# Range Dependent Random Graphs and their Application to Modelling Large Small World Proteome Data Sets

**Peter Grindrod**
Numbercraft Limited
Magdalen Centre
The Oxford Science Park
Oxford OX4 4GA
peterg@numbercraft.co.uk

## Introduction

The study of random graphs, long dominated by the work on the Erdos-Renyi model [1], *G(n,p)*, where an edge between any pair from *n* vertices is present with a certain probability, *p*, has recently been extended to small world graphs [2,3,4]. The desire is to generate graphs with high degree of clustering (tendency for adjacency to be transitive) as well as relatively short paths between all vertices. The most analysed model for such a graph is the Watts-Strogatz graph, where a cyclic lattice (with all k near neighbours connected) is occasionally rewired randomly. In essence this superimposes two graphs: a cyclic lattice introducing local clustering effects, and a random graph producing much longer scale adjacencies. The lattice (or partial lattice) embedded within the graph introduces a natural idea of scale, or range, associated with each edge. The clustering behaviour derives from the lattice, whilst short connection paths derive from the random graph.

The split between (sub)graphs on two scales within the Watts-Strogatz graph suggests we consider other graphs, derived from superposing many (sub)graphs at many distinct length scales. This may seem more natural than a two scale model, providing that the density and scale of the separate sub graphs are properly related so that the final graph has a well behaved vertex degree distribution. In our case this relation follows a power scaling law: the larger the scale the sparser the sub-graph. This thinking is analogous to that underlying fractal (self-affine) structures, obeying scaling laws over a range of different length scales (differing by orders magnitudes, if not actually from the infinitesimal to the infinite). These graphs are the subject of this paper.

We show how such graphs can be defined and parameterised by two simple parameters, and generated stochastically in a manner analogous to the Erdos-Renyi model where the probability of an edge existing is range (scale) dependent. The vertices are to be thought of as ordered in a possibly incomplete one dimensional lattice, so that all edges inherit a natural length scale or range, derived form the distance between the end vertices in the underlying lattice ordering.

An interesting point is that such graphs may be defined over an infinite number of vertices, possessing edges on an infinite number of scales, whilst the degree distribution has finite moments. In fact the mathematics of the generating function and the clustering coefficient is more elegant in the infinite case since there are no truncated series arising from edge effects.

Next we turn to a practical problem: the inverse problem. This problem does not readily arise with simple random graphs, since all edges are equally likely and the vertices are unordered. Suppose we are given a large sparse graph, as a list of vertices and edges, which we believe has been generated by, or can be modelled by, a suitably parameterised version of our model. Then we wish to order the vertices of the given graph in such a way that that it is most likely to have been generated. This yields extra information which can be appended to the data, since once ordered every edge inherits a natural length scale. Of course the ordering must reflect the probabilistic occurrences of edges of all length scale: hence it must respect the local and global structure of the graph.

We will introduce a maximum likelihood method to realise a given graph as a member of our class of graphs (suitably calibrated by global properties of the data). This method can be verified directly for graphs originating from the model class whose vertices have been shuffled to hide the underlying structure.

In practice when we are given information to be interpreted as a graph, it may contain errors: actual edges which are missing in the data, and edges which are erroneously present in the data. We demonstrate that our proposed solution to the inverse problem is robust to small numbers of errors of these types.

The applications we have in mind arise in bioinformatics, where high through put devices mean that that large amounts of gene to gene, or protein to protein interaction data will become increasingly available, both within commercial and public research. The relationships between genes, or the proteins they code for, and (intra cellular up to organism) functions are "many to many". This is directly observed and also a logical consequence of the size of the genome(s) (typically thousands to tens of thousands of genes) when considered in relation to the plethora of such functions. However early work, for example in the analysis of co-expression data from micro arrays has used clustering and discrimination concepts, which are inherently "many to one". Therefore graph theoretic approaches to describing and modelling the structure of all gene to gene or protein to protein relationships offers a step forwards. Nodes/vertices represent proteins (or genes) whilst edges represent associations. These graphs will be large and sparse. The data is also likely to contain errors of both types.

We illustrate both the framework and methods developed in this paper with an example application to the yeast proteome.

**Basic definitions**
Here we propose a simple model for a class of sparse graphs which inherit a simple notion of inter-vertex length scale, or range, by being embedded within a possibly incomplete one dimensional lattice. Generalisations to a cyclic lattice are immediate. The motivation for this is to define a suitable class of stochastic graphs which

- may show the small world characteristics of "localised" clustering coupled with longer range connectivity;
- are amenable to analysis, and characterised by simple global parameters;
- have a hierarchy of edges on different scales (ranges), for which the successively "longer range" edges are less and less likely to exist;
- may be used as candidate frameworks within which to resolve inverse problems via maximum likelihood or other optimization methods.

We begin by considering classes of sparse graphs that are defined over an infinite number of vertices, and possess an infinite number of edges, such that the average vertex degree is finite. In practice we will be interested in finite versions of such graphs, simply truncated, but the analysis of the properties in the infinite case is more elegant, and provides some useful insights.

**Graphs on infinite vertices**
Consider a graph based on a one dimensional enumeration of vertices $v_k$ (for $k$ = …-2, -1, 0, 1, 2,…), where the probability that there is an edge connecting vertex $v_i$ to vertex $v_j$ is given by a function in the form

$$p_{ij} = f(|\, j - i\,|),$$

where $f$ maps the positive integers onto [0,1], and is such that $f(k)$ tends to zero as $k$ tends to infinity.

We define the **range** of the edge to be $|i\text{-}j|$. Note that the probabilistic structure is invariant to translation and reflection with respect to the underlying vertex ordering.

We introduce the generating function, $G_0(x)$ [5] for the probability distribution of vertex degree, defined by

$$G_0(x) = \sum_{j=0}^{\infty} P_j x^j ,$$

where $P_j$ is the probability that a randomly chosen vertex has degree $j$.

In our case we can express this as an infinite product, by considering the possible edges connected to an arbitrary vertex, $v_0$ :

$$G_0(x) = \prod_{\substack{k\neq 0\\ -\infty}}^{\infty} ((1 - p_{0k}) + p_{0k} x) .$$

As before $p_{0k}$ is the probability that $v_0$ is adjacent to $v_k$. This follows since the coefficient of $x^k$ is precisely the probability that $v_0$ is adjacent to exactly $k$ distinct vertices, summing over all such independent combinations. Hence we have

$$G_0(x) = \prod_{k=1}^{\infty}(1 + f(k)(x-1))^2.$$

Now consider the specific class of graphs where the probability that vertex $v_i$ is connected to vertex is $v_j$ by an edge is given by the power law form

$$p_{ij} = f(|j-i|) = \alpha\lambda^{|j-i|-1}.$$

Here the parameters α and λ are in (0,1]. If α=1 then neighbours are certainly connected, by edges with range 1, and the graph contains a Hamiltonian path connecting all immediate neighbours, regardless of λ. If α<1 then global connectedness depends on both parameters. As λ increases from zero the expected number of the long range associations increases.

Our graph could be thought of as the superposition of many subgraphs, with each subgraph containing only edges of a certain range, k, say k=1,2,…, which are present with probability $\alpha\lambda^{k-1}$ .

Our first task is to show how the global parameters, α and λ, relate to some of the graph's global characteristics.

Consider a given vertex, say $v_0$, then the expected number of neighbours is given by a geometric progression:

$$z=2\alpha/(1-\lambda).$$

Now, since $p_{0k} = \alpha\lambda^{|k|-1}$, we have a generating function in the form

$$G_0(x) = \prod_{k=1}^{\infty}(1 - \alpha\lambda^{k-1} + \alpha\lambda^{k-1}x)^2 .$$

We have, by direct calculation,

$$G'_0(x) = 2G_0(x)\sum_{k=1}^{\infty}\alpha\lambda^{k-1}/(1-\alpha\lambda^{k-1} + \alpha\lambda^{k-1}x).$$

It is clear that $G_0(1) = 1$, and $G'_0(1) = z = 2\alpha/(1-\lambda)$, the expected vertex degree. Successive derivatives of $G_0(x)$, evaluated at $x = 1$, can be used to calculate successive moments of the vertex degree distribution. Similarly we may calculate values for $P_k$ from successive derivatives of $G_0(x)$ evaluated at $x = 0$. In fact, if α=1, then both $G_0(0) = 0$, and $G'_0(0)=0$, so that $P_0 = P_1=0$: every vertex has at least two neighbours.

We also have,

$$G''_0(x) = \frac{G'_0(x)^2}{G_0(x)} - 2G_0(x)\sum_{k=1}^{\infty}(\alpha\lambda^{k-1})^2/(1-\alpha\lambda^{k-1} + \alpha\lambda^{k-1}x)^2.$$

Hence

$$G''_0(1) = \frac{z^2}{2}.\left\{\frac{1+3\lambda}{1+\lambda}\right\},$$

which is the expected number of second neighbours of a vertex, denoted by $z_2$.

Now, $z_2$ behaving like the square of $z$ is the sort of behaviour seen in random graphs (normal Poisson random graphs for instance - however these do not show the clustering behaviour discussed below).

The Watts-Strogatz **clustering number**, $C$, defined for the graph, is a measure of the tendency of adjacency to be transitive. It is defined as follows. Consider all connected triplets of vertices, that is triplets $\{v_i, v_j, v_k\}$ having two edges connecting $v_i$ and $v_j$, and $v_j$ and $v_k$. Then C is the fraction of these for which there is also an edge connecting $v_i$ and $v_k$ directly, completing the triangle.

Consider an arbitrary vertex, say $v_0$, and the possible connected triplets centred there. We have

$$C = \frac{\sum_{\substack{i\neq 0\\ j\neq 0\\ j>i}} p_{oi}\, p_{oj}\, p_{ij}}{\sum_{\substack{k\neq 0\\ l\neq 0\\ l>k}} p_{ok}\, p_{ol}}.$$

The denominator gives the expected number of connected triplets $\{v_i, v_0, v_j\}$ centred at $v_0$. The numerator gives the expected number of triangles $\{v_i, v_0, v_j\}$ centred at $v_0$. Since $v_0$ was arbitrary these are the same for all vertices. Hence the ratio gives the fraction of all connected triplets which are also triangles. It is straightforward to deduce that

$$C = \frac{3\sum_{i=1}^{\infty}\sum_{j=1}^{\infty} f(i)\, f(j)\, f(i+j)}{\frac{1}{2}G_0''(1)}.$$

By substituting directly for the probabilities, we have, by direct calculation,

$$C = \frac{3\alpha\lambda}{(1+\lambda)(1+3\lambda).}$$

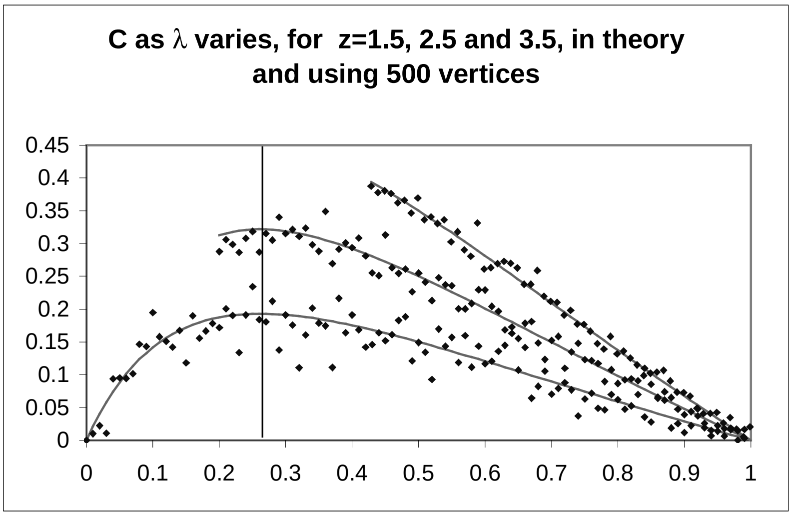
Note that for $\alpha = 1$, as $\lambda$ tends to 1, $C$ tends to 3/8. This is because the graph approaches a completely connected graph in a non-uniform way – there are always vertices far enough away to make adjacency improbable.

$C$ is not a monotonic function of $\lambda$. For fixed $\alpha$ there is a local maximum at $\lambda = 3^{-1/2}$. Hence, there is a kind of "optimum" clustering connectivity at this maximum. If $\lambda$ increases further the probability of having a long range neighbour which is not also connected to more localised neighbours increases, hence $C$ decreases.

If we set $\alpha = z\,(1-\lambda)/2$ , then with the average vertex degree, *z,* fixed, $\lambda$ may vary in (max{0,1-2/z},1]. Hence with this parameterisation $\lambda$ controls the balance between the preponderance of long and short range edges. Again *C*, given by

$$C = \frac{3z(1-\lambda)\lambda}{2(1+\lambda)(1+3\lambda)},$$

has a maximum, this time at $\lambda=(8^{1/2}-1)/7$ = 0.261… (if $\lambda$ is allowed to range this low); and *C* is zero at both extremes $\lambda=0,\ 1$ if z<2 .

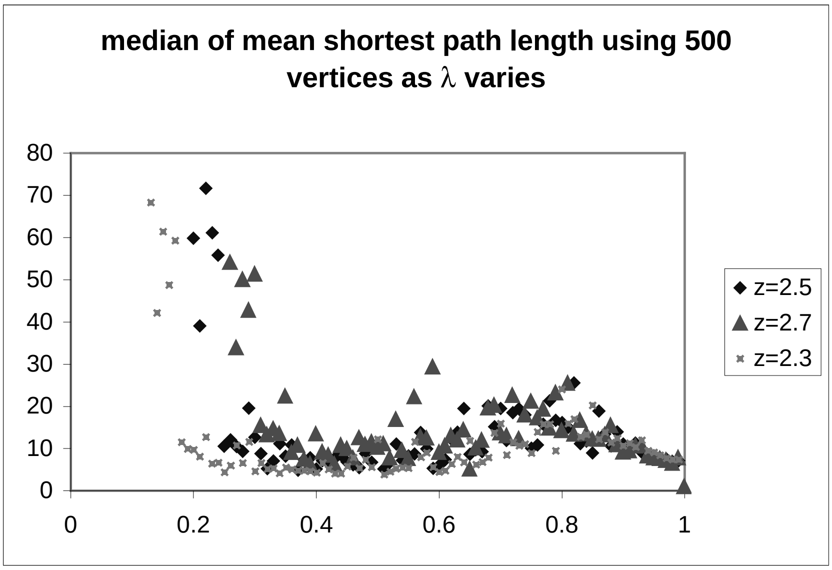



If we approximate such a graph stochastically on a large number of vertices we may contrast the theoretical curve for *C* with exact calculations. This is shown above for three fixed values for *z*.

It is advantageous in considering inverse problems to have a class of graphs, such as this, with an explicit algebraic formula for *C*, since, given a graph as a list of vertices and edges, we will wish to calibrate the global parameters based on matching global properties such as *C*.

The **small world** property, as defined by Watts and Strogatz [2,3], is a combination of the connectedness apparent within classical random graphs, *G(n,p)*, and the clustering behaviour, as measured by *C*. Above we see that, for *z* fixed, *C* is relatively at high at middle values for $\lambda$, falling linearly as $\lambda$ approaches unity to the value expected for a random graph (p=z/(n-1)).

Connectedness is measured by the median (over all vertices) of the mean shortest path lengths (from a vetex to all other vertices), see Watts [2]. As for other small world models we see this measure falls to its asymptotic value for random graphs (where $\lambda$=1) at much smaller values for $\lambda$. This is shown, for three values of *z*, in the graph below. Hence we have the small world effect at intermediate values of $\lambda$, say from shortly after the minimum possible value for $\lambda$, up to 0.9 or so.




Before leaving this section we must point out that some graphs of this type may have explicit generating functions. Recall that this is given by

$$G_0(x) = \prod_{\substack{k\neq 0 \\ -\infty}}^{\infty} ((1 - p_{0k}) + p_{0k}x) .$$

Then if we choose to set

$$p_{i,j} = f(|\ j - i\ |) = \frac{b}{(|\ i - j\ |\ \pi)^2},$$

for some $b$ fixed in $(0,\pi^2]$, then we have (see 1.431 Gradshteyn and Ryzhik)

$$G_0(x) = \prod_{k=1}^{\infty} (1 - \frac{b}{(k\pi)^2}(1-x))^2 = \frac{\sin^2 \sqrt{b(1-x)}}{b(1-x)}.$$

Hence it follows directly that $z=b/3$, $z_2 = 4b^2/45$, $P_o=\sin^2(b^{1/2})/b$, and much else besides.

**Large finite graphs**

In practice we will deal with graphs with a very large, but finite number, $n$, of vertices. There the assumptions used in the case of an infinite number of vertices will be violated due to “edge effects” at either end of the underlying orderring. Since the longer range edges are successively less likely to occur, the edges of the graphs are only seen within a kind of boundary layer (whose size is dependent upon the decay of $f(k)$). Within the interior the averages and sums used to calculate $z$, $z_2$ and $C$, etc., for a vertex will be valid approximations though, and if the graph is large these estimates will dominate in defining the expected behaviour for the graph as a whole. We can see this effect explicitly by considering the behaviour of $C$ where $\lambda$ varies towards unity. In the limit we must have $C=\alpha$ rather than $3\alpha/8$.

Of course this difference is not highly problematical if we have fixed $z$ as a parameter rather than $\alpha$, so that as $\lambda$ approaches unity all edges become equally probable, with probability $p = \alpha = z/(n-1)$, which is close to the zero value obtained in the infinite case (where $\alpha = z\,(1-\lambda)/2$).

Below we will wish to estimate values for $\alpha$ and $\lambda$ consistent (albeit for infinite graphs) with the values of $z$, and $C$ observed for actual finite graphs. Hence if $n$ is not large some care must be taken.

**The inverse problem**

Here we introduce an algorithm to take raw interaction data, estimate the global parameters, and use maximum likelihood modelling to produce an ordering of the vertices (a permutation of the vertices as originally given) from which the data is most likely to have been generated.

This is a two step process. First the graph parameters $\alpha$ and $\lambda$ are estimated from global graph properties (C and z, for example). Then we use a search algorithm to find an ordering of the vertices which maximizes the product of the odds over all edges which exist (within the new ordering).

To be more explicit: if *k(i)* denotes any reordering (permutation) of the original vertices (labelled by *i*), then the likelihood of the given data being generated in the ordering *k(i )* by the calibrated model is given by

$$Likelihood = \prod_{\text{edges } v_i v_j} p_{k(i),k(j)} \times \prod_{\text{No edges } v_i v_j} (1 - p_{k(i),k(j)}) \,.$$

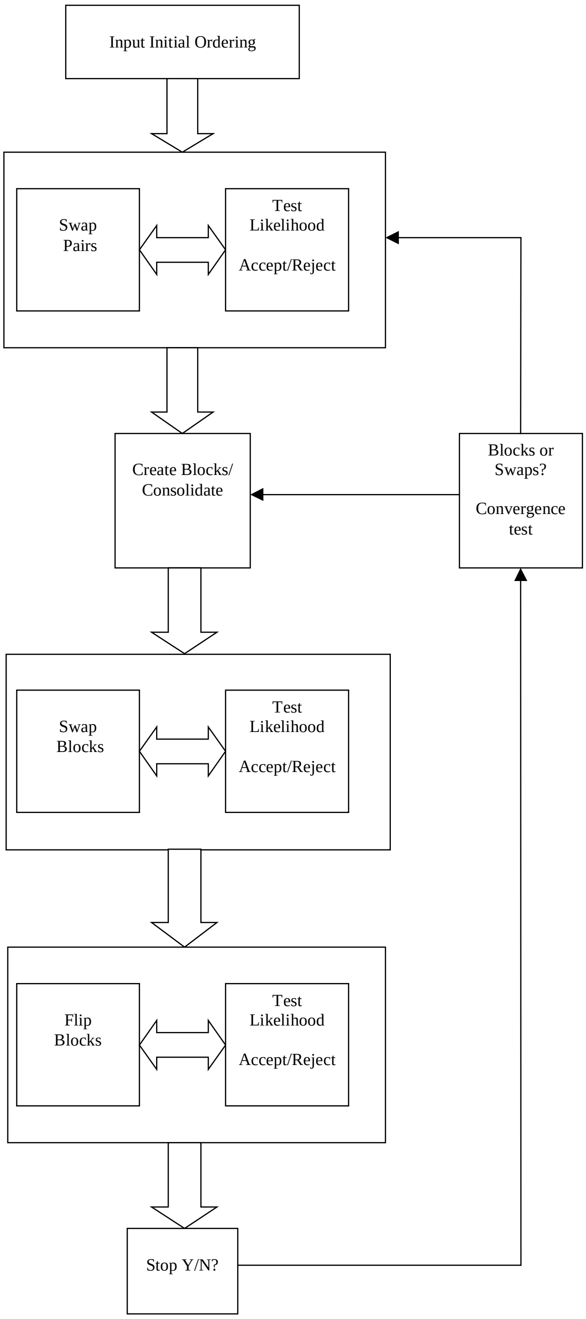
Here we use $p_{k(i),k(j)}$ to denote the probability that the edge between vertices *k(i)* to *k(j)* is present. In maximum likelihood modelling we wish to find an optimal reordering which maximises the likelihood of the actual observations (the data set) being generated by the model. We divide and multiply the above expression by the probability that each edge which actually exists, does not exits, and then factor the probability of producing the null graph (the graph on the same number of vertices with no edges present at all). Hence we wish to find *k* so as to maximise

$$Likelihood = \prod_{\text{edges } v_i v_j} \frac{p_{k(i),k(j)}}{1\text{-}p_{k(i),k(j)}} \times P(\text{NullGraph}) \,.$$

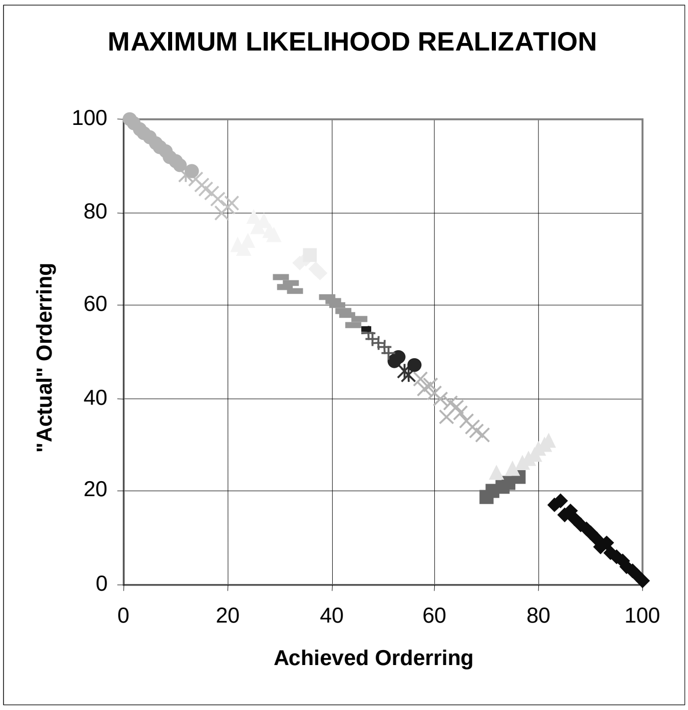
The term *P*(Null Graph) is the probability of generating no edges on N vertices for the given values of $\alpha$ and $\lambda$ and is a constant for all k, so plays no role (and consequently does not need to be calculated). This makes the maximum likelihood modelling efficient.

The above equation shows us how to trade orderings of the graph. If we move a vertex within the present orderring, then we change the distances associated with many edges: some get shorter, some get longer. The proposed reordering is more acceptable if the product of the odds increases.

This has led us to develop an algorithm to search for a suitable representation of the given graph. In essence we start from any given orderring and then sort it further by trading of the odds of the reordered edges appearing in the model (suitably calibrated via the global parameters). The algorithm applied to find permutations is a “double level“ algorithm which we have devised specifically for this purpose. At an individual vertex level we allow swaps between individual pairs of vertices: this tends to get local bocks of vertices, that is sequences of neighbouring vertices which are connected, together. Then, on a macroscopic level, we allow whole block swaps and block flips (reverses). In practice it is necessary to move up and down between these two levels of manipulation. The algorithm is summarised below




In the first example below we carried out a test on a 100-vertex graph. We first generated a test graph using the model, and then shuffled up the vertices to create a randomised ordering. In this form the model graph cannot describe the data since there is no relationship between relative proximity of the shuffled orderring and the likelihood of edges occurring. We then input the resulting graph (as a list of edges denoted by the shuffled ordering) into the search algorithm (swaps, block cycles and block flips). The model trades long and short range associations in permutations of the initial shuffled order. The result shows that we achieved an ordering almost correct (except for the overall sense of direction). However some long range associations are permissible (allowed by the model) whilst closely interrelated vertices are nor ordered closely together.

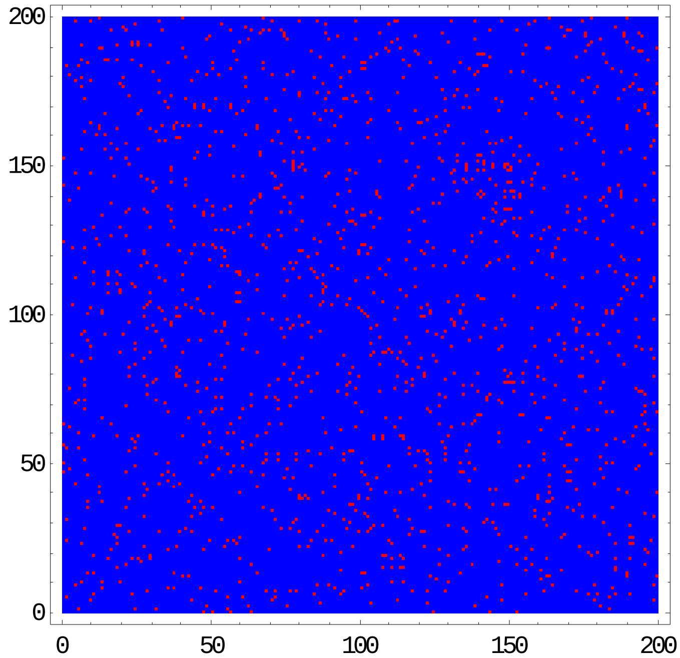


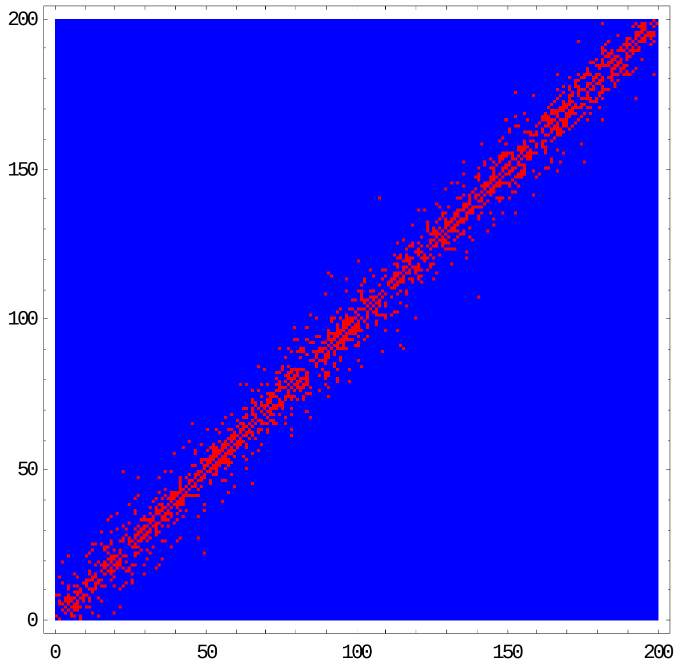
Over the page we show before and after plots of the adjacency matrices, depicting an edge between vertex i and vertex j as a red blob with in a symmetrical 200 by 200 matrix plot. After the optimally achieved reordering we see that most edges are now local (near the diagonal) with successively longer range edges becoming successively rare, as the model predicts.

One of the problems encountered within standard network analysis is the "all or nothing" nature of things like path length or connectivity, when the data a) contains erroneous associations (associations which shouldn’t be there) b) has missing associations.

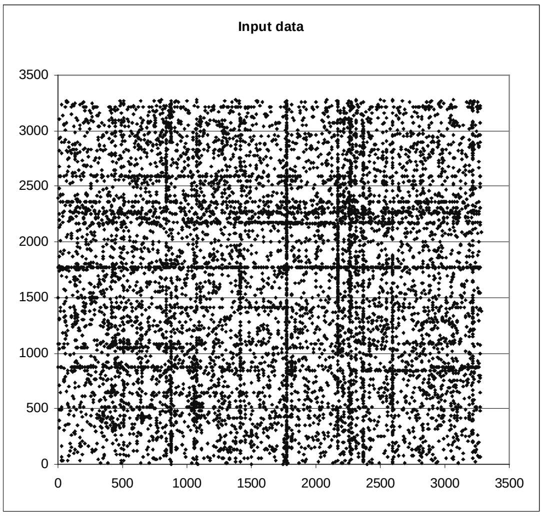
We have investigated both these problems. In our desire to construct a realisation of a given graph (a given set of edges) as a member of a class of graphs, we have shown that the realisation is robust to perturbations of both types of problem. For example if we take a numerically generated example case and delete or add a few associations then the maximum likelihood algorithm, which optimises the fit of the graph to the class, remains stable. In essence the metrics/distances we find from an inferred ordering of the disordered data are stable to a "small" number of such perturbations in the data.

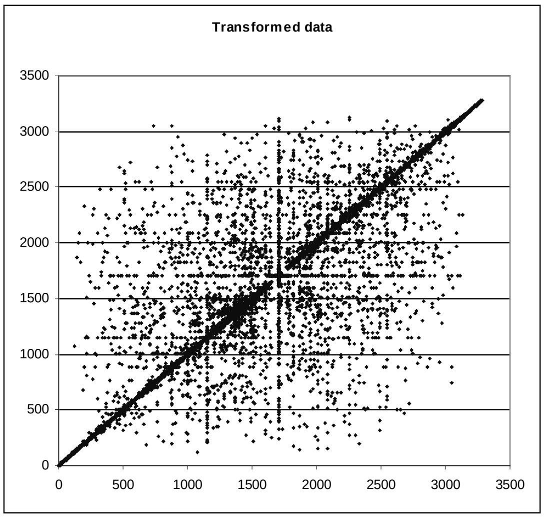
Actually, the example above, demonstrates this already, since we might have generated any other nearby graph and obtained a similar result.




Next we show an example drawn from bioinformatics [6] where protein-protein interactions have been observed for the yeast proteome [7,8,9] containing over 3000 proteins. Here we show the results of applying the inverse algorithm, with suitable parameters, to this large sparse graph (the vertices represent individual identified proteins). The result shows a classified protein graph and a greatly simplified structure. The relocation of proteins within the proteome is useful in that resultant near neighbours, in local cliques, may possess shared or complementary functional roles. This can be verified directly where proteins are annotated.


Input data
3500
3000
2500
2000
1500
1000
500
0
0 500 1000 1500 2000 2500 3000 3500


Transformed data
3500
3000
2500
2000
1500
1000
500
0
0 500 1000 1500 2000 2500 3000 3500

**Conclusions**

We have introduced a new class of range dependent random graphs which we have shown can possess small world characteristics, and can be described by simple global parameters. We have analysed their properties and indicated how an explicit formula can approximate the Watts-Strogatz clustering number when the graphs are very large.

We have devised and demonstrated a maximum likelihood algorithm that can realise a given graph as a member of our calibrated call of graphs. This optimal vertex ordering essentially resolves the inverse problem.

We have applied these ideas to tests data sets, for validation, and to also a large yeast proteome data set. The results are encouraging, indicating possible future analysis and applications in classifying large sparse graphs, with small world properties, and in realising them: the optimal ordering representing additional information conditional on the acceptance of the modelling concept. For bioinformatics the results are valuable in that they are robust to data errors and the optimal ordering may be exploited by building probabilistic models inferring co-functional roles of proteins or genes. This will be the direction of future work.

**References**

[1] B.Bollabas, Random Graphs, Academic press, New York,1995.
[2] D.J. Watts, Small Worlds, The dynamics of networks between Order and Randomness, Princeton University Press, 1999.
[3] D.J. Watts and S.H. Strogatz, Collective dynamics of small world networks, Nature,393,440-442,1998.
[4] M.E.J. Newman, Models of the small world: a review, at achieve: http://www.arxiv.org/cond-mat/ 0001118 v2 May 2000.
[5] M.E.J. Newman, S.H. Strogatz, D.J. Watts, Random graphs with arbitrary degree distribution and their applications, http://www.arxiv.org/cond-mat/ 0007235 July 2000.
[6] T.R. Hazbun, S. Fields, Networking proteins in yeast, PNAS, 98, No. 8, 4277-4278, April 2001.
[7] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, Y. Sakaki, A comprehensive two-hybrid analysis to explore the yeast protein interactome, PNAS, 98, No. 8, 4569-4574, April 2001.
[8] H. Jeong, S.P. Mason, A.-L. Barabesi, Z.N. Oltvai, Lethality and centrality in protein networks, Nature, 411,41-42,May 2001.
[9] H. Jeong, B. Tombor, R. Albert, Z.N. Oltvai, A.-L. Barabesi, The large-scale organisation of metabolic networks, Nature, 407, 651-654, Oct 2000.