IMA Journal of Numerical Analysis (2010) **30**, 195−207
doi:10.1093/imanum/drp047
Advance Access publication on December 30, 2009

Periodic reordering

PETER GRINDROD
Department of Mathematics and Centre for Advanced Computing and Emerging Technologies, University of Reading, Reading RG6 6AX, UK

AND

DESMOND J. HIGHAM† AND GABRIELA KALNA
Department of Mathematics and Statistics, University of Strathclyde, Glasgow G1 1XH, UK

[Received on 28 April 2008; revised on 9 November 2009]

Dedicated to the memory of A. R. Mitchell, 1921–2007.

For many networks in nature, science and technology, it is possible to order the nodes so that most links are short-range, connecting near-neighbours, and relatively few long-range links, or shortcuts, are present. Given a network as a set of observed links (interactions), the task of finding an ordering of the nodes that reveals such a range-dependent structure is closely related to some sparse matrix reordering problems arising in scientific computation. The spectral, or Fiedler vector, approach for sparse matrix reordering has successfully been applied to biological data sets, revealing useful structures and subpatterns. In this work we argue that a periodic analogue of the standard reordering task is also highly relevant. Here, rather than encouraging nonzeros only to lie close to the diagonal of a suitably ordered adjacency matrix, we also allow them to inhabit the off-diagonal corners. Indeed, for the classic small-world model of Watts & Strogatz (1998, Collective dynamics of 'small-world' networks. *Nature*, **393**, 440–442) this type of periodic structure is inherent. We therefore devise and test a new spectral algorithm for periodic reordering. By generalizing the range-dependent random graph class of Grindrod (2002, Range-dependent random graphs and their application to modeling large small-world proteome datasets. *Phys. Rev. E*, **66**, 066702-1–066702-7) to the periodic case, we can also construct a computable likelihood ratio that suggests whether a given network is inherently linear or periodic. Tests on synthetic data show that the new algorithm can detect periodic structure, even in the presence of noise. Further experiments on real biological data sets then show that some networks are better regarded as periodic than linear. Hence, we find both qualitative (reordered networks plots) and quantitative (likelihood ratios) evidence of periodicity in biological networks.

1. Background

Large, sparse networks arise naturally when we describe the interconnectedness of components in complex systems (Strogatz, 2001; Newman, 2003; Alon, 2006). The need to extract useful information creates challenging computational problems that, at least in part, overlap with sparse linear algebra tasks dealt with by numerical analysts. In this work we look at a matrix reordering problem that arises naturally from recent work in network modelling and computational biology. The reordering comes with a twist—a periodic analogue of the more usual 'envelope reduction' or 'two-sum minimization' is required.

[†] Corresponding author. Email: aas96106@maths.strath.ac.uk

The presentation is organized as follows. In Section 2 we outline some recent random graph models that motivate the inverse problem. In Section 3 we give a brief overview of the use of spectral methods for graph reordering, based on the graph Laplacian. We then derive a spectral algorithm for the periodic reordering problem and illustrate its use on specially constructed test data. In Section 4 we show that, under the hypothesis that the data come from a random network class with range-dependent edge probabilities, it is possible to compare the likelihoods of linear and periodic structure. In Section 5 we apply the algorithm to biological network data and, in some cases, find evidence of periodic structure.

2. Network models

Classical random graph theory studies models where either (a) an edge is placed between a pair of nodes with some fixed, independent, probability or (b) a graph with a specified number of nodes and edges is chosen uniformly at random from the collection of all such graphs (Erdös & Rényi, 1959; Gilbert, 1959). Strogatz (2001) makes the point that networks in nature and technology neither look like classical random graphs nor look like regular lattices. Watts & Strogatz (1998) proposed a new model that aimed to capture this 'between order and disorder' appearance. Their model begins with a periodic k-nearest neighbour ring and proceeds by *rewiring*. Given some fixed probability, ρ say, we consider each edge in turn, and with probability ρ we exchange (rewire) one of its end nodes with a node chosen uniformly across the network. The average degree thus remains constant.

In Newman *et al.* (2000), instead of rewiring, the authors added *shortcuts* to create a very similar effect. For each node in turn, with some probability ρ we insert a new edge that connects it to another node chosen uniformly across the network. This construction has the benefit of guaranteeing to maintain connectivity, though it increases the average degree.

Watts and Strogatz coined the term *small-world network* to describe the seemingly unlikely combination of small typical pathlength (randomly chosen nodes can be connected by small chains of edges) and high clustering coefficient (neighbours of neighbours tend to be neighbours). They showed via simulations that the rewired periodic ring has the small-world property for suitable values of ρ, and also showed that many real-life networks are small worlds. Hence, the small-world model goes some way to capturing an essential feature of complex networks.

Grindrod (2002) proposed a variation of the Watts–Strogatz and Watts–Newman–Moore models called range-dependent random graphs (RDRGs; Higham, 2005). Here, shortcuts arise with a probability that depends on the lattice distance between nodes, that is, the *range*. Grindrod argued that this type of connectivity can be used to describe interactions between proteins. The model uses a linear, rather than periodic, node ordering: this assumption was largely pragmatic, anticipating that the number of nodes would be very large in applications.

DEFINITION 2.1 For a given decay function, f, that maps from $\{1, 2, \ldots, N-1\}$ to $[0, 1]$, the RDRG model generates an edge between nodes i and j with independent probability $f(|j-i|)$.

The case of geometric decay, where $f(k) = \alpha\lambda^{k-1}$ for constants $\alpha, \lambda \in [0, 1]$, allows for explicit analysis (employing a generating function method) to calculate the clustering coefficient and other macro properties of the network (Grindrod, 2002). Here we will focus on the case where $\alpha = \lambda$ and consider geometric decay $f(k) = \lambda^k$. An RDRG is illustrated in the upper left picture of Fig. 1.

Given the inherent periodicity in the influential Watts–Strogatz model, it is natural to define a periodic version of the RDRG model in the following manner.

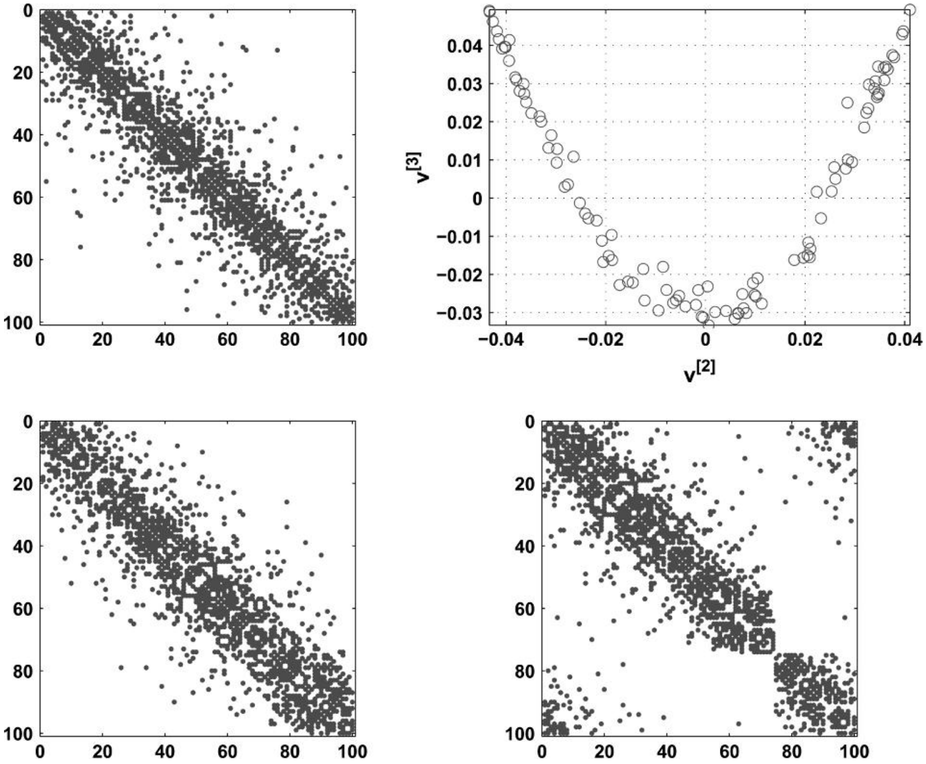

FIG. 1. Linear (RDRG) with $N = 100$ and $\lambda = 0.9$ (upper left) and its linear (lower left) and periodic (lower right) reorderings. Scatter plots of $v^{[2]}$ and $v^{[3]}$ (upper right).

DEFINITION 2.2 For a given decay function, f, that maps from $\{1, 2, \ldots, N-1\}$ to $[0, 1]$, the periodic RDRG (pRDRG) model generates an edge between nodes i and j with independent probability $f(\min\{|j-i|, N-|j-i|\})$.

Here we have defined a pRDRG by using periodic lattice distance, or periodic range, in the decay function, so, for example, nodes 1 and N are a unit distance apart; in the RDRG their separation distance would be $N-1$. The upper left picture in Fig. 2 illustrates a pRDRG.

We will show that pRDRGs not only form a useful class of test networks but also can be used to motivate a measure of periodicity.

3. Spectral reordering

In addition to proposing a model, Grindrod (2002) pointed out that there is, in practice, the need to solve a related inverse problem.

In situations where edges represent observed interactions, they are typically presented in some contrived or an arbitrary order. So given such a data set, it is of interest to look for a new node ordering that reveals a ‘regular lattice plus short cuts’ pattern. (This concept is illustrated on real biological data in Section 5.) This locates (near) cliques close together in the embedded lattice, allowing for some long-range edges. The resultant ordering and the inferred interaction ‘ranges’ provide insight resulting directly from the imposition of the RDRG structure on the data.

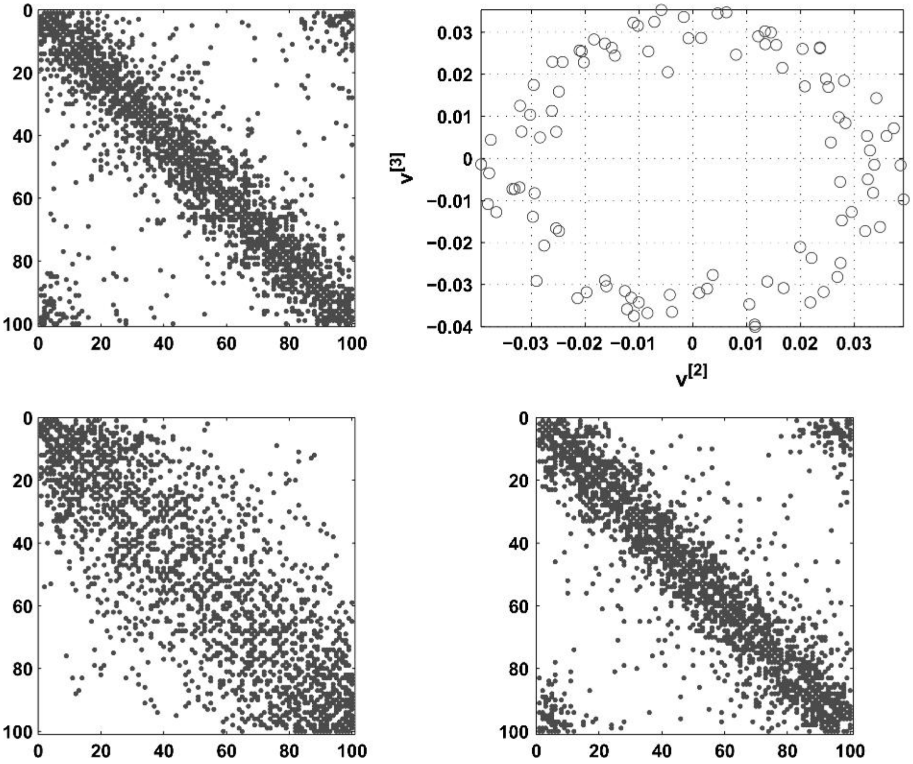

FIG. 2. Periodic (pRDRG), with $N = 100$ and $\lambda = 0.9$ (upper left) and its linear (lower left) and periodic (lower right) reorderings. Scatter plots of $\upsilon^{[2]}$ and $\upsilon^{[3]}$ (upper right).

To achieve this in the case of linear structure, Grindrod proposed a discrete reordering technique that attempted to optimize a log-likelihood function (that given in (4.4) below), essentially tackling a discrete optimization problem by genetic search. Higham (2003) showed that existing spectral reordering algorithms can be much quicker and more effective. We note that very similar aims arise in many other application areas, including pattern recognition (Shi & Malik, 2000), data mining (Eldén, 2007), high performance computing (Van Driessche & Roose, 1995) and sparse matrix computations (Duff *et al.*, 1986; Hu & Scott, 2003). In this work our aims are

1. to develop a spectral algorithm that reveals ‘regular lattice plus short cuts’ in the case where the underlying regular lattice has a periodic, rather than linear, structure and
2. to devise a computational test that determines whether a network is inherently more linear or periodic.

Suppose that $A = (a_{ij}) \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix for an unweighted, undirected graph with N nodes; so $a_{ij} = a_{ji} = 1$ if nodes i and j share an edge and $a_{ij} = a_{ji} = 0$ otherwise. A spectral reordering approach can be motivated by the idea of finding a permutation vector p (a vector containing each integer from 1 to N) so as to minimize the two-sum $\sum_{i=1}^{N} \sum_{j=1}^{N} (p_i - p_j)^2 a_{ij}$ (Barnard *et al.*, 1995; Higham, 2003, 2005; Spence *et al.*, 2007; Strang, 2008; Van Driessche & Roose, 1995). Here, we must seek p so that the edges tend to arise between nodes that are close in this new ordering.

In matrix terms we require nonzeros to lie near the diagonal in the reordered adjacency matrix. This discrete optimization problem is computationally intractable for large networks, but by relaxing to an optimization over real-valued vectors $p \in \mathbb{R}^N$ and imposing suitable constraints, we obtain a quadratic positive semidefinite problem that can be solved with an eigenvector. We could look for a periodic version of the two-sum, such as $\sum_{i=1}^{N}\sum_{j=1}^{N}(\min(|p_i - p_j|, N - |p_i - p_j|))^2 a_{ij}$. Minimizing this quantity would encourage nonzeros to lie either near the diagonal or close to the off-diagonal corners. However, the relaxed version is no longer in the form of a tractable quadratic variational problem. Instead we will look for motivation from the Watts–Strogatz model (Watts & Strogatz, 1998), whose k-nearest neighbour ring can be regarded as a one-dimensional structure embedded into two dimensions. We will therefore look for a projection of the nodes into $\mathbb{R}^2$ rather than $\mathbb{R}^1$ and then infer a one-dimensional ordering from the angular polar coordinate.

Spectral projection of the nodes into a low-dimensional space is itself a well-studied problem, with many algorithmic variants (Alpert & Yao, 1995; Van Driessche & Roose, 1995; Shi & Malik, 2000; Eldén, 2007; Skillicorn, 2007; Kalna *et al.*, 2008; Strang, 2008). Here we outline an approach based on the *normalized Laplacian* that we have found to be useful. For more details the reference Kalna *et al.* (2008) covers projection into more than one dimension and Higham *et al.* (2007) looks at unnormalized versus normalized Laplacians. Our starting point is to consider mapping the kth node into position $(x_k, y_k)^{\mathrm{T}} \in \mathbb{R}^2$ by solving the minimization problem

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N} \left\| \begin{pmatrix} x_i \\ y_i \end{pmatrix} - \begin{pmatrix} x_j \\ y_j \end{pmatrix} \right\|_2^2 a_{ij},$$

where $\|\cdot\|$ denotes the Euclidean vector norm. Here we are attempting to place nodes close together if they are connected by an edge. Let $x = (x_1, \ldots, x_N)^{\mathrm{T}}$ and $y = (y_1, \ldots, y_N)^{\mathrm{T}}$. Then our expression may be rewritten as

$$\min(x^{\mathrm{T}}(D - A)x + y^{\mathrm{T}}(D - A)y), \tag{3.1}$$

where D is the $N \times N$ diagonal matrix, $\mathrm{diag}(d_1, \ldots, d_N)$, containing the vertex degrees $d_i = \sum_{j=1}^{N} a_{ij}$. We let $D^{\frac{1}{2}}$ denote the corresponding half power of D: $\mathrm{diag}\left(d_1^{\frac{1}{2}}, \ldots, d_N^{\frac{1}{2}}\right)$. We also set $\mathbf{1} \in \mathbb{R}^N$ to be the vector with each component equal to one.

To avoid trivial solutions and redundancy we must add some constraints. First we must normalize the vectors x and y to keep them away from the origin. We impose

$$x^{\mathrm{T}}Dx = 1 \quad \text{and} \quad y^{\mathrm{T}}Dy = 1. \tag{3.2}$$

Here scaling each component by the corresponding node degree has the effect of down-playing the influence of highly connected nodes. Second we use

$$\mathbf{1}^{\mathrm{T}}D^{\frac{1}{2}}x = 0 \quad \text{and} \quad \mathbf{1}^{\mathrm{T}}D^{\frac{1}{2}}y = 0 \tag{3.3}$$

to ensure that the nodes are well spread, with the $\sqrt{d_i}$ scaling forcing relatively well-connected nodes to lie closer the origin.

It follows from standard linear algebra arguments, see, for example, Kalna *et al.* (2008), that (3.1) with (3.2) and (3.3) has solution given by $x = D^{\frac{1}{2}}v^{[2]}$ and $y = D^{\frac{1}{2}}v^{[3]}$, where the normalized Laplacian $D^{-\frac{1}{2}}(D - A)D^{-\frac{1}{2}}$ has eigenvalues $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_N$ with corresponding eigenvectors

$v^{[1]}, v^{[2]}, \ldots, v^{[N]}$. By construction, $\lambda_1 = 0$ and $v^{[1]} = D^{\frac{1}{2}}\mathbf{1}/\|D^{\frac{1}{2}}\mathbf{1}\|$. The eigenvalues are bounded above by 2 and $\lambda_2 > 0$ if and only if the underlying network is connected (Van Driessche & Roose, 1995).

We may therefore summarize our new algorithm for computing a permutation vector p that gives a periodic reordering as follows.

Periodic Reordering Algorithm

1. Compute a subdominant eigenvector pair $x := v^{[2]}$ and $y := v^{[3]}$ for the normalized Laplacian $D^{-\frac{1}{2}}(D - A)D^{-\frac{1}{2}}$.
2. Let $\theta_i = \tan^{-1}(y_i/x_i)$.
3. Construct a permutation vector p according to $p_i \leqslant p_j \Longleftrightarrow \theta_i \leqslant \theta_j$.

For comparison a corresponding linear version (Van Driessche & Roose, 1995; Shi & Malik, 2000; Higham *et al.*, 2007; Skillicorn, 2007; Strang, 2008) could be written:

Linear Reordering Algorithm

1. Compute a subdominant eigenvector $x := v^{[2]}$.
2. Construct a permutation vector p according to $p_i \leqslant p_j \Longleftrightarrow x_i \leqslant x_j$.

These algorithms are illustrated in Figs 1 and 2. The upper left picture in Fig. 1 shows an RDRG with $N = 100$ and $\lambda = 0.9$. The upper right picture scatter plots the components of $v^{[2]}$ and $v^{[3]}$. It is clear that the normalized Fiedler vector, $v^{[2]}$, does a good job of uncovering the linear ordering and $v^{[3]}$ can add nothing further. The lower left picture shows the matrix reordered according to the linear reordering algorithm, and the linear range-dependent structure is apparent. The lower right picture shows the result of the periodic reordering algorithm. In this case the algorithm has encouraged some nonzeros into the off-diagonal corners, but we see an unnatural break in the node density as we look down the diagonal.

We emphasize that in practice we would not expect to be given the matrix with the 'correct' ordering shown in the upper left picture. Instead, the nodes would arrive in some arbitrary order (Grindrod, 2002; Higham, 2003), and our task is to find the hidden structure. However, $v^{[2]}$ and $v^{[3]}$ are invariant under reordering (which, of course, corresponds to a similarity transformation), and hence the algorithms would perform exactly the same way if we started with any other node order.

In Fig. 2 we change to a pRDRG. In this case it is clear that both $v^{[2]}$ and $v^{[3]}$ carry useful reordering information. The linear algorithm is forced to increase the spread of nonzeros, whereas the periodic algorithm packs them tightly along the diagonal or in the off-diagonal corners.

4. Likelihood ratio

In Figs 1 and 2 it is visually obvious whether the graphs are inherently linear or periodic and whether one algorithm is more appropriate than the other. For real networks, of course, the issue will not be so clear cut. The idea in this section is to develop a test that gives a quantitative answer to the linear versus periodic question. Such inference issues require assumptions to be made, either implicitly or explicitly (Sivia, 2006), and we will start by assuming that the network comes from either one of the RDRG or pRDRG classes, each with a geometric decay function. We note that Grindrod (2002) used the RDRG model in order to define an objective function that could be maximized over all possible orderings and to find the most likely (linear) ordering under the hypothesis that the data come from that class. In our case the orderings arise from the two algorithms, corresponding to alternative hypotheses, in Section 3, and we compare

1. the likelihood of the linear ordering given that the data came from the RDRG class with geometric decay and
2. the likelihood of the periodic ordering given that the data came from the pRDRG class with geometric decay.

The first step is to fit the geometric decay rate, λ. We do this by matching the total number of edges in the given network to the expected number of edges arising in the RDRG and pRDRG models. In the RDRG case, the expected number of edges is $\sum\sum_{j>i} \lambda^{j-i}$, which has the analytic form

$$\frac{N\lambda}{1-\lambda} - \frac{\lambda(1-\lambda^N)}{(1-\lambda)^2}. \tag{4.1}$$

In the pRDRG case, the expected number of edges, $\sum\sum_{j>i} \lambda^{\min(j-i,N-j-i)}$, has the form

$$\frac{N\lambda}{1-\lambda} - \frac{N\lambda^{(N+1)/2}}{1-\lambda} \tag{4.2}$$

when N is odd and

$$\frac{N\lambda}{1-\lambda} - \frac{1+\lambda}{1-\lambda}\frac{N}{2}\lambda^{N/2} \tag{4.3}$$

when N is even. In each case a monotonically increasing scalar function in λ must be matched to the given edge count, so it is a simple numerical task to produce the values λ_{lin} and λ_{per} for the linear and periodic models, respectively.

Then for any reordering $i \mapsto p_i$, the likelihood of this network arising for the RDRG model is

$$\mathcal{L}_{\text{lin}}(p) := \prod_{\text{edge } p_i \leftrightarrow p_j} \lambda_{\text{lin}}^{|p_i-p_j|} \prod_{\text{no edge } p_i \leftrightarrow p_j} (1-\lambda_{\text{lin}}^{|p_i-p_j|}). \tag{4.4}$$

Similarly, for any reordering $i \mapsto p_i$, the likelihood of this network arising for the pRDRG model is

$$\mathcal{L}_{\text{per}}(p) := \prod_{\text{edge } p_i \leftrightarrow p_j} \lambda_{\text{per}}^{\min(|p_i-p_j|,N-|p_i-p_j|)} \prod_{\text{no edge } p_i \leftrightarrow p_j} (1-\lambda_{\text{per}}^{\min(|p_i-p_j|,N-|p_i-p_j|)}). \tag{4.5}$$

Effectively, the algorithms from Section 3 select suitable reorderings that are close to maximizing $\mathcal{L}_{\text{lin}}(p)$ and $\mathcal{L}_{\text{per}}(p)$ independently. Letting p_{lin} and p_{per} denote the ordering arising from those linear and periodic algorithms, respectively, the *log-likelihood ratio*, L, is defined as

$$L = \frac{2}{N(N-1)} \log\left(\frac{\mathcal{L}_{\text{lin}}(p_{\text{lin}})}{\mathcal{L}_{\text{per}}(p_{\text{per}})}\right), \tag{4.6}$$

with a positive ratio indicating that the network is more likely to be linear and a negative ratio indicating the opposite. Note that we normalize by the term $N(N-1)/2$, representing the number of possible edges, which corresponds to the number of factors within both (4.4) and (4.5): this allows us to contrast results for different sized data sets (if we double N then we roughly quadruple the number of terms in the sum that forms the log-likelihood ratio).

In Figs 1 and 2 we generated RDRG and pRDRG instances with $N = 100$ and $\lambda = 0.9$. In the RDRG case we found $\lambda_{\text{lin}} = 0.9004$ and $\lambda_{\text{per}} = 0.8908$ from (4.1) and (4.2), respectively. Since λ_{lin} is the closer to $\lambda = 0.9$ and the likelihood ratio $L = 1.75 \times 10^{-2}$ is positive, we conclude that the network is more likely to be linear. In the pRDRG case $\lambda_{\text{lin}} = 0.9091$ and $\lambda_{\text{per}} = 0.8994$. Here λ_{per} is closest and the negative likelihood ratio of $L = -1.37 \times 10^{-1}$ supports the hypothesis that the network is more likely to be periodic.

To test the likelihood ratio further, in Tables 1 and 2 we summarize the results of a larger scale experiment. Further tests of a more statistical nature are presented in Grindrod *et al.* (2008). Here we generated instances of RDRG and pRDRG linear and periodic networks and tested whether the likelihood ratio correctly identified the appropriate structure. We used dimensions $N = 100, 200, 500, 1000, 2000$ and a range of λ values in the interval $[0.6, 1)$; smaller values of λ produce unreasonably sparse networks—at $\lambda = 0.6$ the leading term $N\lambda/(1 - \lambda)$ in (4.1)–(4.3) indicates an average of only 1.5 edges per node. Each entry records the frequency of successful predictions over 1000 instances of the random graph. We see that the performance is perfect over a large range of parameter values and generally worsens as we increase N for a fixed λ and generally improves as we increase λ for a fixed N. This is consistent with the fact that decreasing the sparsity provides more information to the algorithm; the same argument accounts for the slightly improved performance on periodic networks in Table 2 over linear in Table 1. Of course, at the extreme case of $\lambda = 1$ all graphs are completely full and hence there can be no meaningful distinction, which explains the poor performance for $\lambda = 0.999$ and small N.

TABLE 1 *Linear RDRD networks: frequency with which the likelihood ratio correctly predicted that the network is linear rather than periodic*

	N				
λ	100	200	500	1000	2000
0.6	0.544	0.570	0.532	0.487	0.541
0.7	0.898	0.904	0.886	0.860	0.763
0.8	0.964	0.997	1	1	1
0.9	0.993	1	1	1	1
0.95	1	1	1	1	1
0.99	0.995	1	1	1	1
0.999	0.025	0.184	1	1	1

TABLE 2 *pRDRG networks: frequency with which the likelihood ratio correctly predicted that the network is periodic rather than linear*

	N				
λ	100	200	500	1000	2000
0.6	0.610	0.491	0.466	0.513	0.479
0.7	0.986	0.987	0.956	0.929	0.756
0.8	1	1	1	1	1
0.9	1	1	1	1	1
0.95	1	1	1	1	1
0.99	1	1	1	1	1
0.999	0.718	1	1	1	1

Overall, Tables 1 and 2 give us some confidence that the biological data sets to be studied in Section 5 are amenable to analysis.

5. Biological data sets

Existing and improving high-throughput technologies in experimental biology produce large-scale data that are often represented by networks. In protein–protein interaction (PPI) networks, nodes stand for proteins and edges between pairs of nodes indicate that, according to the results of an experiment, those proteins interact. We applied the linear and periodic spectral reordering algorithms to publicly available PPI networks to test whether periodic structure is present in real-world networks and, consequently, close and long-distance neighbours can be better differentiated with the new algorithm.

We analysed 13 PPI networks of three different eukaryotic organisms: yeast, worm and human. Two yeast PPI networks are described in von Mering *et al.* (2002): a network defined by the top 11000 interactions (denoted Y11000 in Table 3) and its high confidence part (Y2455). Here an increase in confidence corresponds to keeping only those links that are consistent with other sources of biological data, so higher confidence networks have fewer edges and should contain fewer false positives. A further three yeast PPI networks are the ‘core’ from Ito *et al.* (2000), the network from Uetz *et al.* (2000) and the union of both, denoted YItoCore, YUetz and YItoCoreUetz, respectively.

Human PPI networks used in our experiments include three networks of different confidence level: high (hStelzlH), high and medium (hStelzlHM) and high, medium and low (hStelzlHML) from Stelzl *et al.* (2005) and a network from Rual *et al.* (2005) (hRual). A further two networks were downloaded from databases BIND and MINT (Zanzoni *et al.*, 2002; Bader *et al.*, 2003) (hBIND and hMINT). Finally, two worm PPI networks were tested: WCore denotes the worm *Clostridium elegans* ‘core’ PPI network (Li *et al.*, 2004) and WZhSt denotes the worm PPI network from Zhong & Sternberg (2006).

Note that PPI networks generally consist of a set of disconnected components or *subnetworks*. It is known that if a network has k subnetworks then the lowest k eigenvalues of the Laplacian (or normalized Laplacian) matrix are zero (Ding *et al.*, 2001). The total number of subnetworks is shown as ‘sub’ in Table 3. In each case we studied the largest connected subnetwork. Thus, the original number of proteins

TABLE 3 *Linear versus period reordering for PPI data sets*

PPI	sub	orig.n	red.n	orig.edge	red.edge	λ_{per}	L
Y11000	103	2401	2137	11000	10816	0.84	-1.39×10^{-2}
Y2455	132	988	573	2455	2097	0.79	-1.25×10^{-2}
YItoCore	132	786	417	789	511	0.55	-2.83×10^{-2}
YUetz	163	991	473	915	543	0.53	-1.23×10^{-2}
YItoCoreUetz	160	1417	970	1520	1229	0.56	-9.03×10^{-3}
hStelzlH	22	363	314	756	727	0.70	-3.69×10^{-2}
hStelzlHM	34	1159	1076	2167	2116	0.66	-5.62×10^{-4}
hStelzlHML	47	1529	1411	2667	2594	0.65	1.31×10^{-3}
hRual	84	1873	1686	3463	3359	0.66	1.50×10^{-2}
hBIND	136	2181	1818	3005	2725	0.60	-9.93×10^{-3}
hMINT	109	1753	1446	3113	2896	0.67	-1.23×10^{-2}
WCore	58	1356	1218	1983	1902	0.61	-1.80×10^{-2}
WZhSt	67	2254	2060	18185	18000	0.90	-1.21×10^{-2}

'orig.n' and edges 'orig.edge' from the published networks were reduced to 'red.n' and 'red.edge', corresponding to the largest subnetworks. The last two columns in Table 3 show the decay parameter λ_{per} (λ_{lin} are similar to λ_{per}) and log-likelihood ratio L.

We see from Table 3 that 11 of the 13 networks studied, including the high and high–medium confidence networks, have a negative likelihood ratio, indicating periodicity. Further, the values of the ratio are comparable with those arising when we tested data generated from the pRDRG and RDRG models.

To back up these results we now show some qualitative pictures. The yeast PPI network Y11000 consists of 11000 interactions between 2401 proteins. There are 103 subnetworks and the largest component involves 2137 proteins and 10816 interactions. Note that by reducing the original network to its largest subnetwork we removed only 264 proteins (11%) and 184 edges (1.7%). Figure 3 shows the adjacency matrices for linear and periodic spectral reorderings of these 2137 proteins. We see that the periodic reordering places interactions (edges) into the off-diagonal corners, thereby reducing the envelope around the diagonal, relative to the linear version. This supports the negative likelihood ratio of $L = -1.39 \times 10^{-2}$.

Figure 4 shows linear and periodic reorderings of YItoCore. The largest component consists of 417 proteins (out of 786) and 511 interactions (reduced from 789). This network is very sparse, with less than two edges per node on average. We obtained narrow envelopes with both reorderings; but in the periodic case the interactions are more tightly arranged along the diagonal, and this is reflected in the negative value $L = -2.83 \times 10^{-2}$.

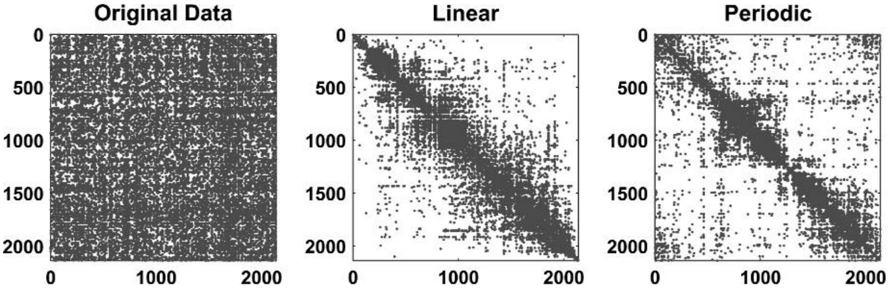

FIG. 3. Y11000—PPI network from von Mering *et al.* (2002): 2137 proteins and 10816 interactions: original adjacency matrix, the linear and periodic reorderings. The network is classified as periodic ($L = -1.39 \times 10^{-2} < 0$).

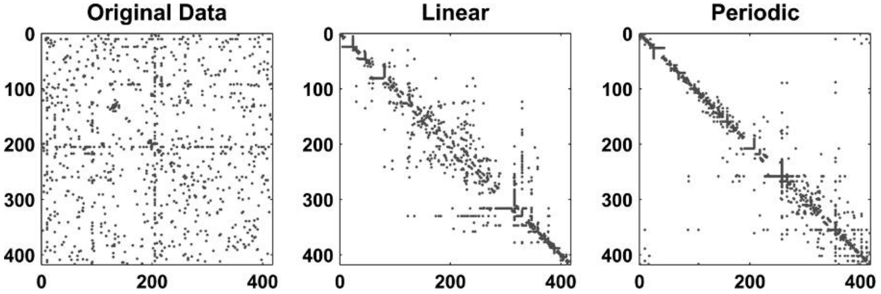

FIG. 4. YItoCore—PPI network from Ito *et al.* (2000): 417 proteins and 511 interactions. The network is classified as periodic ($L = -2.83 \times 10^{-2} < 0$).

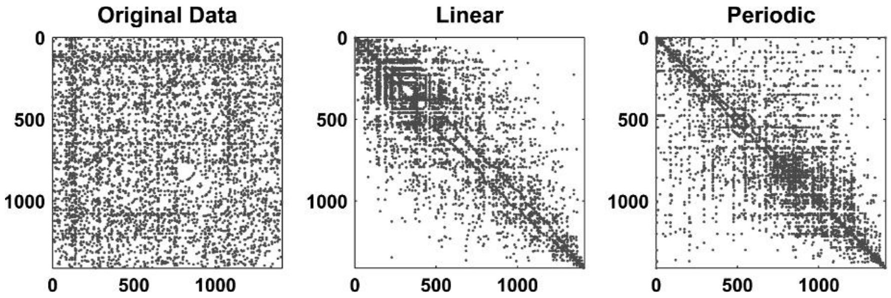

FIG. 5. hStelzlHML—PPI network from Stelzl *et al.* (2005): 1411 proteins and 2594 interactions. The network is classified as linear ($L = 1.31 \times 10^{-3} > 0$).

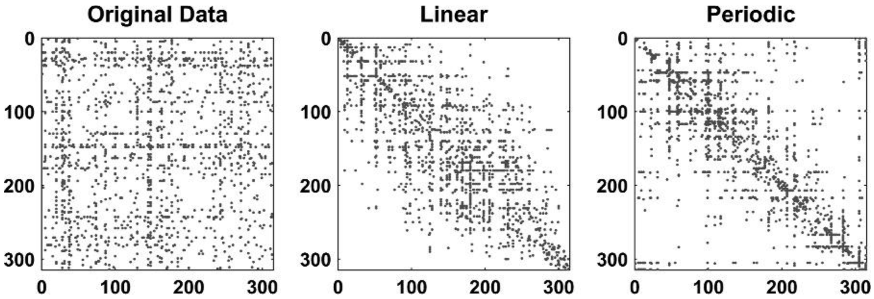

FIG. 6. hStelzlH—PPI network from Stelzl *et al.* (2005): 314 proteins and 727 interactions. The network is classified as periodic ($L = -3.69 \times 10^{-2} < 0$).

The human PPI network of 1411 proteins and 2594 interactions at high, medium and low confidence level, hStelzlHML, is one of the two cases that were classified as linear rather than periodic, $L = 1.31 \times 10^{-3} > 0$. Figure 5 illustrates the reorderings. We see that the periodic algorithm is not able to place nonzeros in the off-diagonal corners and does not tighten the envelope around the diagonal. However, the PPI network with only high confidence interactions (hStelzlH) was classified as periodic $L = -3.69 \times 10^{-2} < 0$ rather than linear; see Fig. 6.

6. Summary

Our aim here was to develop a new computational tool that finds an underlying periodic structure, if it exists, in large, complex, sparse networks. The new algorithm allows for both qualitative plots of the reordered adjacency matrix and a quantitative likelihood ratio for linear versus periodic structure. Applied to protein interactions, the algorithm produced strong evidence of periodicity. We believe that this is a promising approach for extracting meaning from complex networks, and in the context of bioinformatics it has the potential to reveal new insights concerning similarity between proteins and the nature of 'long-range' and 'short-range' interactions, both of which could be followed up experimentally.

Acknowledgement

This manuscript is dedicated to the memory of the late Ron Mitchell. PG and DJH remember him with great fondness as a wonderfully warm and inspirational colleague.

Funding

Engineering and Physical Sciences Research Council (GR/S62383/01); Medical Research Council (G0601353) to D.J.H.

References

ALON, U. (2006) *An Introduction to Systems Biology*. London: Chapman & Hall/CRC.

ALPERT, C. J. & YAO, S.-Z. (1995) Spectral partitioning: the more eigenvectors, the better. *Proceedings of the 32nd Conference on Design Automation*. pp. 195–200.

BADER, G. D., BETEL, D. & HOGUE, C. W. V. (2003) BIND: the biomolecular interaction network database. *Nucleic Acids Res.*, **31**, 248–250.

BARNARD, S. T., POTHEN, A. & SIMON, H. D. (1995) A spectral algorithm for envelope reduction of sparse matrices. *Numer. Linear Algebra Appl.*, **2**, 317–334.

DING, C. H. Q., HE, X., ZHA, H., GU, M. & SIMON, H. D. (2001) A min-max cut algorithm for graph partitioning and data clustering. *Proceedings of the 1st IEEE Conference on Data Mining*, 2001 (Nick Cercone, Tsau Young Lin and Xindong Wu, eds). pp. 107–114.

DUFF, I. S., ERISMAN, A. M. & REID, J. K. (1986) *Direct Methods for Sparse Matrices*. Oxford: Oxford University Press.

ELDÉN, L. (2007) *Matrix Methods in Data Mining and Pattern Recognition*. Philadelphia: SIAM.

ERDÖS, P. & RÉNYI, A. (1959) On random graphs. *Publ. Math. Debrecen*, **6**, 290–297.

GILBERT, E. N. (1959) Random graphs. *Ann. Math. Stat.*, **30**, 1141–1144.

GRINDROD, P. (2002) Range-dependent random graphs and their application to modeling large small-world proteome datasets. *Phys. Rev. E*, **66**, 066702-1–066702-7.

GRINDROD, P., HIGHAM, D. J. & KALNA, G. (2008) *Periodic reordering. Mathematics Research Report 06/2008*. Glasgow: Department of Mathematics, University of Strathclyde.

HIGHAM, D. J. (2003) Unravelling small world networks. *J. Comput. Appl. Math.*, **158**, 61–74.

HIGHAM, D. J. (2005) Spectral reordering of a range-dependent weighted random graph. *IMA J. Numer. Anal.*, **25**, 443–457.

HIGHAM, D. J., KALNA, G. & KIBBLE, M. (2007) Spectral clustering and its use in bioinformatics. *J. Comput. Appl. Math.*, **204**, 25–37.

HU, Y. & SCOTT, J. A. (2003) `HSL_MC73`: a fast multilevel Fiedler and profile reduction code. *RAL-TR-2003-36*. Didcot: Numerical Analysis Group, Computational Science and Engineering Department, Rutherford Appleton Laboratory.

ITO, T., TASHIRO, K., MUTA, S., OZAWA, R., CHIBA, T., NISHIZAWA, M., YAMAMOTO, K., KUHARA, S. & SAKAKI, Y. (2000) Toward a protein-protein interaction map of the budding yeast: a comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proc. Natl. Acad. USA*, **97**, 1143–1147.

KALNA, G., VASS, J. K. & HIGHAM, D. J. (2008) Multidimensional partitioning and bi-partitioning: analysis and application to gene expression datasets. *Int. J. Comput. Math.*, **85**, 475–485.

LI, S., ARMSTRONG, C. M., BERTIN, N., GE, H., MILSTEIN, S., BOXEM, M., VIDALAIN, P.-O., HAN, J.-D. J., CHESNEAU, A., HAO, T., GOLDBERG, D. S., LI, N., MARTINEZ, M., RUAL, J.-F., LAMESCH, P., XU, L., TEWARI, M., WONG, S. L., ZHANG, L. V., BERRIZ, G. F., JACOTOT, L., VAGLIO, P., REBOUL, J., HIROZANE-KISHIKAWA, T., LI, Q., GABEL, H. W., ELEWA, A., BAUMGARTNER, B., ROSE, D. J., YU, H., BOSAK, S., SEQUERRA, R., FRASER, A., MANGO, S. E., SAXTON, W. M., STROME, S., VAN DEN HEUVEL, S., PIANO, F., VANDENHAUTE, J., SARDET, C., GERSTEIN, M., DOUCETTE-STAMM, L., GUNSALUS, K. C., HARPER, J. W., CUSICK, M. E., ROTH, F. P., HILL, D. E. & VIDAL, M. (2004) A map of the interactome network of the metazoan C. elegans. *Science*, **303**, 540–543.

NEWMAN, M. E. J. (2003) The structure and function of complex networks. *SIAM Rev.*, **45**, 167–256.

NEWMAN, M. E. J., MOORE, C. & WATTS, D. J. (2000) Mean-field solution of the small-world network model. *Phys. Rev. Lett.*, **84**, 3201–3204.

RUAL, J. F., VENKATESAN, K., HAO, T., HIROZANE-KISHIKAWA, T., DRICOT, A., LI, N., BERRIZ, G. F., GIBBONS, F. D., DREZE, M., AYIVI-GUEDEHOUSSOU, N., KLITGORD, N., SIMON, C., BOXEM, M., MILSTEIN, S., ROSENBERG, J., GOLDBERG, D. S., ZHANG, L. V., WONG, S. L., FRANKLIN, G., LI, S., ALBALA, J. S., LIM, J., FRAUGHTON, C., LLAMOSAS, E., CEVIK, S., BEX, C., LAMESCH, P., SIKORSKI, R. S., VANDENHAUTE, J., ZOGHBI, H. Y., SMOLYAR, A., BOSAK, S., SEQUERRA, R., DOUCETTE-STAMM, L., CUSICK, M. E., HILL, D. E., ROTH, F. P. & VIDAL, M. (2005) Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, **437**, 1173–1178.

SHI, J. & MALIK, J. (2000) Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, **22**, 888–905.

SIVIA, D. S. (2006) *Data Analysis: A Bayesian Tutorial*, 2nd edn. Cambridge: Oxford University Press.

SKILLICORN, D. (2007) *Understanding Complex Datasets: Data Mining using Matrix Decompositions*. Boca Raton: CRC Press.

SPENCE, A., STOYANOV, Z. & VASS, J. K. (2007) The sensitivity of spectral clustering applied to gene expression data. *Proceedings of the 1st International Conference on Bioinformatics and Biomedical Engineering* (Jack Y. Yang, Mary Qu Yang, Michelle M. Zhu Yanqing Zhang, Hamid R. Arabnia, Youping Deng and Nikolaos G. Bourbakis, eds). pp. 1343–1346.

STELZL, U., WORM, U., LALOWSKI, M., HAENIG, C., BREMBECK, F. H., GOEHLER, H., STROEDICKE, M., ZENKNER, M., SCHOENHERR, A., KOEPPEN, S., TIMM, J., MINTZLAFF, S., ABRAHAM, C., BOCK, N., KIETZMANN, S., GOEDDE, A., TOKSZ, E., DROEGE, A., KROBITSCH, S., KORN, B., BIRCHMEIER, W., LEHRACH, H. & WANKER, E. E. (2005) A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, **122**, 957–968.

STRANG, G. (2008) *Computational Science and Engineering*. Wellesley, MA: Wellesley-Cambridge Press.

STROGATZ, S. H. (2001) Exploring complex networks. *Nature*, **410**, 268–276.

UETZ, P., GIOT, L., CAGNEY, G., MANSFIELD, T. A., JUDSON, R. S., KNIGHT, J. R., LOCKSHON, E., NARAYAN, V., SRINIVASAN, M., POCHART, P., QURESHI-EMILI, A., LI, Y., GODWIN, B., CONOVER, D., KALBFLEISH, T., VIJAYADAMODAR, G., YANG, M., JOHNSTON, M., FIELDS, S. & ROTHBERG, J. M. (2000) A comprehensive analysis of protein-protein interactions in saccharomyces cerevisiae. *Nature*, **403**, 623–627.

VAN DRIESSCHE, R. & ROOSE, D. (1995) An improved spectral bisection algorithm and its application to dynamic load balancing. *Parallel Comput.*, **21**, 29–48.

VON MERING, C., KRAUSE, R., SNEL, B., CORNELL, M., OLIVER, S. G., FIELDS, S., & BORK, P. (2002) Comparative assessment of large-scale data sets of protein-protein interactions. *Nature*, **417**, 399–403.

WATTS, D. J. & STROGATZ, S. H. (1998) Collective dynamics of ‘small-world’ networks. *Nature*, **393**, 440–442.

ZANZONI, A., MONTECCHI-PALAZZI, L., QUONDAM, M., AUSIELLO, G., HELMER-CITTERICH, M. & CESARENI, G. (2002) MINT: a molecular interaction database. *FEBS Lett.*, **513**, 135–140.

ZHONG, W. & STERNBERG, P. W. (2006) Genome-wide prediction of C. elegans genetic interactions. *Science*, **311**, 1481–1484.