\ell) = \frac{1}{n-1}\big[1 - \Big(\sum_{i=0}^{N} g_i(p) f_i(a,\ell)\Big) + (n-1-\ell)(R^N - (R-P_s)^N)\big] \tag{6.7}$$

It is now left to fill in the form of the function $g_i(p)$. We can get at them by filling in the explicit forms of R and P_s. In section (2.4.1) we showed these to be:

$$P_s(p,\ell) = \ell p^2 (1-p)^{\ell-1} \tag{6.8}$$
$$R(p,\ell) = (1-p+p\ell)(1-p)^{\ell-1} \tag{6.9}$$

Substituting into equation (6.4) we get:

$$P(\ell < k) = \Big[(\ell p + (1-p))^N - (\ell p(1-p) + 1 - p)^N\Big](1-p)^{N(\ell-1)} \tag{6.10}$$

Expanding this into powers of ℓ gives us the g_i functions:

$$g_i(p)\ell^i = \alpha_i\big[p^i(1-p)^{N-i} - p^i(1-p)^N\big]\ell^i \tag{6.11}$$

The α_i are factors from Pascal's triangle and are given by:

$$\alpha_i = \frac{N!}{i!(N-i)!}$$

So finally we get:

$$g_i(p) = \frac{N!}{i!(N-i)!} p^i (1-p)^N \big((1-p)^{-i} - 1\big) \tag{6.12}$$

It is useful to test these against the values of the g_i functions from the general two hub case and indeed they match. The final distribution is now given by using the g_i functions in equation (6.7) to get:

$$P(\ell) = \frac{1}{n-1}\big[1 + \sum_{i=0}^{N} g_i(p)\Big((n-1-\ell)\ell^i(1-p)^{N(\ell-1)} - f_i(a,\ell)\Big)\big] \tag{6.13}$$

where $a = (1-p)^N$

6.2 Undirected Links

Throughout this report we've seen that in the scaling limit the directed and undirected networks are equivalent. For this reason we will state the result for the undirected case:

$$g_i(p) = \frac{N!(1-p)^{-2N}}{i!(N-i)!}\left[p^i(2-p)^i\big((1-p)^2-2p\big)^{N-i} - \big(\big(1-p(2-p)\big)p(2-p)\big)^i\big(1-4p+5p^2-2p^3\big)^{N-i}\right] \tag{6.14}$$

$$P(\ell \geq 2) = \frac{2}{n-1}\left[(1-p^2)^N + \sum_{i=0}^{N}\Big(\big(\frac{n-1}{2} - \ell\big)g_i(p)\ell^i(1-p)^{2N(\ell-1)} - g_i(p)f_i'\big((1-p)^{2N}, n\big)\Big)\right] \tag{6.15}$$

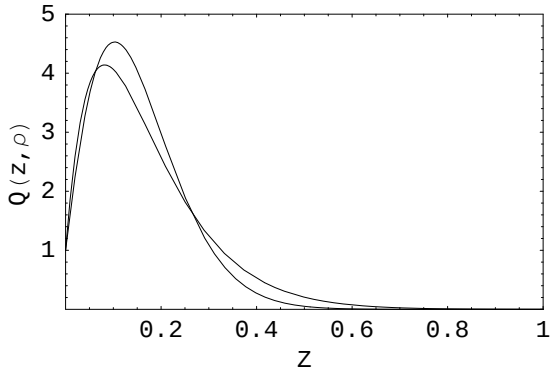


Figure 15: The top curve that peaks highest in this picture is a network with 17 hubs and $\rho = 2$. The other curve is one hub with $\rho = 10$. The latter has roughly 2.5 times as many connections yet has a similar average z. The network with only one hub peaks earlier and comes down more slowly. This is because the hub creates very highly connected regions and other very poorly connected regions. When the connections are added more randomly (as in the many hub case) these regions are much less pronounced.

6.3 Analysing the N-Hub Results

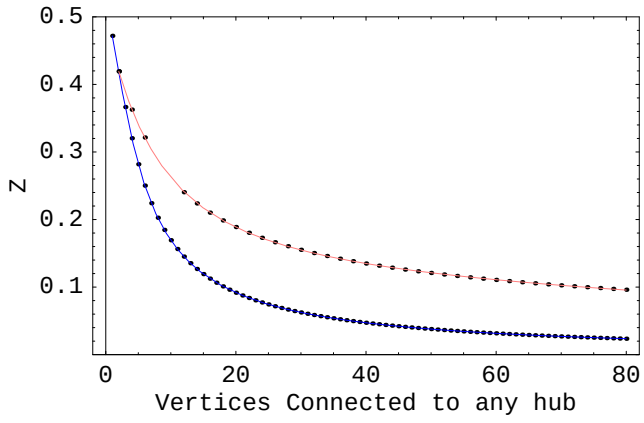
As was discussed in section (3.3) the model proposed by Dorogovtsev and Mendes can be viewed in two ways. Either there really *is* a central hub that we make random connections of length 1/2 to, or there is no hub and all the nodes that were attached to it are instead attached to each other by links of length 1. If we view the model in this way then a certain element of randomness has been removed compared to the model of Watts and Strogatz [6]. The D-M model creates a few nodes with a high degree where as in the W-S model every node has the same degree. Neither models reproduce what we see in nature which is a power-law degree distribution. For the rest of this section we will use the idea that each node that is connected to the hub is really directly connected by $\rho - 1$ separate unit links to every other connected node.

The results derived earlier in this section may allow for some treatment of the cross over between the D-M and W-S models. An interesting starting point is to set $\rho = 2$ for the N hub case. On average this is equivalent to each hub having two connections and thus represents N totally random connections across the network and will create a picture that is more familiar with the W-S model.

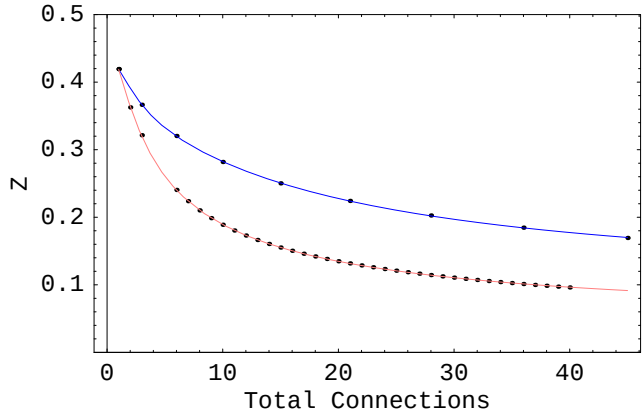
Consider a system where you are only allowed a fixed number of connections. Is it better to put them in randomly, creating a large number of nodes with a small degree, or is it better to create a small number of nodes with high degree? The N hub solution lets us look at this question. We can relate the number of hubs, the number of connections C, and the ρs by the relation:

$$C = \frac{N\rho(\rho - 1)}{2} \tag{6.16}$$

The degree for nodes attached to a hub is given on average by the ρ for that hub, this degree is on top of the given attachments around the ring. There is a small probability a given node will be attached to two different hubs and this will create a rapidly diminishing exponential degree distribution.



a)



b)

Figure 16: Average scaled path across the network against a) number of nodes connected to a hub and b) total number of connections. The network using N hubs with $\rho = 2$ is on top in a) and on the bottom in b). The other curve is a single hub with a variable ρ.

In figure (16) we compare a network made with a variable number of N hubs each with $\rho = 2$, and a network with a single hub and variable ρ. Using equation (6.16) we can relate the two networks by how many connections there are. The curves were generated by numerically integrating over $zP(z, \rho)$ for the directed case to calculate the average path across the network. Figure (16b) clearly shows that the small world effect kicks in much faster when we add the connections completely at random rather than restricting them to a small number of nodes.

How might we go about looking at the cross over between the W-S and N-hub D-M models? Once again fixing $\rho = 2$, we look at varying N. N now represents the number of random cross connections and so we relate this to the variable in the W-S model by $N = np'$ where p' is now the probability of rewiring from the W-S model (see section 2.3). To understand what will happen next we need to fully understand the summation functions $f_m(a, n)$ (see appendix A) and how they behave for high N. It would be interesting to see if this gives the same logarithmic scaling form for the average shortest path as the W-S model.

7 Conclusion

In this report we've derived a number of new results based on the small-world model set up by Dorogovtsev and Mendes that models networks where the far connections are usually through some common centre.

In extending this model we explored two different areas of interest:

1. Changing the number of centres, looking at how the different ways of wiring up the network affect its behaviour.

2. Addressing the problem that in the real world if most long distance connections are made through a central hub then this may cause congestion.

We started off by adding an extra central hub. To do this we had to assume an infinite cost for using both, although this is a compromise it is not hard to imagine systems that use central hubs where those that use them are forced to only choose one. The results showed that two hubs with probability of attachment p and q are not the same as having one hub with probability $p+q$, whilst if we allow both hubs to be used the parts of the network that connect the two hubs serve to make the distribution look more like that of a single hub.

By simplifying the problem a little bit further and only using identical hubs ($p = q$) we were able to find the distribution for N hubs. This showed that if the hubs are thought of as making connections between pairs then it is better to create many links between random pairs than create a small amount of nodes highly connected.

The average degree of the nodes connected to a hub, i, is ρ_i. With the two hub result this allows us to create a certain number of nodes with degree ρ_p and another number of nodes with degree ρ_q. A system of N identical hubs creates N nodes with degree ρ_N. Although we haven't talked about it much in this report, in many real-world networks we find that the degree distribution instead of falling off quickly (exponentially) has the form, $P(d) \simeq d^{-\gamma}$ where $P(d)$ is the probability a randomly chosen node will have degree d and γ the scaling coefficient. An interesting avenue of further study might be to see if by using multiple hubs a power-law distribution can be created and the network solved for $\bar{\ell}$.

Another interesting lead that the N hub result gives us is the possibility of solving a network that doesn't have a central hub but instead has completely random connections wired across it in a manner more akin to the original small-world model by Watts and Strogatz.

To tackle the problem of how congestion might effect our small-world networks we added a cost for travelling through the hub. This leads to one of our central results: If the cost is proportional to the number of connections made to the central hub then the small-world effect is destroyed and the average shortest path now scales as $\bar{\ell}_{min} \propto n^{1/2}$. If the cost is such that it is not how many connections that are made to the hub, but instead how many *pairs* it has to service, then we find the average shortest path scales like $\bar{\ell}_{min} \propto n^{2/3}$.

We also derived the optimum number of connections that should be made if a cost is incurred for using a central route. It goes some way to answering the question "at what point is it better to take the long route and avoid the congestion?". These results could apply to a range of networks including biological supply networks, communication networks and computer networks.

The distributions for two hubs with two different costs c_p and c_q were also derived. These distributions were plotted but they haven't been studied in much detail. A possible way to extend the congestion problem might be to look at how the system behaves when there are two competing hubs, perhaps the cost associated with each hub could be coupled and the dynamic between them studied.

References

[1] Newman M. E. J., "Ego-centered networks and the ripple effect", *Social Networks* **25**, 83-95 (2003).

[2] Killworth P. D. ,Johnsen E. C., Bernard H. R.,Shelley G. A. , and McCarty C., "Estimating the size of personal networks". *Social Networks*, **12**, 289312 (1990).

[3] Milgram S., "The Small World Problem", *Psychol Today* **2**, pp 60 - 67, (1967)

[4] Newman M. E. J., "The structure and function of complex networks", *SIAM Review* **45**, 167-256 (2003). cond-mat/0303516.

[5] Dorogovtsev S.N. and Mendes J. F. F.,*Adv. Phys.* **51**, 1079-1187 (2002). cond-mat/0106144.

[6] Watts D. J. and Strogatz S. H., "Collective dynamics of small-world networks", *Nature* **393**, 440-442 (1998)

[7] Newman M. E. J. and Watts D. J., "Renormalization group analysis of the small-world network model" , *Phys. Lett. A* **263**, 341-346 (1999). cond-mat/9903357.

[8] Dorogovtsev, S. N. and Mendes, J. F. F., "Exactly solvable small-world network", *Europhys. Lett,* **50**(1), pp. 1-7 (2000).

[9] Buchanan M., "Know thy neighbour", *New Scientist* vol **181** issue 2430, p32 (2004)

[10] BraunsteinL. A., Buldyrev S. V., Cohen R., Havlin S., and Stanley H. E., "Optimal Paths in Disordered Complex Networks", *Phys. Rev. Lett.* **91**, 168701 (2003).

[11] Bollobás B., "Random Graphs", Academic Press, New York, 2nd Ed (2001).

A Appendix - Summation functions

A.1 f_m functions

Here are listed the summation functions used through out this report. For the directed cases we use:

$$f_m(a,n) = \sum_{i=1}^{n-1} i^m a^{i-1} \tag{A.1}$$

The functions are related by the recursion relation:

$$f_{m+1}(a,n) = \frac{d}{dx}\left(a f_m(a,n)\right) \tag{A.2}$$

Listed below are the first few examples, and the most commonly used in this report.

$$f_0(a,n) = \frac{1}{a}\left(\frac{1-a^n}{1-a} - 1\right) \tag{A.3}$$

$$f_1(a,n) = \frac{1-a^n}{(1-a)^2} - \frac{na^{n-1}}{1-a} \tag{A.4}$$

$$f_2(a,n) = \frac{a(1-a^n)}{(1-a)^3} + \frac{(1-(2n+1)a^n)}{(1-a)^2} - \frac{n^2a^{n-1}}{1-a} \tag{A.5}$$

A.2 f'_m functions

The undirected case often calls for the sum to start from two rather than one so we define:

$$f_m(a,n) = \sum_{i=2}^{n-1} i^m a^{i-1} \tag{A.6}$$

The recursion relation is the same as in (A.2) with the first few equations being:

$$f'_0(a,n) = \frac{1}{a}\left(\frac{1-a^n}{1-a} - (1+a)\right) \tag{A.7}$$

$$f'_1(a,n) = \frac{1-a^n}{(1-a)^2} - \frac{na^{n-1}}{1-a} - 1 \tag{A.8}$$

$$f'_2(a,n) = \frac{a(1-a^n)}{(1-a)^3} + \frac{(1-(2n+1)a^n)}{(1-a)^2} - \frac{n^2a^{n-1}}{1-a} - 1 \tag{A.9}$$

B Appendix - Source Code

Included in this appendix is the C++ source code that was used to test the results in the report. To create random numbers that were sufficiently 'random' required using the Mersenne Twister source code[4]. Many different permutations of this code were used, however the version below contains the most important points.

The basic method for finding the shortest path between two points is to do it like a person would do it. Find the shortest distance to the hub, find the shortest distance from the hub to the target - and you're there. It is advisable to look through function main first (towards the end of the code) and then look at the relevant procedures when they arise.

```
#define N               // Size of network
#define N_plus          //N+1
#define SAMPLES_PER_RUN // Instead of all pairs we can take random pairs
#define RUNS            // How many networks to make
#define N_HUBS          // The number of hubs in the network
#define COSTP
#define COSTQ

// GLOBAL

bool link_matrix [N][N_HUBS] ; int bridge[N_HUBS][N_HUBS];

// procedures

void smallest_bridges() {
    int i, j, k;
    int bridge_length, bridge_min;
    int fwd,start;
    bool no_connection;

    // Find the smallest bridges, we only need to this once per network
    // A bridge links two hubs together.  We need to analyse N_HUBS*(N_HUBS-1) of them

    for (i=0;i<N_HUBS;i++)
    {
        for (j=0;j<N_HUBS;j++)
        {
            // First find someone connected to the hub and then do a loop
            // round the network looking for the shortest links

            no_connection=false;
            fwd=0;
            bridge_min = N;

            while ((!link_matrix[fwd][i]) && (!no_connection))
            {
                fwd++;
                if (fwd > (N-1)) no_connection = true;
            }

            // we're now in position to start
            start=fwd;

            if (!no_connection)
            {
                for (k=0;k<N;k++)
                {

                    if (link_matrix[fwd][j])
                    {
                        bridge_length = abs(fwd - start);
                        if (fwd < start) bridge_length = N - bridge_length;

                        if (bridge_length<bridge_min) bridge_min = bridge_length;
                    }
                    fwd ++;
                    if (fwd > (N-1)) fwd=0;
```

[4] http://www.math.sci.hiroshima-u.ac.jp/ m-mat/MT/emt.html

```
                    // If we run over a connection to the starting hub
                    // we have to rest the counting

                    if (link_matrix[fwd][i]) start = fwd;
                } // end for k
            } // end if no connection
            bridge[i][j]=bridge_min;
        } // end for j
    } // end for i

    // BECAUSE THIS IS A NON-DIRECTIONAL CASE I'M GOING TO FORCE SYMMETRY ON THE MATRIX
    for (i=0;i<N_HUBS;i++)
    {
        for (j=0;j<N_HUBS;j++)
        {
            if (bridge[i][j]<bridge[j][i]) bridge[j][i]=bridge[i][j];
            else bridge[i][j]=bridge[j][i];
        }
    }

} // end function

int minimum_length(int start, int target) {

    // THIS FUNCTION WORKS OUT THE MINIMUM DISTANCE BETWEEN ANY TWO POINTS
    // ON THE NETWORK.  THERE ARE ONLY A FEW POSSIBLE "MINIMUM" ROUTES THAT
    // CAN BE FOLLOWED.  ALL ARE EXPLORED AND THE SHORTEST IS RETURNED
    // THE smallest_bridges() PROCEDURE MUST BE RAN PRIOR TO CALLING THIS FUNCTION

    int fwd,bwd;            // position indexes on the network
    int l_min;          // shortest path
    int target_temp, i, j;      // working variables and indexes
    int temp_length;

    int start2hub[N_HUBS];      // shortest distance from a vertex to each hub
    int target2hub[N_HUBS];     // shortest distance from each hub to a vertex
    int physical_path;      // if all else fails don't use a hub

    bool found_hub;         // working variable

    // Work out the physical path

    physical_path = abs(target - start);

    if (physical_path>(N/2))
        physical_path = N - physical_path;

    // Work out the distance of the start to the hubs

    for(i=0;i<N_HUBS;i++)
    {
    start2hub[i]=0;

    found_hub=false;

    fwd=start;
    bwd=start;

        if (link_matrix [start][i]) found_hub=true;

        while (!found_hub)
        {
            fwd++;          // Move round both ways
            bwd--;
            // Keep us on the circle
            if (fwd>(N-1)) fwd=0;
            if (bwd<0) bwd=(N-1);

            if ((link_matrix [bwd][i]) || (link_matrix [fwd][i])) found_hub=true;
            start2hub[i] ++;        // Increment the distance;

            if (start2hub[i] > (N-2)/2) found_hub=true;
        }
```

```
    }

    // Work out the distance of the target to the hubs

    for(i=0;i<N_HUBS;i++)
    {

    target2hub[i]=0;

    found_hub=false;

    bwd=target;
    fwd=target;

        if (link_matrix [target][i]) found_hub=true;

        while (!found_hub)
        {
            bwd--;          // Move round bothways
            fwd++;

            // Keep us on the circle
            if (bwd<0) bwd=(N-1);
            if (fwd>(N-1)) fwd=0;

            if ((link_matrix [bwd][i]) || (link_matrix [fwd][i])) found_hub=true;

            target2hub[i] ++;       // Increment the distance;

            if (target2hub[i] > (N-2)) found_hub=true;
        }
    }

    // We now know how far to each hub from each vertex and the smallest bridges
    // we just need to choose which path is shortest from
    //
    // vertex -> ... -> p -> ... -> vertex
    // vertex -> ... -> q -> ... -> vertex
    // vertex -> ... -> p -> bridge -> q -> ... -> vertex
    // vertex -> ... -> q -> bridge -> p -> ... -> vertex
    // vertex -> ............... -> vertex
    // We can also extend this to N hubs by using a loop, although if we allow
    // bridges the problem becomes large.

    l_min = start2hub[0] + target2hub[0] + 1 + COSTP;

    temp_length = start2hub[1] + target2hub[1] + 1 +COSTQ;
    if (temp_length < l_min) l_min = temp_length;

/*  HERE WE DO NOT ALLOW THE BRIDGES SO THESE LINES ARE COMMENTED
OUT
    temp_length = start2hub[0] + bridge[0][1] + target2hub[1] + 2;
    if (temp_length < l_min) l_min = temp_length;

    temp_length = start2hub[1] + bridge[1][0] + target2hub[0] + 2;
    if (temp_length < l_min) l_min = temp_length;
*/
    if (physical_path < l_min) l_min = physical_path;

    return l_min;

}

//********************
//  FUNCTION MAIN
//********************

void main() {
    long i,j,k,l;       // Position in array
    int x;          // Working integer
    double p[N_HUBS];   // Probability linked to hub
    double N_pairs, counts; // Used for normalising data
    time_t seconds;     // Time variable
    ofstream datafile;  // Output file
```

```
    long cumulative[N_plus];// Cumulative Frequency

    for (i=0;i<N+1;i++) // Initialize variable
    {
        cumulative[i]=0;
    }

    for (j=0;j<N_HUBS;j++)
    {
        cout << "p" << j << ":";
        cin >> p[j];
        cout << endl;
    }

    time(&seconds);

    TRandomMersenne random_number((int) seconds);
    TRandomMersenne integer_number(((int) seconds)/2);

    l=0;
    for (k=0;k<RUNS;k++)
    {

        if (l==100)
        {cout << k << endl;
        l=0;}
        l++;

        //  Create the connections to the hub

        for (j=0;j<N_HUBS;j++)
        {

            for (i=0;i<N;i++)
            {
                if (random_number.Random() > p[j])
                {
                    link_matrix[i][j]=false;
                }
                else
                {
                    link_matrix[i][j]=true;
                }

            }
        }

        smallest_bridges(); // Used to calculate shortest paths

        // Now calculate the paths between random vertices
        for (i=0;i<SAMPLES_PER_RUN;i++)
        {
            x=minimum_length(integer_number.IRandom(0,N-1),integer_number.IRandom(0,N-1));
            // Count number of links length x
            cumulative[x]++;
        }

    } // END k LOOP

    // Create csv file
    datafile.open("data/athnd.csv");

    for (i=1;i<N;i++)
    {
        counts = ((double) cumulative[i] / (SAMPLES_PER_RUN * RUNS));

        datafile << i << "," << counts << endl;
    }
    datafile.close();
}

\\ END OF CODE
```