A SPECTRAL ALGORITHM FOR SERIATION AND THE CONSECUTIVE ONES PROBLEM*

JONATHAN E. ATKINS†, ERIK G. BOMAN‡, AND BRUCE HENDRICKSON§

Abstract.
In applications ranging from DNA sequencing through archaeological dating to sparse matrix reordering, a recurrent problem is the sequencing of elements in such a way that highly correlated pairs of elements are near each other. That is, given a correlation function f reflecting the desire for each pair of elements to be near each other, find all permutations π with the property that if $\pi(i) < \pi(j) < \pi(k)$ then $f(i,j) \geq f(i,k)$ and $f(j,k) \geq f(i,k)$. This *seriation problem* is a generalization of the well studied consecutive ones problem. We present a spectral algorithm for this problem that has a number of interesting features. Whereas most previous applications of spectral techniques provide only bounds or heuristics, our result is an algorithm that correctly solves a nontrivial combinatorial problem. In addition, spectral methods are being successfully applied as heuristics to a variety of sequencing problems, and our result helps explain and justify these applications.

Key words. seriation, consecutive ones property, eigenvector, Fiedler vector, analysis of algorithms

1. Introduction. Many applied computational problems involve ordering a set so that closely coupled elements are placed near each other. This is the underlying problem in such diverse applications as genomic sequencing, sparse matrix envelope reduction and graph linear arrangement as well as less familiar settings such as archaeological dating. In this paper we present a *spectral algorithm* for this class of problems. Unlike traditional combinatorial methods, our approach uses an eigenvector of a matrix to order the elements. Our main result is that this approach correctly solves an important ordering problem we call the *seriation problem* which includes the well known consecutive ones problem [5] as a special case.

More formally, we are given a set of n elements to sequence; that is, we wish to bijectively map the elements to the integers $1, \ldots, n$. We also have a symmetric, real valued *correlation function* (sometimes called a *similarity function*) that reflects the desire for elements i and j to be near each other in the sequence. We now wish to find all ways to sequence the elements so that the correlations are *consistent*; that is, if π is our permutation of elements and $\pi(i) < \pi(j) < \pi(k)$ then $f(i,j) \geq f(i,k)$ and $f(j,k) \geq f(i,k)$. Although there may be an exponential number of such orderings, they can all be described in a compact data structure known as a PQ-tree [5] which we review in the next section. Not all correlation functions allow for a consistent sequencing. If a consistent ordering is possible we will say the problem is *well posed*. Determining an ordering from a correlation function is what we will call the *seriation problem*, reflecting its origins in archaeology [29, 33].

* To Appear in the SIAM J. Comput.
This work was supported by the Applied Mathematical Sciences program, U.S. Dept. of Energy, Office of Energy Research, and was performed at Sandia National Labs, operated for the U.S. DOE under contract No. DE-AC04-76DP00789.

† Dept. Mathematics, University of Michigan, 2072 East Hall, Ann Arbor, MI 48109. atkinsje@math.lsa.umich.edu.
‡ Scientific Computing & Computational Mathematics, Gates Bldg. 2B, Stanford Univ., Stanford, CA 94305–9025. boman@sccm.stanford.edu.
§ Applied & Numerical Mathematics Dept., Sandia National Labs, Albuquerque, NM 87185–1110. bah@cs.sandia.gov.

The consecutive ones problem (C1P) is a closely related ordering problem. A $(0,1)$-matrix C has the *consecutive ones property* if there exists a permutation matrix Π such that for each column in ΠC, all the ones form a consecutive sequence. If a matrix has the consecutive ones property, then the consecutive ones problem is to find all such permutations. As shown by Kendall [19] and reviewed in §6, C1P is a special case of the seriation problem.

Our algorithm orders elements using their value in an eigenvector of a *Laplacian matrix* which we formally define in §2. Eigenvectors related to graphs have been studied since the 1950's (see, for example, the survey books by Cvetković et al. [8, 7]. Most of the early work involved eigenvectors of adjacency matrices. Laplacian eigenvectors were first studied by Fiedler [10, 11] and independently by Donath and Hoffman [9]. More recently, there have been a number of attempts to apply spectral graph theory to problems in combinatorial optimization. For example, spectral algorithms have been developed for graph coloring [3], graph partitioning [9, 28] and envelope reduction [4], and more examples can be found in the survey papers of Mohar [23, 24]. However, in most previous applications, these techniques have been used to provide bounds, heuristics, or in a few cases, approximation algorithms [2, 6, 14] for NP-hard problems. There are only a small number of previous results in which eigenvector techniques have been used to exactly solve combinatorial problems including finding the number of connected components of a graph [10], coloring k-partite graphs [3], and finding stable sets (independent sets) in perfect graphs [16]. This paper describes another such application.

Spectral methods are closely related to the more general method of *semidefinite programming*, which has been applied successfully to many combinatorial problems (e.g. MAX-CUT and MAX-2SAT[14] and graph coloring[18]). See Alizadeh[1] for a survey of semidefinite programming with applications to combinatorial optimization.

Our result is important for several reasons. First, it provides new insight into the well-studied consecutive ones problem. Second, some important practical problems like envelope reduction for matrices and genomic reconstruction can be thought of as variations on seriation. For example, if biological experiments were error-free, the genomic reconstruction problem would be precisely C1P. Unfortunately, real experimental data always contain errors, and attempts to generalize the consecutive ones concept to data with errors seems to invariably lead to NP-complete problems [31, 15]. A spectral heuristic based upon our approach has recently been applied to such problems and found to be highly successful in practice [15]. Our result helps explain this empirical success by revealing that in the error-free case the technique will correctly solve the problem. This places the spectral method on a stronger theoretical footing as a cross between a heuristic and an exact algorithm. Similar comments apply to envelope reduction. Matrices with dense envelopes are closely related to matrices with the consecutive ones property. Recent work has shown spectral techniques to be better in practice than any existing combinatorial approaches at reducing envelopes [4]. Our result sheds some light on this success.

Another way to interpret our result is that we provide an algorithm for C1P that generalizes to become an attractive heuristic in the presence of errors. Designed as decision algorithms for the consecutive ones property, existing combinatorial approaches for C1P break down if there are errors and fail to provide useful approximate orderings. However, our goal here is not to analyze the approach as an approximation algorithm, but rather to prove that it correctly solves error-free problem instances.

This paper is organized in the following way. In the next section we introduce the mathematical notation and the results from matrix theory that we will need later. We also describe a spectral heuristic for ordering problems which motivates the remainder of the paper. The theorem that underpins our algorithm is proved in §3, the proof of which requires the use of a classic theorem from matrix analysis. Several additional results in §4 lead us to an algorithm and its analysis in §5. We review the connection to C1P in §6.

2. Mathematical background.

2.1. Notation and Definitions. Matrix concepts are useful because the correlation function defined above can be considered as a real, symmetric matrix. A permutation of the elements corresponds to a symmetric permutation of this matrix, a permutation of the matrix elements formed by permuting the rows and the columns in the same fashion. The question of whether or not the ordering problem is well posed can also be asked as a property of this matrix. Specifically, suppose the matrix has been permuted to reflect a consistent solution to the ordering problem. The off-diagonal matrix entries must now be non-increasing as we move away from the diagonal. More formally, we will say a matrix A is an *R-matrix*[1] if and only if A is symmetric and

$$\begin{aligned} a_{i,j} &\leq a_{i,k} \qquad \text{for } j < k < i, \\ a_{i,j} &\geq a_{i,k} \qquad \text{for } i < j < k. \end{aligned}$$

The diagonal entries of an R-matrix are unspecified. If A can be symmetrically permuted to become an R-matrix, then we say that A is *pre-R*. Note that pre-R matrices correspond precisely to well-posed ordering problems. Also, the R-matrix property is preserved if we add a constant to all off-diagonal entries, so we can assume without loss of generality that all off-diagonal values are non-negative.

When π is a permutation of the natural numbers $\{1, \ldots, n\}$ and x is a column vector, i.e. $x = [x_1, \ldots, x_n]^T$, we will denote by x^π the permutation of x by π, i.e. $x^\pi_i = x_{\pi(i)}$. Similarly, A^π is the symmetric permutation of A by π, i.e. $a^\pi_{i,j} = a_{\pi(i),\pi(j)}$. We denote by e the vector whose entries are all 1, by e_i the vector consisting of zeros except for a 1 in position i, and by I the identity matrix. A symmetric matrix A is *reducible* if there exists a permutation π such that

$$A^\pi = \begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix},$$

where B and C are non-empty square matrices. If no such permutation exists then A is *irreducible*. If B and C are themselves irreducible, then we refer to them as the *irreducible blocks* of A.

We say that λ is an *eigenvalue* of A if $Ax = \lambda x$ for some vector $x \neq 0$. A corresponding vector x is an *eigenvector*. An $n \times n$ real, symmetric matrix has n eigenvectors that can be constructed to be pairwise orthogonal, and its eigenvalues are all real. We will assume that the eigenvalues are sorted by increasing value, and refer to them as λ_i, $i = 1, \ldots, n$. The *(algebraic) multiplicity* of an eigenvalue λ is

[1] This class of matrices is named after W. S. Robinson who first defined this property in his work on seriation methods in archaeology [29].

defined as the number of times λ occurs as a root in the characteristic polynomial $p(z) = \det(A - zI)$. A value that occurs only once is called *simple*; the eigenvector of a simple eigenvalue is unique (up to normalization). We write $A \geq 0$ and say A is non-negative if all its elements $a_{i,j}$ are non-negative. A real vector x is *monotone* if $x_i \leq x_{i+1}$ for all $1 \leq i < n$ or if $x_i \geq x_{i+1}$ for all $1 \leq i < n$.

We define the *Laplacian* of a symmetric matrix A to be $L_A = D_A - A$, where D_A is a diagonal matrix with $d_{i,i} = \sum_{j=1}^{n} a_{i,j}$. The minimum eigenvalue with an eigenvector orthogonal to e (the vector of all ones) is called the *Fiedler value* and a corresponding eigenvector is called a *Fiedler vector*[2]. Alternatively, the Fiedler value is given by

$$\min_{x^T e = 0, x^T x = 1} x^T L_A x,$$

and a Fiedler vector is any vector x that achieves this minimum while satisfying these constraints. When $A \geq 0$ and irreducible, it is not hard to show that the Fiedler value is the smallest non-zero eigenvalue and a Fiedler vector is any corresponding eigenvector. We will be notationally cavalier and refer to the Fiedler value and vector of A when we really mean those of L_A.

2.2. PQ-trees. A *PQ-tree* is a data structure introduced by Booth and Lueker to efficiently encode a set of related permutations [5]. A PQ-tree over a set $U = \{u_1, u_2, \ldots, u_n\}$ is a rooted, ordered tree whose leaves are elements of U and whose internal nodes are distinguished as either P-nodes or Q-nodes. A PQ-tree is *proper* when the following three conditions hold:

1. Every element $u_i \in U$ appears precisely once as a leaf.
2. Every P-node has at least two children.
3. Every Q-node has at least three children.

Two PQ-trees are said to be equivalent if one can be transformed into the other by applying a sequence of the following two equivalence transformations:

1. Arbitrarily permute the children of a P-node.
2. Reverse the children of a Q-node.

Conveniently, the equivalence class represented by a PQ-tree corresponds precisely to the set of permutations consistent with an instance of a seriation problem. In §5 we describe an algorithm which uses Laplacian eigenvectors to construct a PQ-tree for an instance of the seriation problem.

2.3. Motivation for Spectral Methods. With the above definitions we can describe a simple heuristic for the seriation problem that will motivate the remainder of the paper. This heuristic is at the heart of the more complex algorithms we will devise, and underlies many previous applications of spectral algorithms [17]. We begin by constructing a simple penalty function g whose value will be small when closely correlated elements are close to each other. We define $g(\pi) = \sum_{(i,j)} f(i,j)(\pi_i - \pi_j)^2$. Unfortunately, minimizing g is NP-hard due to the discrete nature of the permutation [13]. Instead we approximate it by a function h of continuous variables x_i that we can minimize and that maintains much of the structure of g. We define $h(x) = \sum_{(i,j)} f(i,j)(x_i - x_j)^2$. Note that h does not have a unique minimizer, since its value does not change if we add a constant to each x component. To avoid this ambiguity, we need to add a constraint like $\sum_i x_i = 0$. We still have a trivial solution

[2] This is in recognition of the work of Miroslav Fiedler [10, 11].

when all the x_i's are zero, so we need a second constraint like $\sum_i x_i^2 = 1$. The resulting minimization problem is now well defined.

$$\text{(1)} \qquad \begin{aligned} &\text{Minimize } h(x) = \sum_{(i,j)} f(i,j)(x_i - x_j)^2 \\ &\text{subject to: } \sum_i x_i = 0, \text{ and } \sum_i x_i^2 = 1. \end{aligned}$$

The solution to this continuous problem can be used as a heuristic for sequencing. Merely construct the solution vector x, sort the elements x_i and sequence based upon their sorted order. One reason this heuristic is attractive is that the minimization problem has an elegant solution. We can rewrite $h(x)$ as $x^T L_F x$ where $F = \{f_{ij}\}$ is the correlation matrix. The constraints require that x be a unit vector orthogonal to e, and since L_A is symmetric, all other eigenvectors satisfy the constraints. Consequently, a solution to the constrained minimization problem is just a Fiedler vector.

Even if the problem is not well posed, sorting the entries of the Fiedler vector generates an ordering that tries to keep highly correlated elements near each other. As mentioned above, this technique is being used for a variety of sequencing problems [4, 15, 17]. The algorithm we describe in the remainder of the paper is based upon this idea. However, when we encounter ties in entries of the Fiedler vector, we need to recurse on the subproblem encompassing the tied values. In this way, we are able to find all permutations which make a pre-R matrix into an R matrix.

3. The key theorem. Our main result is that a modification of the simple heuristic presented in §2.3 is actually an algorithm for well-posed instances of the seriation problem. Completely proving this will require us to deal with the special cases of multiple Fiedler vectors and ties within the Fiedler vector. The cornerstone of our analysis is a classical result in matrix theory due to Perron and Frobenius [27]. The particular formulation below can be found on page 46 of [30].

THEOREM 3.1 (PERRON-FROBENIUS). *Let M be a real, non-negative matrix. If we define $\rho(M) = \max_i |\lambda_i(M)|$, then*

1. *$\rho(M)$ is an eigenvalue of M, and*
2. *there is a vector $x \geq 0$ such that $Mx = \rho(M)x$.*

We are now ready to state and prove our main theorem.

THEOREM 3.2. *If A is an R-matrix then it has a monotone Fiedler vector.*

Proof. Our proof uses the Perron-Frobenius Theorem 3.1. The non-negative vector in that theorem will consist of differences between neighboring entries in the Fiedler vector of the Laplacian of A.

First define the matrix $S \in \mathbb{R}^{(n-1)\times n}$ as

$$S = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix}.$$

Note that for any vector x, $Sx = (x_2 - x_1, \ldots, x_n - x_{n-1})^T$. Define $T \in \mathbb{R}^{n \times (n-1)}$ by

$$T = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 1 & 0 & & 0 \\ 1 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 1 & 1 & \cdots & 1 \end{bmatrix}.$$

It is easy to verify that $ST = I_{n-1}$, and that $TS = I_n - ee_1^T$. We define $M_A = SL_AT = \{m_{i,j}\}$ and let $L_A = \{l_{i,j}\}$. We now show that Sx is an eigenvector of M_A if and only if x is an eigenvector of L_A and $x \neq \alpha e$.

$$\begin{aligned} L_Ax &= \lambda x, \;\; x \neq \alpha e \Longleftrightarrow \\ SL_Ax &= \lambda Sx, \;\; x \neq \alpha e \Longleftrightarrow \\ SL_A(I - ee_1^T)x &= \lambda Sx, \;\; x \neq \alpha e \Longleftrightarrow \\ SL_ATSx &= \lambda Sx, \;\; x \neq \alpha e \Longleftrightarrow \\ M_Ay &= \lambda y, \text{ where } y = Sx \neq 0. \end{aligned}$$

The transformation from the second to the third line follows from $L_Ae = 0$. Equivalence holds between all the above equations, so λ is an eigenvalue for both L_A and M_A for eigenvectors of L_A other than e. Hence the eigenvalues of M_A are the same as the eigenvalues of L_A with the zero eigenvalue removed, and the eigenvectors of M_A are differences between neighboring entries of the corresponding eigenvectors of L_A.

It is easily seen that $(SL_A)_{i,k} = -l_{i,k} + l_{i+1,k}$ for all i, k, so

$$m_{i,j} = \sum_{k=1}^{n}(SL_A)_{i,k}T_{k,j} = \sum_{k=j+1}^{n}(-l_{i,k} + l_{i+1,k}) = \sum_{k=j+1}^{n}(a_{i,k} - a_{i+1,k}).$$

Since, by assumption, A is an R-matrix, $a_{i,k} \leq a_{i+1,k}$ for $i < k+1$, and therefore $m_{i,j} \leq 0$ for $i < j$. For $i > j$ we can use the fact that $\sum_{k=1}^{n} l_{i,k} = 0$ to obtain

$$m_{i,j} = \sum_{k=j+1}^{n}(-l_{i,k} + l_{i+1,k}) = \sum_{k=1}^{j}(l_{i,k} - l_{i+1,k}) = \sum_{k=1}^{j}(-a_{i,k} + a_{i+1,k}).$$

Again, from the R-matrix property we conclude that $m_{ij} \leq 0$ for $i > j$. Consequently, all the off-diagonal elements in M_A are non-positive.

Now let β be a value greater than $\max_i\{\lambda_i, m_{ii}\}$, where λ_i are the eigenvalues of M_A. Then $\tilde{M}_A = \beta I - M_A$ is non-negative with eigenvalues $\tilde{\lambda}_i = \beta - \lambda_i$. Also, $\tilde{M}_A$ and M_A share the same set of eigenvectors. By Theorem 3.1, there exists a non-negative eigenvector y of $\tilde{M}_A$ corresponding to the largest eigenvalue of $\tilde{M}_A$. But y is also an eigenvector of M_A corresponding to M_A's smallest eigenvalue. And this is just Sx, where x is a Fiedler vector of L_A. Since $y = Sx$ is non-negative, the corresponding Fiedler vector of L_A is non-decreasing and the theorem follows. (Note that since the sign of an eigenvector is unspecified, the Fiedler vector could also be non-increasing.)
□

THEOREM 3.3. *Let A be a pre-R matrix with a simple Fiedler value and a Fiedler vector with no repeated values. Let π_1 (respectively π_2) be the permutation induced by* *sorting the values in the Fiedler vector in increasing (decreasing) order. Then A^{π_1} and A^{π_2} are R-matrices, and no other permutations of A produce R-matrices.*

Proof. First note that since the Fiedler value is simple, the Fiedler vector is unique up to a multiplicative constant. Next observe that if x is the Fiedler vector of A, then x^π is the Fiedler vector of A^π. So applying a permutation to A merely changes the order of the entries in the Fiedler vector. Now let π_* be a permutation such that A^{π_*} is an R-matrix. By Theorem 3.2 x^{π_*} is monotone since x is the only Fiedler vector. Since x has no repeated values, π_* must be either π_1 or π_2. □

Theorem 3.3 provides the essence of our algorithm for the seriation problem, but it is too restrictive as the Fiedler value must be simple and contain no repeated values. We will show how to remove these limitations in the next section.

4. Removing the restrictions. Several observations about the seriation problem will simplify our analysis. First note that if we add a constant to all the correlation values the set of solutions is unchanged. Consequently, we can assume without loss of generality that the smallest value of the correlation function is zero. Note that subtracting the smallest value from all correlation values does not change whether or not the matrix is pre-R. In our algebraic formulation this translates into the following.

LEMMA 4.1. *Let A be a symmetric matrix and let $\bar{A} = A - \alpha ee^T$ for some real α. A vector x is a Fiedler vector of A iff x is a Fiedler vector of $\bar{A}$. So without loss of generality we can assume that the smallest off-diagonal entry of A is zero.*

Proof. By the definition of a Laplacian it follows that $L_{\bar{A}} = L_A + \alpha ee^T - \alpha nI$, where n is the dimension of A. Then $L_{\bar{A}}e = 0$ but for any other eigenvector x of L_A, $L_{\bar{A}}x = L_A x + 0 - \alpha nx$. That is, the eigenvalues are simply shifted down by αn while the eigenvectors are preserved. □

This will justify the first step of our algorithm, which subtracts the value of the smallest correlation from every correlation. Accordingly, we now make the assumption that our pre-R matrix has smallest off diagonal entry of zero. Next observe that if A is reducible then the seriation problem can be decoupled. The irreducible blocks of the matrix correspond to connected components in the graph of the nonzero values of the correlation function. We can solve the subproblems induced by each of these connected components, and link the pieces together in an arbitrary order. More formally, we have the following lemma.

LEMMA 4.2. *Let A_i, $i = 1, \ldots, k$ be the irreducible blocks of a pre-R matrix A, and let π_i be a permutation of block A_i such that the submatrix $A_i^{\pi_i}$ is an R-matrix. Then any permutation formed by concatenating the π_i's will make A become an R-matrix. In terms of a PQ-tree, the π_i permutations are children of a single P-node.*

Proof. By Lemma 4.1, we can assume all entries in the irreducible blocks are non-negative. Consequently, the correlation between elements within a block will always be at least as strong as the correlation between elements in different blocks. Also, by the definition of irreducibility, each element within a block must have some positive correlation with another element in that block. Hence, any ordering that makes A_i an R-matrix must not interleave elements between different irreducible blocks. As long as the blocks themselves are ordered to be R-matrices, any ordering of blocks will make A an R-matrix since correlations across blocks are all identical. □

With these preliminaries, we will now assume that the smallest off-diagonal value is zero and that the matrix is irreducible. As the following three lemmas and theorem show, this is sufficient to ensure that the Fiedler vector is unique up to a multiplicative constant.

LEMMA 4.3. *Let A be an $n \times n$ R-matrix with a monotone Fiedler vector x. If $\mathcal{J} = [r, s]$ is a maximal interval such that $x_r = x_s$, then for any $k \notin \mathcal{J}$, $a_{r,k} = a_{r+1,k} = \ldots = a_{s,k}$.*

Proof. We can without loss of generality assume x is non-decreasing since $-x$ is also a Fiedler vector. We will show that $a_{r,k} = a_{s,k}$ for all $k \notin \mathcal{J}$, and since A is an R-matrix then all elements between $a_{r,k}$ and $a_{s,k}$ must also be equal. Consider rows r and s in the equation $L_A x = \lambda x$.

$$\sum_{k=1}^{n} (l_{s,k} - l_{r,k}) x_k = \lambda(x_s - x_r) = 0$$

Since L_A is a Laplacian, we know that $\sum_{k=1}^{n} l_{i,k} = 0$ for all i. We get

$$\begin{aligned} 0 &= \sum_{k=1}^{n} (l_{sk} - l_{rk})(x_r - x_k) \\ &= \sum_{k=1}^{r-1} \underbrace{(l_{s,k} - l_{r,k})}_{\geq 0} \underbrace{(x_r - x_k)}_{>0} + \sum_{k=s+1}^{n} \underbrace{(l_{s,k} - l_{r,k})}_{\leq 0} \underbrace{(x_r - x_k)}_{<0} \end{aligned}$$

where we have used the fact that x is non-decreasing. Because all terms in the sum are non-negative, all terms must be exactly zero. By assumption, $x_k \neq x_r$ for $k \notin \mathcal{J}$ and consequently $l_{r,k} = l_{s,k}$ for $k \notin \mathcal{J}$ and the result follows. □

The following lemma is essentially a converse of this. Its proof requires detailed algebra, but it is not fundamental to what follows. Consequently, the proof is relegated to the end of this section.

LEMMA 4.4. *Let A be an irreducible $n \times n$ R-matrix with $a_{n,1} = 0$. If $\mathcal{J} = [r, s] \neq [1, n]$ is an interval such that $a_{r,k} = a_{s,k}$ for all $k \notin \mathcal{J}$, then $x_r = x_{r+1} = \ldots = x_s$ for any Fiedler vector x.*

LEMMA 4.5. *Let A be an irreducible R-matrix with $a_{n,1} = 0$, and x a monotone Fiedler vector of A. If $\mathcal{J} = [r, s]$ is an interval such that $x_r = x_{r+1} = \ldots = x_s$, then for any Fiedler vector y, $y_r = y_{r+1} = \ldots = y_s$.*

Proof. First apply Lemma 4.3 to conclude that for any $k \notin \mathcal{J}$, $a_{r,k} = a_{r+1,k} = \ldots = a_{s,k}$. Since $x^T e = 0$, it follows that $\mathcal{J} \neq [1, n]$. Now use this in conjunction with Lemma 4.4 to obtain the result. □

THEOREM 4.6. *If A is an irreducible R-matrix with $a_{n,1} = 0$, then the Fiedler value λ_2 is a simple eigenvalue.*

Proof. We will assume that λ_2 is a repeated eigenvalue and produce a contradiction. Let x and y be two linearly independent Fiedler vectors with x non-decreasing. Define $z(\theta) = \cos(\theta)x + \sin(\theta)y$, with $0 \leq \theta \leq \pi$. Let θ^* be the smallest value of θ that makes $z_k = z_{k+1}$ for some k where $x_k \neq x_{k+1}$. Such a θ^* must exist since x and y are linearly independent.

By Lemma 4.5 the indices of any repeated values in x are indices of repeated values in y and $z(\theta)$. Coupled with the monotonicity of x, this implies that $z(\theta^*)$ is monotone. By Lemma 4.5 the indices of any repeated values in $z(\theta^*)$ must be repeated in x which gives the desired contradiction. □

All that remains is to handle the situation where the Fiedler vector has repeated values. As the following theorem shows, repeated values decouple the problem into pieces that can be solved recursively.

THEOREM 4.7. *Let A be a pre-R matrix with a simple Fiedler value and Fiedler vector x. Suppose there is some repeated value β in x and define $\mathcal{I}$, $\mathcal{J}$ and $\mathcal{K}$ to be the indices for which*

1. *$x_i < \beta$ for all $i \in \mathcal{I}$,*
2. *$x_i = \beta$ for all $i \in \mathcal{J}$,*
3. *$x_i > \beta$ for all $i \in \mathcal{K}$.*

Then π is an R-matrix ordering for A iff π or its reversal can be expressed as (π_i, π_j, π_k), where π_j is an R-matrix ordering for the submatrix $A(\mathcal{J}, \mathcal{J})$ of A induced by $\mathcal{J}$, and π_i and π_k are the restrictions of some R-matrix ordering for A to $\mathcal{I}$ and $\mathcal{K}$, respectively.

Proof. From Theorem 3.2 we know that for any R-matrix ordering A^π, x^π is monotone, so elements in $\mathcal{I}$ must appear before (after) elements from $\mathcal{J}$ and elements from $\mathcal{K}$ must appear after (before) elements from $\mathcal{J}$. By Lemma 4.3, we have $a_{ik} = a_{jk}$ for all $i, j \in \mathcal{J}$ and $k \notin \mathcal{J}$. Hence the orderings of elements inside $\mathcal{J}$ must be indifferent to the ordering outside of $\mathcal{J}$ and vice versa. Consequently, the R-matrix ordering of elements in $\mathcal{J}$ depends only of $A(\mathcal{J}, \mathcal{J})$. □

Algorithmically, this theorem means that we can break ties in the Fiedler vector by recursing on the submatrix $A(\mathcal{J}, \mathcal{J})$ where $\mathcal{J}$ corresponds to the set of repeated values. The distinct values in the Fiedler vector of A constrain R-matrix orderings, but repeated values need to be handled recursively. In the language of PQ-trees, the distinct values are combined via a Q-node, and the components (subtrees) of the Q-node must then be expanded recursively.

Proof of Lemma 4.4. First we recall that the Fiedler value is the value obtained by

$$\min_{x^T e=0, x^T x=1} x^T L_A x = \min_{x^T e=0, x^T x=1} \sum_{i>j} a_{i,j}(x_i - x_j)^2, \tag{2}$$

and a Fiedler vector is a vector that achieves this minimum. We note that if we replace A by a matrix that is at least as large on an elementwise comparison then $x^T L_A x$ cannot decrease for any vector x.

We consider $A(\mathcal{J}, \mathcal{J})$, the diagonal block of A indexed by $\mathcal{J}$. By the definition of an R-matrix, all values in $A(\mathcal{J}, \mathcal{J})$ must be at least as large as $a_{r,s}$. However, $a_{r,s}$ must be greater than zero. Otherwise, by the R-matrix property $a_{i,j} = 0$ for all $i \geq r$ and $j < s$ and for all $j \geq r$ and $i < s$. But then by the statement of the theorem $a_{i,j} = 0$ for all $i \geq s$ and $j < s$ and all $j \geq r$ and $j < s$ which would make the matrix reducible.

The remainder of the proof will proceed in two stages. First we will force all the off-diagonal values in $A(\mathcal{J}, \mathcal{J})$ to be $a_{r,s}$ and show the result for this modified matrix. We will then extend the result to our original matrix.

Stage 1:
We define the matrix B to be identical to A outside of $B(\mathcal{J}, \mathcal{J})$, but all off-diagonal values of B within $B(\mathcal{J}, \mathcal{J})$ are set to $\alpha = a_{r,s}$. It follows from the hypotheses that B is an R-matrix. We define $\delta = l_{i,i}$ for $i \in \mathcal{J}$ and note that, by the R-matrix property, $\delta \leq (n-1)\alpha$.

We now define $\tilde{L_B} = L_B - (\delta + \alpha)I$ and consider the eigenvalue equation $\tilde{L_B} x = \tilde{\lambda}_2 x$. This matrix has the same eigenvectors as L_B with eigenvalues shifted by $\delta + \alpha$. Since $\tilde{l}_{ii} = \delta - (\delta + \alpha) = \alpha$ for $i \in \mathcal{J}$, all rows of $\tilde{L_B}$ in $\mathcal{J}$ are identical. Consequently, either all elements of x in $\mathcal{J}$ are equal, or $\tilde{\lambda}_2 = 0$ (which is equivalent to $\lambda_2 = \delta + \alpha$).

We will show that irreducibility and $a_{n1} = 0$ implies $\lambda_2 \neq \delta + \alpha$, which will complete the proof of **Stage 1**.

We assume $\lambda_2 = \delta + \alpha$ and look for a contradiction. We introduce a new matrix $\hat{B}$ as follows

$$\hat{b}_{i,j} = \begin{cases} b_{i,j} & \text{if } i < r \text{ and } j < r, \\ b_{i,j} & \text{if } i > s \text{ and } j > s, \\ \alpha & \text{otherwise.} \end{cases}$$

Since B is an R-matrix, $\hat{B}$ is at least as large as B elementwise, so $\lambda_2(\hat{B}) \geq \lambda_2(B)$. We define the vector $\hat{y}$ by

$$\hat{y}_i = \begin{cases} -(n-s), & \text{if } i < r, \\ 0, & \text{if } r \leq i \leq s, \\ r-1, & \text{if } i > s, \end{cases}$$

and $\hat{x}$ to be the unit vector in the direction of $\hat{y}$. We note that $\hat{x}^T e = 0$, and that $\hat{x}^T L_{\hat{B}} \hat{x} = n\alpha$. We have the following chain of inequalities.

$$\lambda_2 = \min_{x^T e = 0, x^T x = 1} x^T L_B x \leq \hat{x}^T L_B \hat{x} < \hat{x}^T L_{\hat{B}} \hat{x} = n\alpha. \tag{3}$$

The last inequality is strict since $\hat{b}_{n,1} = \alpha$ while $b_{n,1} = 0$ and $(\hat{x}_n - \hat{x}_1)^2 > 0$.

If $\lambda_2 = \delta + \alpha$ then we can combine an inequality due to Fiedler [10],

$$\lambda_2 \leq \frac{n}{n-1} \min_i l_{ii},$$

with the observation that $\min_i l_{i,i} \leq \delta$ to obtain $\lambda_2 \leq \frac{n}{n-1}\delta \leq \delta + \alpha = \lambda_2$. This can only be true if equality holds throughout, implying that $\delta = (n-1)\alpha$ and $\lambda_2 = n\alpha$. But this contradicts (3), so $\lambda_2 \neq \delta + \alpha$ and the proof of **Stage 1** is complete.

Stage 2:

We will now show that A and B have the same Fiedler vectors. Since A is elementwise at least as large as B, for any vector z, $z^T L_A z \geq z^T L_B z$. From **Stage 1** we know that any Fiedler vector of B satisfies $x_r = x_{r+1} = \ldots = x_s$. In this vector, $(x_i - x_j) = 0$ for $i, j \in \mathcal{J}$ so the contribution to the sum in (2) from $B(\mathcal{J}, \mathcal{J})$ is zero. But this contribution will also be zero when applied to $A(\mathcal{J}, \mathcal{J})$. Since A and B are identical outside of $A(\mathcal{J}, \mathcal{J})$ and $B(\mathcal{J}, \mathcal{J})$, we now have that a Fiedler vector of B gives an upper bound for the Fiedler value of A; that is, $\lambda_2(A) \leq \lambda_2(B)$. It follows that the Fiedler vectors of B are also Fiedler vectors of A and vice versa. $\square$

5. A spectral algorithm for the seriation problem. We can now bring all the preceding results together to produce an algorithm for well-posed instances of the seriation problem. Specifically, given a well-posed correlation function we will generate all consistent orderings. Given a pre-R matrix, our algorithm constructs a PQ-tree for the set of permutations that produce an R-matrix.

Our **Spectral-Sort** algorithm is presented in Fig. 1. It begins by translating all the correlations so that the smallest is 0. It then separates the irreducible blocks (if there are more than one) into the children of a P-node and recurses. If there is only one such block, it sorts the elements into the children of a Q-node based on their values in a Fiedler vector. If there are ties in the entries of the Fiedler vector, the algorithm is invoked recursively.

Input: A, an $n \times n$ pre-R matrix
U, a set of indices for the rows/columns of A
Output: T, a PQ-tree that encodes the set of all permutations π such that A^π is an R-matrix
begin
(1) $\alpha := \min_{i \neq j} a_{i,j}$
(1) $A := A - \alpha e e^T$
(2) $\{A_1, \ldots, A_k\} :=$ the irreducible blocks of A
(2) $\{U_1, \ldots, U_k\} :=$ the corresponding index sets
(2) **if** $k > 1$
(2) **for** $j := 1 : k$
(2) $T_j :=$ **Spectral-Sort**(A_j, U_j)
(2) **end**
(2) $T :=$ P-node$(T_1, T_2, \ldots, T_k)$
else
(3) **if** (n = 1)
(3) $T := u_1$
(3) **else if** (n = 2)
(3) $T :=$ P-node(u_1, u_2)
else
(4) $x :=$ Fiedler vector for L_A
(4) Sort x
(5) $t :=$ number of distinct values in x
(5) **for** $j := 1 : t$
(5) $V_j :=$ indices of elements in x with jth value
(5) $T_j :=$ **Spectral-Sort**$(A(V_j, V_j), V_j)$
(5) **end**
$T :=$ Q-node$(T_1, \ldots, T_T)$
end
end
end

Fig. 1. Algorithm **Spectral-Sort.**

We now prove that the algorithm is correct. Step (1) is justified by Lemma 4.1, and requires time proportional to the number of nonzeros in the matrix. The identification of irreducible blocks in step (2) can be performed with a breadth-first or depth-first search algorithm, also requiring time proportional to the number of nonzeros. Combining the permutations of the resulting blocks with a P-node is correct by Lemma 4.2.

Step (3) handles the boundary conditions of the recursion, while in step (4) the Fiedler vector is computed and sorted. If there are no repeated elements in the Fiedler vector then the Q-node for the permutation is correct by Theorem 3.3. Steps (3) and (4) are the dominant computational steps and we will discuss their run time below. The recursion in step (5) is justified by Theorem 4.7.

Note that this algorithm produces a tree whether A is pre-R or not. To determine whether A is pre-R, simply apply one of the generated permutations. If the result is an R-matrix then all permutations in the PQ-tree will solve the seriation problem, otherwise the problem is not well posed.

The most expensive steps in algorithm **Spectral-Sort** are the generation and sorting of the eigenvector. Since the algorithm can invoke itself recursively, these operations can occur on problems of size n, $n-1$, ...,1. So if the time for an eigencalculation on a matrix of size n is $T(n)$, the runtime of algorithm **Spectral-Sort** is $O(n(T(n)+n\log n))$.

A formal analysis of the complexity of the eigenvector calculation can be simplified by noting that for a Pre-R matrix, all that matters is the dominance relationships between matrix entries. So, without loss of generality, we can assume that all entries are integers less than n^2. With this observation, it is possible to compute the components of the Fiedler vector to a sufficient precision that the components can be correctly sorted in polynomial time. We now sketch one way this can be done, although we don't recommend this procedure in a real-world implementation.

Let λ denote a specific eigenvalue of L, in our case the Fiedler value. This can be computed in polynomial time as discussed in [25]. Then we can compute the corresponding eigenvector x symbolically by solving

$$(L-zI)x = 0 \bmod p(z)$$

where $p(z)$ is the characteristic polynomial of L. Gaussian elimination over a field is in P [21], so if $p(z)$ is irreducible we obtain a solution x where each component x_i is given by a polynomial in z with bounded integer coefficients. We note that letting z be any eigenvalue will force x to be a true eigenvector. If $p(z)$ is reducible, we try the above. If we fail to solve the equation, we will instead find a factorization of $p(z)$ and proceed by replacing $p(z)$ with the factor containing λ as a root. This yields a polynomial formula for each x_i and we can identify equal elements by e.g. the method in [22]. To decide the order of the remaining components, we evaluate the root λ to a sufficient precision and then compute the x_i's numerically and sort. Since λ is algebraic, the x_i's cannot be arbitrarily close [22] and polynomial precision is sufficient.

In practice, eigencalculations are a mainstay of the numerical analysis community. To calculate eigenvectors corresponding to the few highest or lowest eigenvalues (like the Fiedler vector), the method of choice is known as the Lanczos algorithm. This is an iterative algorithm in which the dominant cost in each iteration is a matrix-vector multiplication which requires $O(m)$ time. The algorithm generally converges in many fewer than n iterations, often only $O(\sqrt{n})$ [26]. However, a careful analysis reveals a dependence on the difference between the distinct eigenvalues.

6. The consecutive ones problem. Ordering an R-matrix is closely related to the consecutive ones problem. As mentioned in §1, a $(0,1)$-matrix C has the *consecutive ones property* if there exists a permutation matrix Π such that for each column in ΠC, all the ones form a consecutive sequence.[3] A matrix that has this property without any rearrangement (i.e. $\Pi = I$) is in Petrie form[4] and is called a P-matrix. Analogous to R-matrices, we say a matrix with the consecutive ones property is *pre-P*. The consecutive ones problem can be restated as: Given a pre-P matrix C, find a permutation matrix Π such that ΠC is a P-matrix.

[3] Some authors define this property in terms of rows instead of columns.

[4] Sir William M. F. Petrie was an archaeologist who studied mathematical methods for seriation in the 1890's.

There is a close relationship between P-matrices and R-matrices. The following results are due to D.G. Kendall and are proved in [19] and [33].

LEMMA 6.1. *If C is a P-matrix, then $A = CC^T$ is an R-matrix.*

LEMMA 6.2. *If C is pre-P and $A = CC^T$ is an R-matrix, then C is a P-matrix.*

THEOREM 6.3. *Let C be a pre-P matrix, let $A = CC^T$, and let Π be a permutation matrix. Then ΠC is a P-matrix if and only if $\Pi A \Pi^T$ is an R-matrix.*

This theorem allows us to use algorithm **Spectral-Sort** to solve the consecutive ones problem. First construct $A = CC^T$, and then apply our algorithm to A (note that the elements of A are small non-negative integers). Now apply one of the permutations generated by the algorithm to C. If the result is a P-matrix then all the permutations produce C1P orderings. If not, then C has no C1P orderings.

The run time for this technique is not competitive with the linear time algorithm for this problem due to Booth and Lueker [5]. However, unlike their approach, our **Spectral-Sort** algorithm does not break down in the presence of errors and can instead serve as a heuristic.

Several other combinatorial problems have been shown to be equivalent to the consecutive ones problem. Among these are recognizing interval graphs [5, 12] and finding dense envelope orderings of matrices [5].

One generalization of P-matrices is to matrices with unimodal columns (a unimodal sequence is a sequence that is non-decreasing until it reaches its maximum, then non-increasing). These matrices are called unimodal matrices [32]. Kendall [20] showed that the results 6.1 – 6.3 are also valid for unimodal matrices if the regular matrix product is replaced by the matrix *circle product* defined by

$$(A \circ B)_{ij} = \sum_k \min(a_{ik}, b_{kj}).$$

Note that P-matrices are just a special case of unimodal matrices, and that the circle product is equivalent to matrix product for $(0,1)$-matrices. Kendall's result implies that our spectral algorithm will correctly identify and order unimodal matrices.

Acknowledgements. We are indebted to Robert Leland for innumerable discussions about spectral techniques and to Sorin Istrail for his insights into the consecutive ones problem and his constructive feedback on an earlier version of this paper. We are further indebted to David Greenberg for his experimental testing of our approach on simulated genomic data, and to Nabil Kahale for showing us how to simplify the proof of Theorem 3.2. We also appreciate the highly constructive feedback provided by an anonymous referee.

REFERENCES

[1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM J. Optim.*, 5:13–51, 1995.

[2] N. Alon and N. Kahale. A spectral technique for coloring random 3–colorable graphs. In *Proc. 26th Annual Symposium on Theory of Computing*, pages 346–355. ACM, 1994.

[3] B. Aspvall and J. R. Gilbert. Graph coloring using eigenvalue decomposition. *SIAM J. Alg. Disc. Meth.*, 5(4):526–538, 1984.

[4] S. T. Barnard, A. Pothen, and H. D. Simon. A spectral algorithm for envelope reduction of sparse matrices. In *Proc. Supercomputing '93*, pages 493–502. IEEE, 1993.

[5] K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ–tree algorithms. *JCSS*, pages 333–379, 1976.

[6] F. R. K. Chung and S.-T. Yau. A near optimal algorithm for edge seperators (preliminary version). In *Proc. 26th Annual Symposium on Theory of Computing*, pages 1–8. ACM, 1994.

[7] D. M. Cvetković, M. Doob, I. Gutman, and A. Torgasev. *Recent Results in the Theory of Graph Spectra*. North-Holland, 1988. (Annals of Discrete Mathematics 36).

[8] D. M. Cvetković, M. Doob, and H. Sachs. *Spectra of Graphs: Theory and Application*. Academic Press, 1980.

[9] W. E. Donath and A. J. Hoffman. Lower bounds for the partitioning of graphs. *IBM J. Res. Develop.*, 17:420–425, 1973.

[10] M. Fiedler. Algebraic connectivity of graphs. *Czech. Math. Journal*, 23:298–305, 1973.

[11] M. Fiedler. A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czech. Math. Journal*, 25:619–633, 1975.

[12] D. R. Fulkerson and O. A. Gross. Incidence matrices and interval graphs. *Pac. J. Math.*, 3:835–855, 1965.

[13] A. George and A. Pothen. An analysis of spectral envelope–reduction via quadratic assignment problems. Technical Report ICASE Technical Report 94–81, NASA LaRC, Hampton, VA, September 1994. To appear in SIAM J. Matrix Anal.

[14] M. X. Goemans and D. P. Williamson. .878–approximation algorithms for MAX CUT and MAX 2SAT. In *Proc. 26th Annual Symposium on Theory of Computing*, pages 422–431. ACM, 1994.

[15] D. S. Greenberg and S. C. Istrail. Physical mapping by STS hybridization: Algorithmic strategies and the challenge of software evaluation. *J. Computational Biology*, 2(2):219–274, 1995.

[16] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988.

[17] M. Juvan and B. Mohar. Optimal linear labelings and eigenvalues of graphs. *Disc. Appl. Math.*, 36:153–168, 1992.

[18] D. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. *J. of the ACM*, 1996.

[19] D. G. Kendall. Incidence matrices, interval graphs and seriation in archaeology. *Pac. J. Math.*, 28(3):565–570, 1969.

[20] D. G. Kendall. Abundance matrices and seriation in archaeology. *Zeitschrift für Wahrscheinlichkeitstheorie*, 17:104–112, 1971.

[21] D. C. Kozen. *The Design and Analysis of Algorithms*. Springer–Verlag, 1992.

[22] M. Mignotte. Identification of algebraic numbers. *J. Algorithms*, 3:197–204, 1982.

[23] B. Mohar. The Laplacian spectrum of graphs. In Y. Alavi et al., editor, *Graph Theory, Combinatorics and Applications*, pages 871–898. J. Wiley, New York, 1991.

[24] B. Mohar. Laplace eigenvalues of graphs – a survey. *Disc. Math.*, 109:171–183, 1992.

[25] V. Pan. Algebraic complexity of computing polynomial zeros. *Comput. Math. Applic.*, 14(4):285–304, 1987.

[26] B. Parlett and D. Scott. The Lanczos algorithm with selective orthogonalization. *Math. Comp.*, 33:217–238, 1979.

[27] O. Perron. Zur Theorie der Matrizen. *Math. Ann.*, pages 248–263, 1907.

[28] A. Pothen, H. D. Simon, and K.-P. Liou. Partitioning sparse matrices with eigenvectors of graphs. *SIAM J. Matrix Anal.*, 11(3):430–452, July 1990.

[29] W. S. Robinson. A method for chronologically ordering archaeological deposits. *American Antiquity*, 16(4):293–301, 1951.

[30] R. S. Varga. *Matrix Iterative Analysis*. Prentice–Hall, 1962.

[31] M. Veldhorst. Approximation of the consecutive ones matrix augmentation problem. *SIAM J. Comput.*, 14(3):709–729, 1985.

[32] E. M. Wilkinson. *Mathematics in the Archaeological and Historical Sciences*, chapter Archaeological Seriation and the Travelling Salesman Problem, pages 276–284. University Press, Edinburgh, 1971.

[33] E. M. Wilkinson. Techniques of data analysis and seriation theory. In *Technische und Naturwissenschaftliche Beiträge Zur Feldarchäologie*, pages 1–142. Rheinland–Verlag, 1974.