An r-Dimensional Quadratic Placement Algorithm

Author(s): Kenneth M. Hall

Source: *Management Science*, Nov., 1970, Vol. 17, No. 3, Theory Series (Nov., 1970), pp. 219-229

Published by: INFORMS

Stable URL: https://www.jstor.org/stable/2629091

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at https://about.jstor.org/terms

INFORMS is collaborating with JSTOR to digitize, preserve and extend access to *Management Science*

JSTOR

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

MANAGEMENT SCIENCE
Vol. 17, No. 3, November, 1970
Printed in U.S.A.

AN r-DIMENSIONAL QUADRATIC PLACEMENT ALGORITHM*

KENNETH M. HALL†

State of California, Department of General Services

In this paper the solution to the problem of placing n connected points (or nodes) in r-dimensional Euclidean space is given. The criterion for optimality is minimizing a weighted sum of squared distances between the points subject to quadratic constraints of the form $X'X = 1$, for each of the r unknown coordinate vectors. It is proved that the problem reduces to the minimization of a sum or r positive semi-definite quadratic forms which, under the quadratic constraints, reduces to the problem of finding r eigenvectors of a special "disconnection" matrix. It is shown, by example, how this can serve as a basis for cluster identification.

1. Introduction

Many sequencing and placement problems can be characterized as follows: Given n points (or nodes) and an $n \times n$ symmetric connection matrix, $C = (c_{ij})$, where $c_{ii} = 0$, and $c_{ij} \geqq 0$, $i \neq j$, $i = 1, 2, \cdots, n$, is the "connection" between point i and point j, find locations for the n points which minimizes the weighted sum of squared distances between the points (i.e., weighted by c_{ij}).

If x_i denotes the X-coordinate of point i and z denotes the weighted sum of squared distances between the points, then the 1-dimensional problem is to find the row vector $X' = (x_1, x_2, \cdots, x_n)$ which minimizes

$$z = \tfrac{1}{2} \textstyle\sum_{i=1}^{n} \sum_{j=1}^{n} (x_i - x_j)^2 c_{ij} \tag{1.1}$$

where the prime denotes vector transposition. To avoid the trivial solution $x_i = 0$, for all i, the following quadratic constraint is imposed:

$$X'X = 1. \tag{1.2}$$

The solution to (1.1) and (1.2) is given in the next section. It is assumed that the non-interesting solution $x_i = x_j$, for all i and j, is to be avoided. Extensions to higher dimensions are given in §§3 and 4.

2. Optimum Solution in 1-Dimension (placement on a line)

Let $c_{i\cdot}$ and $c_{\cdot j}$ be the ith row sum and the jth column sum, respectively, of the (symmetric) matrix C. Define a diagonal matrix $D = (d_{ij})$ as follows:

$$\begin{aligned} d_{ij} &= 0, \qquad i \neq j, \\ &= c_{i\cdot}, \qquad i = j. \end{aligned}$$

Now, define the following matrix:

$$B = D - C. \tag{2.1}$$

* Received July 1968; revised May 1969, December 1969.

† The author wishes to acknowledge C. H. Mays for his valuable contributions in the original formulation of this model and the characterization of its solution. He has also pointed out some applications of this model the author was not aware of. The author also wishes to acknowledge one of the referees for pointing out Reference 5 and its relationship to the present work.

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

In words, the ith diagonal entry b_{ii} of B is the ith row (or column) sum of the connection matrix C and the off diagonal element b_{ij} is the negative of the corresponding entry in C. The matrix B plays a very fundamental role in this problem as we shall soon see. For brevity, B will be called the *disconnection* matrix.

Let $X' = (x_1, x_2, \cdots, x_n)$ be a row vector of X-coordinates, where the prime denotes vector transposition. Then (1.1) can be rewritten as $z = X'BX$, i.e.

$$z = \tfrac{1}{2} \textstyle\sum_i \sum_j (x_i - x_j)^2 c_{ij} \tag{2.2}$$

$$= \tfrac{1}{2} \textstyle\sum_i \sum_j (x_i^2 - 2x_i x_j + x_j^2) c_{ij} \tag{2.3}$$

$$= \tfrac{1}{2} (\textstyle\sum_i x_i^2 c_{i\cdot} - 2 \sum_i \sum_j x_i x_j c_{ij} + \sum_j x_j^2 c_{\cdot j}) \tag{2.4}$$

$$= \textstyle\sum_i x_i^2 c_{i\cdot} - \sum_j \sum_{i \neq j} x_i x_j c_{ij} \tag{2.5}$$

$$= X'BX. \tag{2.6}$$

Equation (2.5) follows because C is symmetrical (i.e., $c_{i\cdot} = c_{\cdot j}$). Equation (2.6) is immediate since (2.5) has yielded a quadratic form. Now we prove the following:

THEOREM. *Let G denote the underlying graph of the connection matrix C (i.e., an arc in G exists between node i and node j if and only if $c_{ij} > 0$). Then the following is true about the disconnection matrix, B:*

(i) *B is positive semi-definite ($B \geqq 0$), and*

(ii) *whenever G is connected, B is of rank $n - 1$.*

PROOF. To prove (i), we simply note from equations (2.6) and (2.2) that $X'BX$ can be written as a sum of nonnegative terms. Thus $B \geqq 0$. That the bound of zero can be reached can be seen from (2.2) by letting $x_i = x_j$ for all i and j.

Before proving (ii), we first note from (2.1) that the row sums of B are zero, so B has an eigenvector which is proportional to the unit vector, $U' = (1, 1, \cdots, 1)$. The associated eigenvalue is zero. If B is to have rank $n - 1$ the remaining $n - 1$ eigenvalues of B must necessarily be positive (a direct result from (i) above). We will prove that the required eigenvalues are, in fact, positive.

Let $0 = \lambda_1 \leqq \lambda_2 \leqq \lambda_3 \leqq \cdots \leqq \lambda_n$ be the eigenvalues of matrix B, with corresponding eigenvectors $E_1, E_2, \cdots, E_n$. E_1 is proportional to the unit vector, U, because the row sums of B are zero. The remaining eigenvectors, $E_2, E_3, \cdots, E_n$, being orthogonal to U (or E_1) must each have the sum of its components equal to zero. Therefore, some components will be negative and some will be positive and hence, not all components will be equal. Therefore, if we can prove: For connected G with $x_i \neq x_j$ for all i and j, that $X'BX$ is positive, our proof would be complete. We will prove this by contradiction, i.e., assume $X'BX = 0$ and show that it contradicts the hypothesis that $x_i \neq x_j$ for all i and j.

Rewrite (2.5) and (2.2) as

$$X'BX = \textstyle\sum_{i=1}^{n-1} \sum_{i<j} (x_i - x_j)^2 c_{ij} + \sum_{i=1}^{n-1} (x_i - x_n)^2 c_{in}.$$

Whenever $X'BX = 0$, both of the above terms on the right-hand side must be zero. Refer to these as RHSL and RHSR, respectively. Since G is connected, one or more of the coefficients c_{in}, $i \neq n$, must be positive. In all these cases x_i must equal x_n if RHSR is to be zero. Now form two sets of subscripts: $S_1 = \{i : c_{in} = 0\}$, $S_2 = \{i : x_i = x_n\}$. Note that S_2 contains all the subscripts with $c_{in} > 0$. If we can show that S_1 is a subset of S_2, our proof will be complete because then x_i would equal x_j for all i and j (providing us with the desired contradiction). Also, the proof is immediate if S_1 is empty. There-

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

fore, assume S_1 contains $m > 0$ elements. Choose an element from S_1, say i_1; then $c_{i_1 i_2} > 0$ for some $i_2 \neq i_1$ or else G would not be connected. This implies that $x_{i_2} = x_{i_1}$ or else RHSL would not be zero. Two cases must now be considered for i_2; either (1) $c_{i_2 n} > 0$, or (2) $c_{i_2 n} = 0$. If (1) holds then $x_{i_2} = x_n$, (or else RHSR would not be zero) which would imply that i_2 is in the set S_2. Since $x_{i_1} = x_{i_2}$ then i_1 would also be in S_2. If (2) holds then $c_{i_1 i_3} + c_{i_2 i_3} > 0$ for some $i_3 \neq i_2 \neq i_1$ or else G would not be connected. In any case, this implies $x_{i_3} = x_{i_2} = x_{i_1}$ or else RHSL would not be zero. As in (1) above, if $c_{i_3 n} > 0$, then i_3 and, consequently, i_1 and i_2 are in S_2. On the other hand, as in (2) above, if $c_{i_3 n} = 0$ we continue building up (from S_1) a subset of $r - 1 \leqq m$ elements, $\{i_j\}_{j=1}^{r-1}$, with $c_{i_1 i_2} > 0$, $c_{i_1 i_3} + c_{i_2 i_3} > 0, \cdots, \sum_{j=1}^{r-1} c_{i_j i_r} > 0$; $x_{i_1} = x_{i_2} = \cdots = x_{i_{r-1}}$; $c_{i_j n} = 0, j = 1, r - 1$, and $c_{i_r n} > 0$. The element i_r will eventually be reached if G is connected. When it is reached then i_r and, consequently, the subset $\{i_j\}_{j=1}^{r-1}$ will be in S_2. If $r = m + 1$, the proof would be finished since S_1 would be a subset of S_2. If $r \leqq m$, repeat the above process by building up a new subset of connected elements from S_1 (having equal coordinates if RHSL is to be zero) which eventually become "connected" to element n in S_2. When this happens, the entire subset will be in S_2. Only a finite number (at most, m) of such subsets need to be constructed to account for all the elements of S_1. Then S_1 will be a subset of S_2 since x_1 will equal x_j for all i and j. This provides the desired contradiction and completes our proof.

Now the problem has been reduced to the following form. Minimize

$$z = X'BX, \qquad B \geqq 0 \tag{2.7}$$

subject to the quadratic constraint

$$X'X = 1. \tag{2.8}$$

To minimize (2.7) subject to the constraint (2.8) introduce the Lagrange multiplier λ and form the Lagrangian $L = X'BX - \lambda(X'X - 1)$. Taking the first partial derivative of L with respect to the vector X and setting the result equal to zero yields $2BX - 2\lambda X = 0$. If I is the identity matrix, the above can be rewritten as

$$(B - \lambda I)X = 0 \tag{2.9}$$

which yields a nontrivial solution, X, if and only if λ is an eigenvalue of the matrix B and X is the corresponding eigenvector. If (2.9) is premultiplied by X' and the constraint (2.8) is imposed we obtain

$$\lambda = X'BX. \tag{2.10}$$

Thus, the formal solution to (1.1) and (1.2) is simply that X is the eigenvector of B which minimizes z and λ $(=z)$ is the corresponding eigenvalue. The minimum eigenvalue, zero, yields the noninteresting solution $X' = (1, 1, \cdots, 1)/\sqrt{n}$. Therefore the second smallest eigenvalue and the associated eigenvector yields the optimum solution. It is important to note that if the original problem is changed to a maximizing problem, then the maximum eigenvalue of B and the associated eigenvector will be the desired solution.

3. Extension to 2-Dimensions

Let $Y' = (y_1, y_2, \cdots, y_n)$ be a row vector of Y-coordinate of the n points. Then the problem is to determine X and Y which minimizes

$$z = X'BX + Y'BY, \qquad B \geqq 0 \tag{3.1}$$

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

subject to the following constraints

$$X'X = 1, \tag{3.2}$$

$$Y'Y = 1. \tag{3.3}$$

To solve (3.1)–(3.3), introduce the Lagrange multipliers α and β and form the Lagrangian $L = X'BX + Y'BY - \alpha(X'X - 1) - \beta(Y'Y - 1)$. Taking the first partial derivative of L with respect to the vector X and also with respect to the vector Y and setting the results equal to zero yields the two systems of equations

$$2BX - 2\alpha X = 0, \tag{3.4}$$

$$2BY - 2\beta Y = 0. \tag{3.5}$$

These yield nontrivial solutions X and Y if and only if X and Y are eigenvectors of B, associated with the eigenvalues α and β, respectively.

If $0 = \lambda_1 < \lambda_2 \leqq \lambda_3 \leqq \cdots \leqq \lambda_n$ denote the n eigenvalues of matrix B, then (3.1)–(3.3) are solved by taking $\alpha = \beta = \lambda_1$. If it is desired that X not be proportional to Y, then take $\alpha = \lambda_1$, $\beta = \lambda_2$. If, further, it is desired that not all x_i are equal, and not all y_i are equal, then take $\alpha = \lambda_2$, $\beta = \lambda_3$. The vectors X and Y will be the eigenvectors associated with the eigenvalues α and β in any case. Sometimes, it is desirable to have X and Y orthogonal. This will be true whenever $\alpha \neq \beta$. If (3.4) and (3.5) are premultiplied by X' and Y', respectively, and the constraints (3.2) and (3.3) are imposed, then we see that $z = \alpha + \beta$. Thus, the sum of the relevant eigenvalues used will yield the final value of z.

4. Extension to r-dimensions

For the r-dimensional problem z is simply the sum of r quadratic forms, one for each dimension. If each of the coordinate vectors is constrained to have inner product equal to 1, then setting each coordinate vector equal to the eigenvector associated with the eigenvalue λ_1 would solve the problem. If in each dimension it is required that not all components of the solution vector be equal, then taking the eigenvector associated with λ_2 would solve the problem. If the coordinate vectors must not be proportional to each other, take the eigenvector associated with the eigenvalues $\lambda_2, \lambda_3, \cdots, \lambda_{r+1}$. Thus, after finding the X-vector (from a knowledge of λ_2) and the Y-vector (from a knowledge of λ_3) additional coordinate vectors are found from a knowledge of successively larger eigenvalues. The final value of z will be the sum of the eigenvalues used.

5. Applications

Let c_{ij} be the flow between work center i and work center j in a job shop. Choosing X and Y to be the eigenvectors associated with λ_2 and λ_3, respectively, results in optimum global placement of the work centers in the plane.

Let c_{ij} denote the "distance" (or "dissimilarity") between animal i and animal j based on a set of measurements. Choosing X to be associated with the maximum eigenvalue, λ_n, results in an optimum sequencing of the animals on a line (numerical taxonomy).

Let c_{ij} represent the number of wires interconnecting a pair of electronic components i and j. Choosing X and Y to be the eigenvectors associated with λ_2 and λ_3, respectively, results in optimum placement of the electronic components in the plane in the sense of minimum squared wirelength.

Let c_{ij} be the flow between economic facility i and economic facility j. Choosing X

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

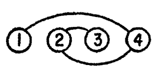
and Y to be the eigenvectors associated with λ_2, and λ_3, respectively, results in optimum global placement of the economic facilities in the plane.

6. Examples

Example 1. A 4-node graph, its connection matrix C and disconnection matrix B are illustrated in Figure 6.1. Arcs denote direct connections between nodes, with corresponding values given in matrix C.

The 4 eigenvalues of B and their associated eigenvectors E_1, E_2, E_3 and E_4 are shown in Figure 6.2. A plot of the 4 nodes is also shown in Figure 6.2, where E_2 and E_3 has been used as the X and Y-coordinate vectors, respectively. It should be noted that E_2 has "unraveled" the graph, whereas E_4 (the maximization problem) would have made it worse. E_4 would have yielded the sequence 4, 3, 1, 2 rather than 1, 4, 2, 3.

Example 2. Consider the 4-node graph, its connection matrix C and disconnection matrix B illustrated in Figure 6.3. The eigenvalues of B with their associated eigen vectors are shown in Figure 6.4. A plot of the 4 nodes, using E_2 and E_3 is also shown



$$C = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 2 & -1 & -1 \\ 0 & -1 & 1 & 0 \\ -1 & -1 & 0 & 2 \end{bmatrix}$$

FIGURE 6.1. A 4-Stage Shift Register

Node	E_1	E_2	E_3	E_4
1	0.5	−0.653	−0.5	0.270
2	0.5	0.270	0.5	0.653
3	0.5	0.653	−0.5	−0.270
4	0.5	−0.270	0.5	−0.653
λ =	(0.0	0.586	2.0	3.414)

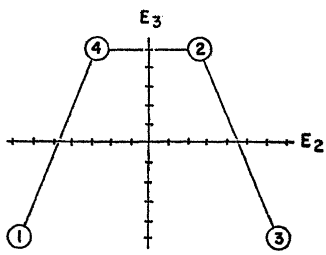


FIGURE 6.2. Plot of Figure 6.1 Using Eigenvectors E_2 and E_3

$$C = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix}$$

FIGURE 6.3. 4-Stage Shift Register with End Around Feedback

Node	E_1	E_2	E_3	E_4
1	0.5	−0.5	0.5	−0.5
2	0.5	0.5	0.5	0.5
3	0.5	0.5	−0.5	−0.5
4	0.5	−0.5	−0.5	0.5
λ =	(0.0	2.0	2.0	4.0)

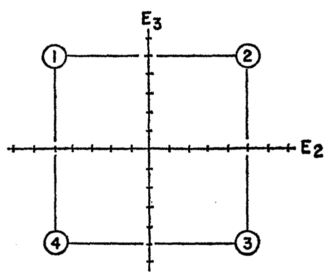
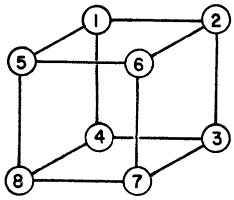


FIGURE 6.4. Plot of Figure 6.3 Using E_2 and E_3

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms



$$C = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

FIGURE 6.5. Vertices of a Cube

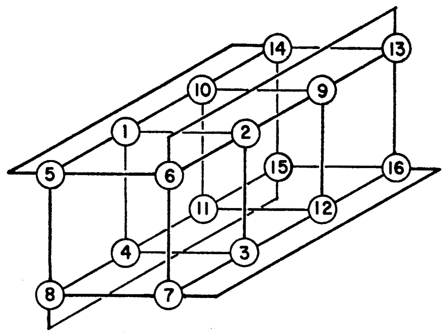


FIGURE 6.6. 4-Dimensional Hypercube

It should be noted that the 2 eigenvalues $\lambda_2 = \lambda_3 = 2$ used in Figure 6.4 are equal. Thus, the "Square" shown in Figure 4 can be rotated any amount about the origin without changing the value of $z = X'BX + Y'BY$ where $X = E_2$ and $Y = E_3$. For example, a 45° rotation clockwise rotation would yield the 2 new eigenvectors.

$$E'_2 = (0, 1, 0, -1)/\sqrt{2}, \qquad E'_3 = (1, 0, -1, 0)/\sqrt{2}$$

which also have eigenvalues $\lambda_2 = \lambda_3 = 2$.

Example 3. Consider the 8-node graph and its connection matrix shown in Figure 6.5. The disconnection matrix B is not shown since it can easily be constructed from C by inspection. The first 4 eigenvectors of B are given by

$$E'_1 = (1, 1, 1, 1, 1, 1, 1, 1)/\sqrt{8}$$

$$E'_2 = (-1, 1, 1, -1, -1, 1, 1, -1)/\sqrt{8}$$

$$E'_3 = (1, 1, -1, -1, 1, 1, -1, -1)/\sqrt{8}$$

$$E'_4 = (1, 1, 1, 1, -1, -1, -1, -1)/\sqrt{8}$$

which have eigenvalues $\lambda_1 = 0, \lambda_2 = \lambda_3 = \lambda_4 = 2$. If E_2, E_3 and E_4 are used to reposition the 8 nodes of Figure 6.5 in the X, Y and Z-directions, respectively, we find that the 8 nodes form the vertices of a cube. In this case there are 3 tied eigenvalues. Because of this spherical symmetry the cube can be rotated about the origin without changing the value of the loss function $z = X'BX + Y'BY + Z'BZ$, where $X = E_2$, $Y = E_3$ and $Z = E_4$. If only E_2 and E_3 are used, a 2-dimensional projection of the cube will result.

It is a simple matter to construct graphs with 4 and higher order ties e.g., the 4-dimensional hypercube of Figure 6.6. These situations do not arise much in practice.

Example 4 (Steinberg). Steinberg (1961) has described a 34 node problem in which the objective was to map 34 electronic components into a 4×9 rectangular grid (the

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

backboard). Various authors have tried their skill at this mapping problem with varying degrees of success. Although the general mapping problem is not solved in this paper, this author feels that one should first solve the global placement problem and use this solution as a starting point for mapping. Eigenvectors E_2, E_3, E_4 and E_5 are given below:

TABLE 6.1
Eigenvectors for Steinberg Problem

NODE	E2	E3	E4	E5
1	-0.0432012	-0.0694208	0.0052049	-0.0164071
2	-0.0651679	-0.1543718	-0.0077823	0.0296335
3	-0.0571173	-0.1095359	0.0060353	-0.0267673
4	-0.0463958	-0.0802566	0.0103127	-0.0174345
5	-0.0432456	-0.0754842	0.0296073	-0.0188960
6	-0.0371948	-0.0580382	0.0206740	-0.0207142
7	-0.0571307	-0.0633301	-0.0116307	-0.0549437
8	-0.0558912	-0.1151081	-0.0073685	0.0200939
9	-0.0553631	-0.1048123	0.0055148	-0.0274094
10	-0.0524252	-0.0916281	-0.0048700	-0.0069470
11	-0.0093916	-0.0343772	0.0114102	-0.0177897
12	0.0011335	-0.0388740	0.0087087	-0.0193481
13	-0.0241184	-0.0553201	0.0200537	-0.0186262
14	-0.0265103	0.0261925	-0.0049970	-0.0086094
15	-0.0733625	-0.0900708	-0.0422770	-0.2019506
16	-0.1241581	-0.2826598	-0.1615442	-0.7278175
17	-0.1273314	-0.4699478	-0.1974118	0.6141847
18	-0.0796975	-0.2166614	-0.0690472	0.1618230
19	-0.0628896	0.1268366	0.0193869	0.0066971
20	-0.0344391	0.0012820	0.0197461	0.0020348
21	0.0998517	0.0275425	0.4928094	0.0244346
22	0.1002615	0.0190828	0.7292042	0.0356757
23	0.0685555	0.0084281	0.0612788	0.0002766
24	0.6397462	0.0519768	-0.2774723	0.0089584
25	0.4273481	0.0241581	-0.0866789	0.0017937
26	0.4485591	0.0298216	-0.0816017	0.0031537
27	0.0962194	-0.0171636	0.0647895	0.0004566
28	-0.0838568	0.1101282	-0.0325383	-0.0036700
29	-0.1066123	0.2023506	-0.0590732	0.0091841
30	-0.1125110	0.2456635	-0.0769316	0.0328500
31	-0.1507772	0.3917571	-0.1376127	0.0687224
32	-0.0958222	0.2118927	-0.0466467	0.0381619
33	-0.1295471	0.3330358	-0.1028930	0.0697682
34	-0.1274837	0.3169311	-0.0963279	0.0594337
LAMBDA=(	14.9619904	21.5561523	26.0068207	29.4585571)

On examination of the above coordinates, it becomes an easy matter to identify the "top" of the circuit, the "bottom" of the circuit, the "left" and "right", etc. The proximity of elements (e.g., one element lies near another) is perhaps more important than the actual value of their coordinates. In Figure 6.7 Steinberg's data is graphed using E_2 and E_3 for the X and Y-coordinates, respectively. It seems as though, because of the way in which E_2 and E_3 has separated subsets of nodes in this problem (e.g., the nodes 19, 28–34 have been separated from the rest of the circuit), mapping of nodes into discrete locations might be facilitated if algorithms are constructed which incorporate these spacial relationships.

The author could not resist the temptation to try his hand at mapping the 34 components into the required 4 × 9 grid. In doing so, the graph of Figure 6.7 was followed

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

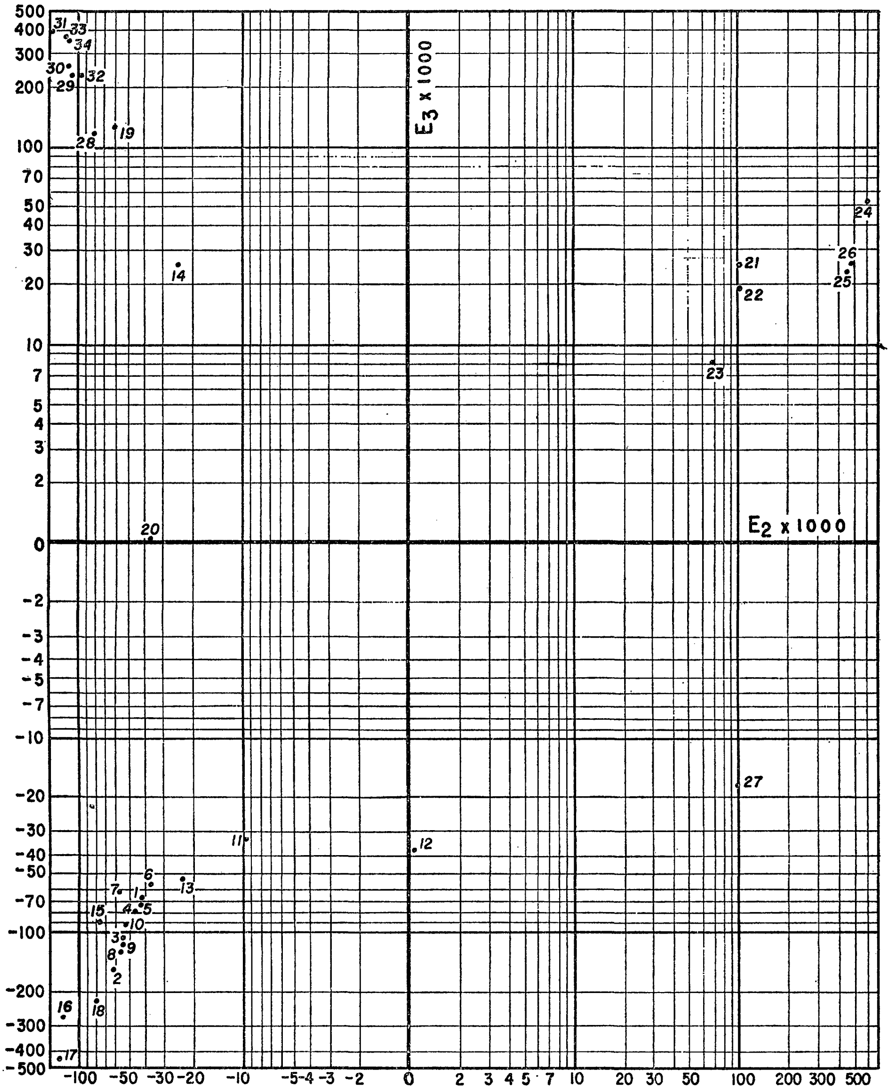


FIGURE 6.7. Graph of Steinberg Data

very closely, along with an occasional look at E_4 to resolve local ties. The grid was turned on end so the mapping was actually done on a 9×4 grid. After a little trial and error the result obtained is as follows:

34	33	23	26
31	22	21	25
30	32	27	24
29	14	11	12
19	20	6	13
28	7	1	5
16	15	10	4
17	18	8	9
		2	3

DISTANCES:

EUCLIDEAN = 4419.13

MANHATTAN (Dog-leg) = 5139.00

SQUARED EUCLIDEAN = 9699.00

FIGURE 6.8. Resulting Map from Analog Solution

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

In the next table we compare this solution to solutions that other authors have found.

AUTHOR	SQUARED EUCLIDEAN	EUCLIDEAN
Gilmore (n^5 algorithm)	11,929.000	4680.36
Graves & Whinston	11,909.000	4490.70
Steinberg	11,875.000	4894.54
Hillier & Connors	10,929.993	4821.78
Gilmore (n^4 algorithm)	10,656.000	4547.54
Hall	9,699.000	4419.13

FIGURE 6.9. Comparison of Solutions

Example 5 (Sokal). R. Sokal (1966) describes a problem in which the dissimilarity between 27 individuals from seven species of nematode worms (OTU's, or Operational Taxonomic Units) were measured and the object was to sequence the individuals on a line into homogeneous groups. A nematode pentagram (a tree-like structure) is then constructed, based on the final sequencing, to illustrate how the individuals are related. Complete data in the dissimilarity matrix was not given. Instead, six different intervals were used, representing the values 0, .09–.48, .49–.88, .89–1.28, 1.29–1.68 and 1.69–2.08.

Since dissimilarity is being measured, this is a maximization problem. In order to test this placement algorithm on Sokal's data, the 6 intervals were quantized with the values 0, .3, .7, 1.1, 1.5, and 1.9. The eigenvectors E_{27}, E_{26}, E_{25} and E_{24} are given below.

TABLE 6.2

Eigenvectors for Sokal Problem

OTU	E27	E26	E25	E24
1	-0.0196284	-0.1299278	-0.2100324	0.2194112
2	-0.0174247	0.0079917	-0.1291837	-0.0143678
3	-0.0526643	0.0083841	-0.1268731	0.0013172
4	-0.0270066	0.0780911	-0.1957602	0.1855876
5	0.0615105	-0.0781528	-0.1544196	-0.3392692
6	-0.0609313	0.2313899	0.2634283	-0.0609143
7	-0.1752974	0.1088812	-0.3299048	0.4549611
8	0.0683077	-0.3481232	0.2492595	0.0473912
9	-0.0163202	0.2641273	0.2327649	0.0386984
10	-0.0153026	-0.0260781	-0.0978453	-0.2366427
11	-0.0384865	0.0152113	-0.1135472	-0.0019631
12	-0.0251980	0.1255451	-0.2313861	0.1865790
13	0.9100240	-0.0595143	-0.1439505	-0.2789607
14	0.0738811	-0.3426431	-0.0258946	0.0992212
15	-0.0153034	0.2720411	0.2676228	0.0000965
16	-0.0695177	0.0152115	-0.1135479	-0.0019636
17	-0.0376632	-0.0838335	-0.1691312	-0.5065462
18	0.0067246	0.1262516	-0.2265009	0.1684259
19	-0.1714458	0.1936861	0.1841411	-0.0356510
20	-0.0473349	-0.3091024	0.1082395	0.1780271
21	0.0769080	0.0807679	-0.1657397	0.1395029
22	-0.1861492	0.2762499	0.2731388	0.0037467
23	-0.1275033	-0.3696333	0.1841339	0.1973473
24	-0.0152448	-0.2019032	0.2215934	0.0420966
25	-0.0149359	0.0154212	-0.1130940	-0.0021011
26	0.0659477	-0.0307920	-0.0886206	-0.2716177
27	-0.1299278	0.2405394	0.2216874	-0.0482839
LAMBDA=(	45.7771912	44.3530121	42.9558411	40.8111115)

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

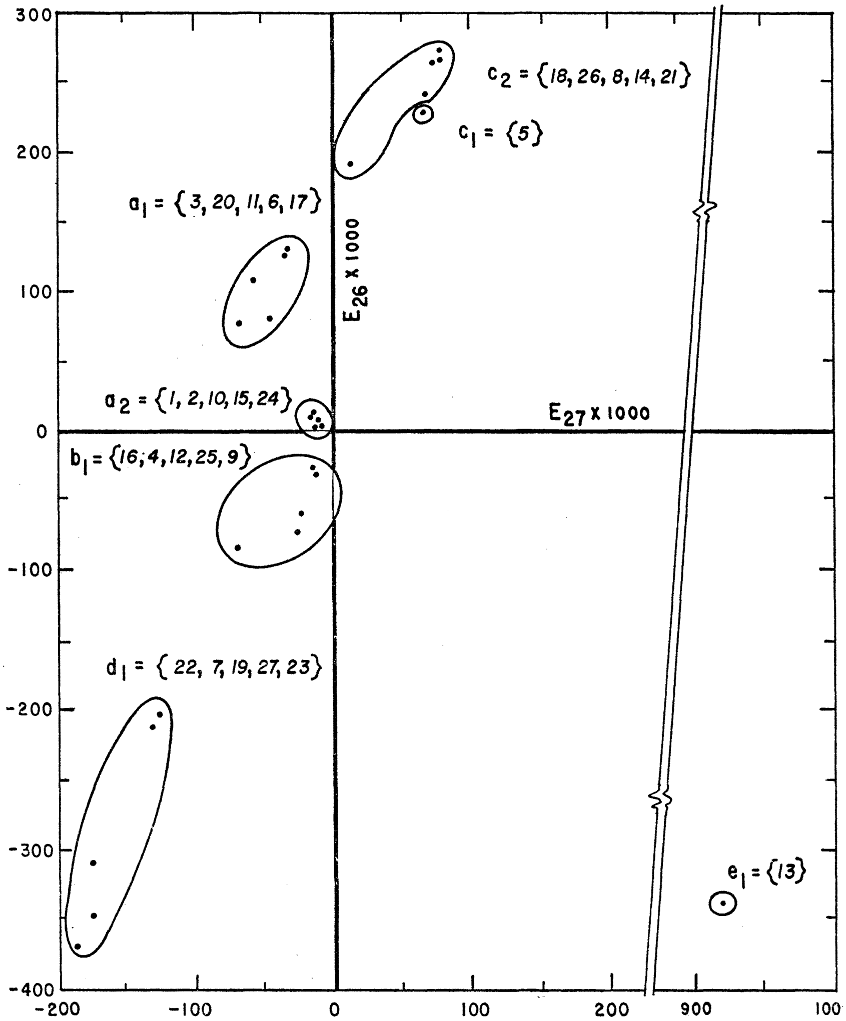


FIGURE 6.10. Graph of Sokal's Clusters

When the above data is plotted in 2 dimensions, 6 distinct clusters appear. Sokal originally defined 7 clusters, which are shown in Figure 6.10. This data indicates that c_1 and c_2 should be combined into one cluster; however, this data is not really accurate since the original data was quantized.

7. Summary

An algorithm has been described for solving quadratic placement problems in r-dimensions. Sums of quadratic forms are either maximized or minimized, depending on the nature of the problem, to yield an optimum solution in any number of dimensions. The r solution (or coordinate) vectors are simple to obtain because they are eigenvectors of a positive semi-definite disconnection matrix B, which is easily constructed from a basic connection matrix C. The n nodes of the graph (i.e., the items which must be positioned) can then be placed at the derived locations.

The solution vectors seem to do a good job separating nodes into local clusters. Therefore, this algorithm may serve as a basis for cluster identification (or separation) problems. Also, it may serve as a basis for mapping problems, where the analog posi-

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms

tions of the nodes must be mapped into discrete locations (perhaps with minimum squared motion).

Solution times are quite fast. On the IBM 360/44, all 34 eigenvectors and eigenvalues were generated for the Steinberg problem in 27 seconds and all 27 eigenvectors and eigenvalues for the Sokal problem were generated in 17 seconds. The Jacobi method (Ralston and Wilf, 1964) was used for generating the eigenvalues and eigenvectors. The time required for N nodes is approximately $10(2v + u)N^3$ where v is the addition time and u is the multiplication time of the computer.

The 1-dimensional quadratic form $X'BX$ leads quite naturally into a quadratic programming problem if it is desired to drop the quadratic constraint $X'X = 1$ and replace it with a set of linear inequality constraints, $AX \leqq b$. This is, in fact, the method one may use for mapping nodes to discrete locations. Thus, after solving the eigenvector problem, some nodes can be assigned positions (e.g., around the border of a rectangular grid) and then linear constraints (e.g., $X_1 - X_2 \leqq 5$, $X_3 \geqq 6$) can be imposed to find the remaining positions.

This problem can be reduced to solving a set of linear equations if all the constraints are given in the form of linear equalities. Kodres (1959) originally pointed this out for the 2-dimensional case. He made use of some specified x_i and y_i values to insure a nontrivial solution of the linear equations for the remaining coordinates. When some coordinates can be specified in advance, his method is very useful. It has been the author's experience, however, that quite often not enough is known about the problem to force such constraints. It is for this very reason that the methods of this paper were developed.

The norm for this problem was chosen because of its mathematical tractability. Extensions to other norms have not been considered here. Unfortunately, no formal link has been found between the Quadratic model of this paper and the Quadratic assignment problem. In particular, to obtain Figure 6.8, the bottom of the grid was first filled out with nodes having negative y-coordinates. The upper part of the grid was filled out next. Because of the way this mapping was done, the largest distortions seem to appear at the top of the grid. This points out that the shape of the grid has a great deal to do with the mapping.

References

1. Armour, G. C. and Buffa, E. S., "A Heuristic Algorithm and Simulation Approach to Relative Location of Facilities," *Management Science*, Vol. 9, No. 2 (January 1963), pp. 294–309.
2. Gilmore, P. C., "Optimal and Suboptimal Algorithms for the Quadratic Assignment Problem," *Journal of the Society for Industrial and Applied Mathematics*, Vol. 10, No. 2 (June 1962), pp. 305–313.
3. Graves, G. W. and Whinston, A. B., "An Algorithm for the Quadratic Assignment Problem," Working Paper No. 110, Western Management Science Institute, University of California, Los Angeles, November 1966.
4. Hillier, F. S. and Conners, M. M., "Quadratic Assignment Algorithms and the Location of Indivisible Facilities," *Management Science*, Vol. 13, No. 1 (September 1966), pp. 42–57.
5. Kodres, U. R., "Geometrical Positioning of Circuit Elements in a Computer," Conference Paper 1172, AIEE Fall General Meeting, October 1959.
6. Lawler, E. L., "The Quadratic Assignment Problem," *Management Science*, Vol. 9, No. 4 (July 1963), pp. 586–599.
7. Ralston, A. and Wilf, S., *Mathematical Methods for Digital Computers*, Vol. 1, John Wiley & Sons, New York, 1964, pp. 84–91.
8. Sokal, R. R., "Numerical Taxonomy," *Scientific American*, Vol. 215, No. 6 (December 1966), pp. 106–116.
9. Steinberg, L., "The Backboard Wiring Problem: A Placement Algorithm," *Society for Industrial and Applied Mathematics Review*, Vol. 3, No. 1 (January 1961), pp. 37–50.

This content downloaded from
132.239.10.108 on Tue, 24 Oct 2023 21:06:55 +00:00
All use subject to https://about.jstor.org/terms