Range-dependent random graphs and their application to modeling large small-world Proteome datasets

Peter Grindrod*
Numbercraft Limited, Magdalen Centre, The Oxford Science Park, Oxford OX4 4GA, United Kingdom
(Received 21 January 2002; published 10 December 2002)

In this paper we consider the problem of characterizing and modeling large-scale networks using classes of range-dependent graphs which possess appropriate small-world properties. The application we have in mind is to bioinformatics, where methods of rapid protein identification mean that such proteome datasets, listing various observed protein-protein associations, will become more and more prevalent. We introduce a class of range-dependent graphs, governed by a power law relating intervertex range to edge probability, which are amenable to analysis, and for which macroscopic graph parameters are given by explicit forms. We show how these may be employed in representing a given network using a maximum likelihood approach. This in turn annotates every given edge with its range, representing the tendency for such an association to be transitive. We apply this technique to published proteome data, and demonstrate that known protein associations are thus identified.

DOI: 10.1103/PhysRevE.66.066702 PACS number(s): 02.50.Ey, 87.10.+e

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 [2–4]. The goal 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 analyzed model for such a graph is the Watts-Strogatz graph, where a cyclic lattice (with all k near neighbors 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 behavior 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 subgraphs are properly related so that the final graph has a well-behaved vertex degree distribution. [This approach is analogous to that underlying fractal (self-affine) structures, obeying scaling laws over a range of different length scales—differing by orders of magnitude, if not actually from the infinitesimal to the infinite.] These are the subjects of this paper.

We show how such graphs with power-law probability can be defined and parametrized 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 from 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. Such types of graphs, describing long-range bond processes, have been introduced in percolation theory (see the discussion by Grimmett in Ref. [5], and the references therein), where conditions for the existence of infinite connected components are sought. Quite general graphs on infinite vertices where the mean range over all edges is finite, are known to contain no such component. The specific classes of graphs introduced here, where the relation between range and edge probability is given by a power scaling law (rather than a polynomial), and their small-world clustering properties, do not appear to have been considered though.

There are two recent reviews on the statistical mechanics and evolution of networks [6,7] which provide further background on the fast progress in related subjects, and underlying analytical methods.

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 modeled by, a suitably parametrized version of our model. Then we wish to order the vertices of the given graph in a way that it is most likely to have been generated. This yields extra information that 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 realize a given graph as a member of our class of graphs (suitably calibrated by global properties of the data). This method

*Email address: peterg@numbercraft.co.uk

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 that are missing in the data, and edges that 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 throughput devices mean 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 (intracellular 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 coexpression data from microarrays has used clustering and discrimination concepts, which are inherently ''many to one.'' Therefore, graph theoretic approaches for describing and modelling the structure of all gene-to-gene or protein-to-protein relationships offer a step forwards. Nodes (vertices) represent proteins (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 that inherit a simple notion of intervertex length scale, or range, by being embedded in a possibly incomplete one-dimensional lattice. Generalizations to a cyclic lattice are immediate. The motivation for this is to define a suitable class of stochastic graphs which (1) may show the small-world characteristics of ''localized'' clustering coupled with longer range connectivity; (2) are amenable to analysis, and characterized by simple global parameters; (3) have a hierarchy of edges on different scales (ranges), for which the successively ''longer range'' edges are less and less likely to exist; and (4) 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 $k=\ldots,-2,-1,0,1,2,\ldots$), where the probability of an edge connecting vertex v_i to vertex v_j is given by a function of 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 $|i-j|$. Note that the probabilistic structure is invariant to translation and reflection with respect to the underlying vertex ordering.

We introduce the generating function $G_0(x)$ [8] 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 $\alpha=1$ then neighbors are certainly connected, by edges with range 1, and the graph contains a Hamiltonian path connecting all immediate neighbors, regardless of λ. If $\alpha<1$, 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 $k=1,2,\ldots$, 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 graphs global characteristics.

Consider a given vertex, say v_0, then the expected number of neighbors 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 $\alpha=1$, then both $G_0(0)=0$, and $G_0'(0)=0$, so that $P_0=P_1=0$: every vertex has at least two neighbors.

We also have

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

Hence

$$G_0''(1)=\frac{z^2}{2}\left\{\frac{1+3\lambda}{1+\lambda}\right\},$$

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

Now, z_2 behaving like the square of z is the sort of behavior seen in random graphs (normal Poisson random graphs for instance—however these do not show the clustering behavior discussed below). The Watts-Strogatz clustering number, C, [2,3] 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_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 centered 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 $\{v_i,v_0,v_j\}$ centered at v_0. The numerator gives the expected number of triangles $\{v_i,v_0,v_j\}$ centered at v_0.

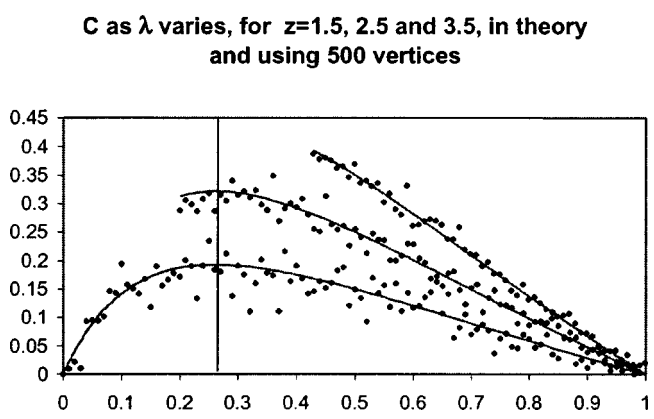

FIG. 1. Cluster coefficient estimates.

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 λ tends to 1, C tends to $\frac{3}{8}$. This is because the graph approaches a completely connected graph in a nonuniform way—there are always vertices far enough away to make adjacency improbable.

C is not a monotonic function of λ. For fixed α there is a local maximum at $\lambda=3^{-1/2}$. Hence, there is a kind of ''optimum'' clustering connectivity at this maximum. If λ increases further, the probability of having a long range neighbor that is not also connected to more localized neighbors increases, hence C decreases.

If we set $\alpha=z(1-\lambda)/2$, then with the average vertex degree, z, fixed, λ may vary in $(\max\{0,1-2/z\},1]$. Hence with this parameterization λ 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 λ 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 in Fig. 1 for three fixed values for z.

It is advantageous when 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

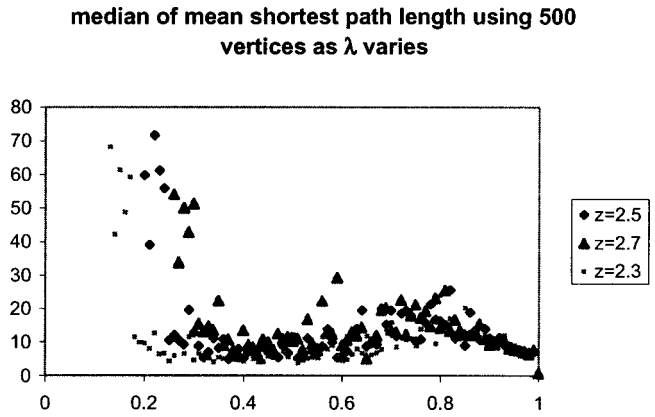

FIG. 2. Estimates of the median of the mean shortest path lengths.

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 behavior, as measured by C. Above we see that, for z fixed, C is relatively at high at middle values for λ, falling linearly as λ 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 vertex to all other vertices), see Watts [2]. As for other small-world models, we see that this measure falls to its asymptotic value for random graphs (where $\lambda=1$) at much smaller values for λ. This is shown, for three values of z, in Fig. 2. Hence we have the small-world effect at intermediate values of λ, say from shortly after the minimum possible value for λ, up to 0.9 or so.

Before concluding 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

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

Hence it follows directly that $z=b/3$, $z_2=4b^2/45$, $P_o=\sin^2(b^{1/2})/b$, and much else. Grimmet also considers this type of graph, defined by asymptotic polynomial behavior [5].

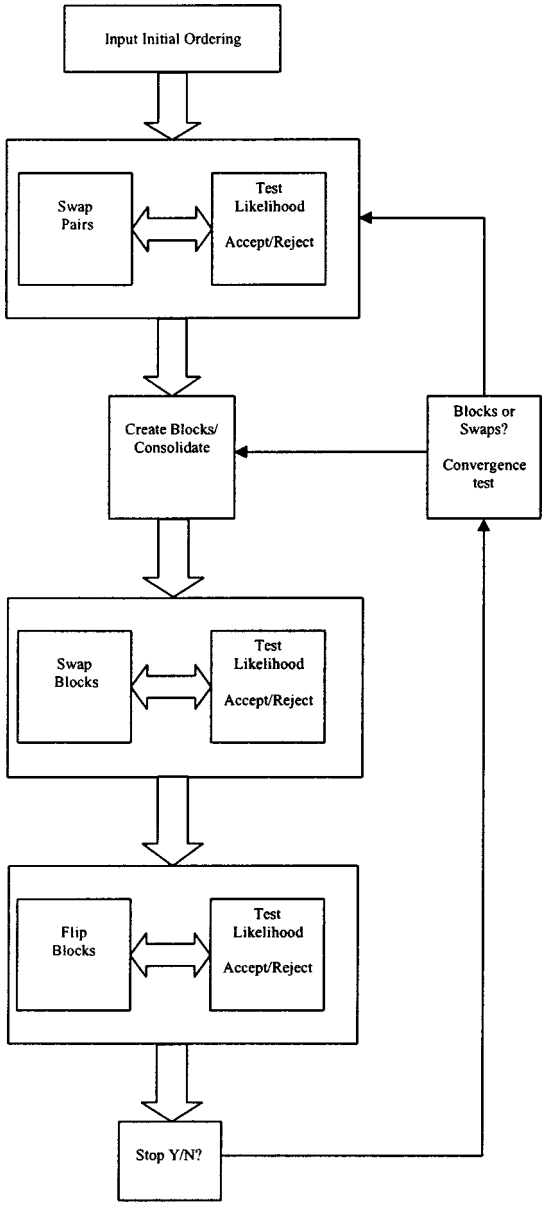

FIG. 3. Search algorithm.

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 ordering. 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 on the decay of $f(k)$]. Within, the averages and sums used to calculate z, z_2, C, etc., for a vertex will be valid approximations though; and if the graph is large, these estimates will dominate in defining the expected behavior of the graph as a whole. We can see this effect explicitly by considering the behavior of C where λ varies towards unity. In the limit we must have $C=\alpha$ rather than $3\alpha/8$.

Of course this difference is not highly problematic if we fix z as a parameter rather than α, so that as λ approaches

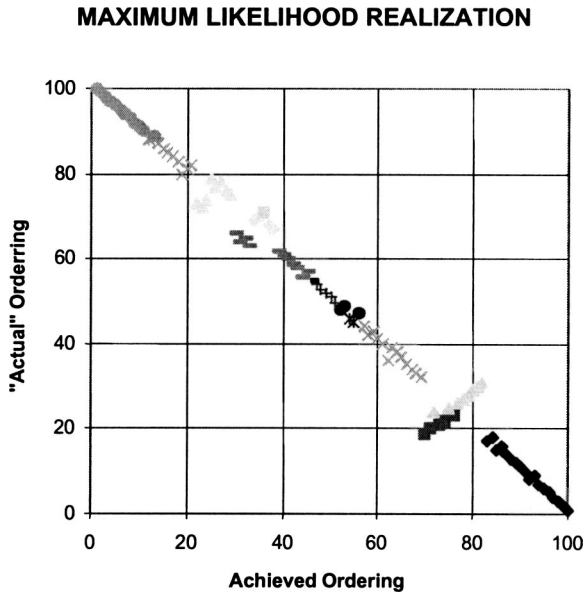

FIG. 4. Verification test: actual vs achieved orderings.

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 α and λ 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 modeling 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 α and λ are estimated from global graph properties (C and z, for example). Then we use a search algorithm to find an ordering of the vertices that maximizes the product of the odds over all edges that exist (within the new ordering).

To be more explicit, if $k(i)$ denotes any reordering (permutation) of the original vertices (labeled by i), then the likelihood $\mathcal{L}$ of the given data being generated in the ordering $k(i)$ by the calibrated model is given by

$$\mathcal{L}=\prod_{\text{edges } \nu_i\nu_j} p_{k(i),k(j)} \prod_{\text{No edges } \nu_i\nu_j} (1-p_{k(i),k(j)}).$$

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 modeling, we wish to find an optimal reordering that maximizes the likelihood of the actual observations (the dataset) being generated by the model. We divide and multiply the above expression by the probability that each edge which actually exists, does not exist, 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 maximize

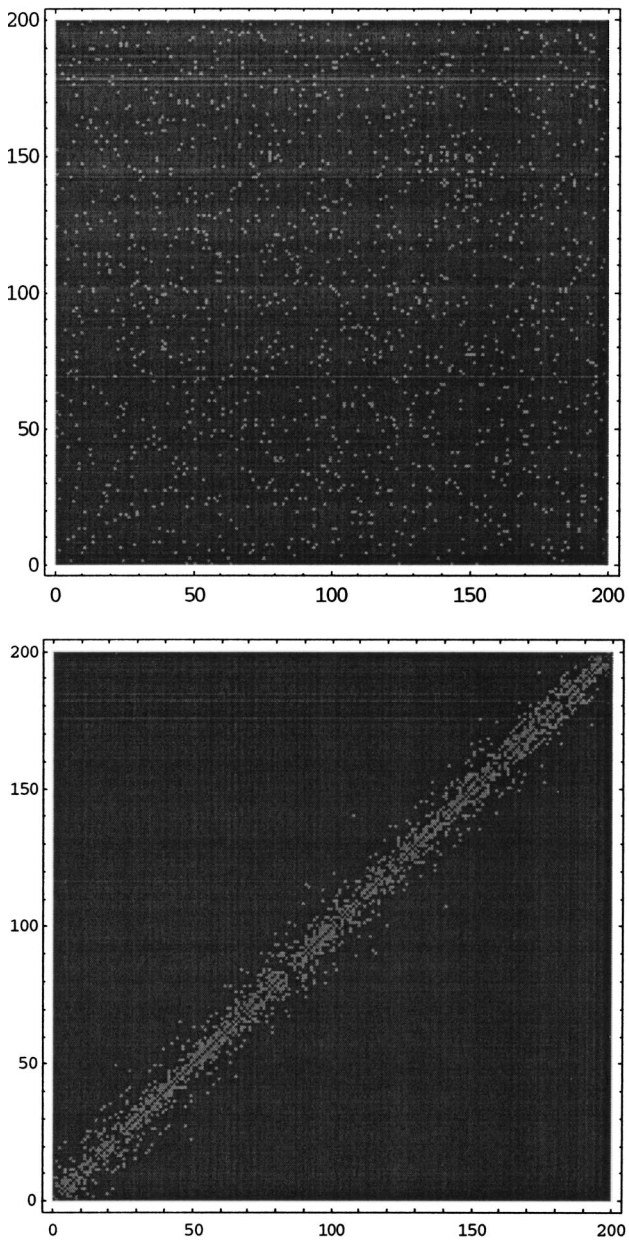

FIG. 5. Plots of the adjacency matrices before and after reordering.

$$\mathcal{L}=\prod_{\text{edges } \nu_i\nu_j} \frac{p_{k(i),k(j)}}{1-p_{k(i),k(j)}} P(\varnothing).$$

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

The above equation shows us how to trade orderings of the graph. If we move a vertex within the present ordering, 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

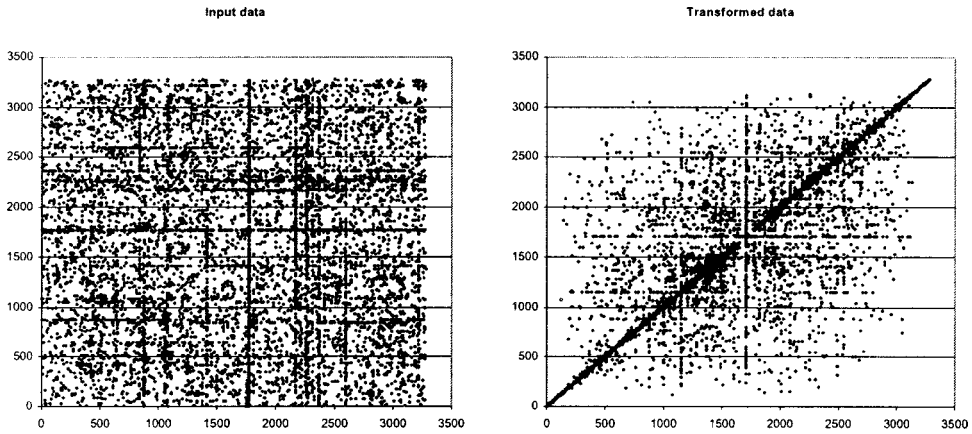

FIG. 6. Results of applying the inverse algorithm.

start from any given ordering 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 that 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 blocks of vertices, that is sequences of neighboring vertices that 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 summarized in Fig. 3.

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 randomized ordering. In this form, the model graph cannot describe the data since there is no relationship between relative proximity of the shuffled ordering 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 shown in Fig. 4 shows that we achieved an almost correct ordering (except for the overall sense of direction). However, some long range associations are permissible (allowed by the model) whilst closely interrelated vertices are not ordered closely together. In Fig. 5, we show before and after plots of the adjacency matrices, depicting an edge between vertex i and vertex j as a blob within a symmetrical 200×200 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 such as path length or connectivity, when the data (a) contains erroneous associations (associations which should not be there) (b) has missing associations.

We have investigated both these problems. In our effort to construct a realization of a given graph (a given set of edges) as a member of a class of graphs, we have shown that the realization is robust to perturbations of both types of problems. For example, if we take a numerically generated case and delete or add a few associations, then the maximum likelihood algorithm, which optimizes 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 above example, 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 [9] where protein-protein interactions have been observed for the yeast proteome [10–12] containing over 3000 proteins. In Fig. 6 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 neighbors, in local cliques, may possess shared or complementary functional roles. This can be verified directly where proteins are annotated.

CONCLUSIONS

We have introduced a 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 analyzed 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 realize a given graph as a member of our calibrated class of graphs. This optimal vertex ordering essentially resolves the inverse problem.

We have applied these ideas to test datasets, for validation, and also to a large yeast proteome dataset. The results are encouraging, indicating possible future analysis and applications in classifying large sparse graphs, with small-world properties, and in realizing them: the optimal ordering representing additional information depending on the acceptance of the modeling 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 cofunctional roles of proteins or genes. This will be the direction of future work.

[1] B. Bollabas, *Random Graphs* (Academic, New York, 1995).
[2] D. J. Watts, *Small Worlds* (Princeton University Press, Princeton, NJ, 1999).
[3] D. J. Watts and S. H. Strogatz, Nature (London) **393**, 440 (1998).
[4] M. E. J. Newman, e-print http://www.arxiv.org/cond-mat/0001118 v2
[5] G. Grimmett, *Percolation*, 2nd ed. (Springer, Berlin, 1999).
[6] S. N. Dorogovtsev and J. F. F. Mendes, Adv. Phys. (to be published); e-print cond-mat/0106144.
[7] Réka Albert and Albert-László Barabási, Rev. Mod. Phys. **74**, 47 (2002).
[8] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, e-print http://www.arxiv.org/cond-mat/0007235
[9] T. R. Hazbun and S. Fields, Proc. Natl. Acad. Sci. U.S.A. **98**, 4277 (2001).
[10] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki, Proc. Natl. Acad. Sci. U.S.A. **98**, 4569 (2001).
[11] H. Jeong, S. P. Mason, A.-L. Barabesi, and Z. N. Oltvai, Nature (London) **411**, 41 (2001).
[12] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabesi, Nature (London) **407**, 651 (2000).