European Journal of Operational Research 57 (1992) 253–270
North-Holland

A distance assignment approach to the facility layout problem

Meir J. Rosenblatt

Faculty of Industrial Engineering and Management, The Technion – Israel Institute of Technology, 32000 Haifa, Israel, and Olin School of Business, Washington University, St. Louis, MO 63130, USA

Boaz Golany

Faculty of Industrial Engineering and Management. The Technion – Israel Institute of Technology, 32000 Haifa, Israel

Received June 1990; revised December 1990

Abstract: The facility layout problem has generally been formulated as a quadratic assignment problem, where the interacting facilities are assigned to the various sites. In this paper a different approach, based on assigning distances between pairs of sites to pairs of facilities, is taken. A network formulation and analysis of this problem is provided. An algorithm, composed of three phases, is developed. The first phase involves preprocessing of the distance matrix. The second phase involves solving the (generalized assignment) network problem. The third phase employs a heuristic procedure which transforms the solution of the second phase into a feasible assignment of facilities to sites. Finally, computational experience with the new algorithm is provided.

Keywords: Facility layout, quadratic assignment problems, networks, heuristic algorithms

Introduction

The facility layout problem deals with the physical arrangement of departments or machines within a given configuration. The importance of the subject and its impact on the performance of a firm is stated by Tompkins and White (1984) who claim that between 20–50% of the total operating expenses within manufacturing are attributed to material handling. As a result of these findings, many studies were initiated to explore means to minimize material handling costs. The general treatment of the problem assumes a static and deterministic environment where actual flow among the various departments is known. Extensions of the model to the stochastic case are provided in Rosenblatt and Lee (1987) and Rosenblatt and Kropp (1990), and an extension to the dynamic deterministic case is given in Rosenblatt (1986).

The facility layout problem has traditionally been formulated as a Quadratic Assignment Problem (QAP). This formulation, originally suggested by Koopmans and Beckmann (1957), assigns n facilities to n mutually exclusive sites (locations). The layout problem considered here assumes that each facility consumes the same area, and so any facility can be assigned to any site. The objective function involves minimizing total material handling cost (i.e. cost incurred by the flow of material among facilities).

The overall minimum material handling cost is usually modeled by minimizing the product of interdepartmental flow and distance. A variety of QAP formulations have been suggested, some of which include the fixed cost of assigning facilities to sites. Following Lawler (1963), Pierce and Crowston (1971), and Rosenblatt (1979), the QAP formulation is given by: (QAP)

$$\min \quad z = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{m=1}^{n} A_{ijkm} x_{ij} x_{km}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, \ldots, n,$$

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, n,$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, \ldots, n$$

where

$$x_{ij} = \begin{cases} 1, & \text{if facility } i \text{ is assigned to site } j, \\ 0, & \text{otherwise,} \end{cases}$$

c_{ij} = fixed (location) cost associated with assigning facility i to site j,
d_{jm} = distance between sites j and m (proportional to travel cost), $d_{jj} = 0$,
f_{ik} = work flow between facilities i and k, $f_{ii} = 0$,

$$A_{ijkm} = \begin{cases} f_{ik} \cdot d_{jm} & \text{if } i \neq k \text{ or } j \neq m, \\ c_{ij} & \text{if } i = k \text{ and } j = m, \\ \infty & \text{if } i = k \text{ and } j \neq m. \end{cases}$$

Since the early 1960s, the QAP has been applied to a variety of other similar problems, such as control board layouts, typewriter keyboards, etc. The QAP has also been applied to the multiobjective (sometimes conflicting) formulation of the plant layout problem (Rosenblatt, 1979; Fortenberry and Cox, 1985; Rosenblatt and Sinuany-Stern, 1986; Urban, 1987). These formulations incorporate 'closeness' ratings and material handling cost into the QAP formulation.

Various approaches for solving the QAP formulation are discussed in the next section. A different approach is suggested in this paper which involves assigning distances between sites to pairs of departments. A network representation of the distance assignment problem is presented, followed by a description of the three phases of the algorithm. Applying just the first two phases of the algorithm provides a lower bound on the solution. An illustrative example is then provided to demonstrate the algorithm's steps. Finally, computational experience with the algorithm is reported, along with some general conclusions and suggestions for further research.

Preliminaries

Several optimal procedures (e.g. Gilmore, 1962; Lawler, 1963; Gavett and Plyter, 1966) were developed for the QAP formulation. These approaches are basically derivatives of branch-and-bound techniques and proved to be impractical for all but small-sized problems. The computational difficulty in solving this problem motivated the development of many heuristic algorithms, (e.g., see Armour and Buffa, 1963; Hillier, 1963; Hillier and Connors, 1966; Vollmann, Nugent and Zartler, 1968; Heider, 1973; Drezner, 1980). For reviews of QAP solution techniques, the reader is referred to Francis and White (1974) and Burkard and Startmann (1978).

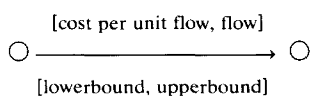
Another line of thought, and one related to the algorithm developed in this paper, is the attempt to linearize the QAP by introducing additional variables and constraints. This approach was suggested, to name but a few, by Beale and Tomlin (1972), Murty (1976, p. 413), Kaufman and Broeckx (1978), Bazaraa and Sherali (1980), and Adams and Sherali (1986). This approach reduces the difficulty of solving a quadratic program by transforming it into an integer linear-programming formulation. However, all of these attempts resulted in methods which are computationally intractable.

Another approach to model this problem relies on definitions and techniques taken from graph theory. In work reported by Foulds (1983), Mountriel et al. (1987) and others, heuristic algorithms were proposed to construct planar graphs representing the adjacency relations as desired by the different departments. Unlike the problem considered here, their analysis is concerned only with the relationship between adjacent (neighboring) facilities, and thus distances are unimportant (similar to the ALDEP approach, see Seehof and Evans, 1967). An interesting feature of the work of Montreuil et al. (1987) is the introduction of the b-matching procedure to maximize the weighted (in terms of segments of the perimeter) adjacencies. Also, in another development, Montreuil and Ratliff (1989) describe the inter-departmental flow in terms of a connected graph, and proceed by constructing a special spanning (cut) tree. However, both procedures do not guarantee optimality, and heuristic (sometimes interactive) approaches are used to solve the layout problem.

The approach adopted here may be viewed as a hybrid of optimal and heuristic methods. The 'optimal' method constructs a network formulation which assigns the available distances to pairs of departments. The heuristic technique uses a branch-and-bound framework to amend the optimal distance assignment into a feasible department's assignment. Thus, instead of assigning facilities to sites, the focus here is on assigning distances between sites to pairs of facilities. This idea was briefly mentioned by Tompkins and White (1984, p. 504), and Gavett (1968, p. 213), but to the best of our knowledge, has never been developed into a solution scheme.

Network representation of the distance assignment problem

Interesting insights into the QAP formulation can be obtained by formulating the problem of assigning distances to pairs of departments as a (minimum-cost) network problem. The flow in such a network determines the distance between each pair of departments. The network representation allows one to take advantage of network theory and algorithms. In particular, the strong property of uni-modularity (i.e. all-integer solutions) in the context of network problems is employed here to guarantee the desired integer solution of the Distance Assignment Problem (DAP). The following notation is used for the network representation:



and we define

$$x_{ikd} = \begin{cases} 1, & \text{if distance between facilities } i, k \text{ is } d, \\ 0, & \text{otherwise.} \end{cases}$$

Figure 1 is a graphical illustration of a network which corresponds to a layout with eight equal-sized sites arranged in a 2-by-4 configuration. (For lack of space, only part of the network appears in Figure 1 with flow parameters and bounds written only in the top portion.) The flow in this network represents the type of distance which is assigned to pairs of departments. The incoming flow, 7 units for each department, implies that it should be connected to 7 other departments. The arc capacities [0,1] on the *department-to-pair* arcs simply allocate one type of distance to each pair of departments (i, k). The arc capacities [0,1] on the *pair-to-distance* arcs mean that the distance between any pair (a decision variable) must be one (and only one) of the distances listed as 'depots'. The outgoing flow from the *destination* nodes restricts the total number of distances of any length d to be equal to a parameter determined by the layout configuration. Thus, e.g. in the 2×4 configuration, there are 20 distances of length 1. The costs associated with the *pair-to-distance* arcs represent material handling costs. The values for these figures are based on the distances with which their respective arcs are associated. Thus, if flow is channeled through an arc associated with a distance of length 2, the cost will be twice of that associated with a distance of length 1. The capacities on the *distance-to-destination* arcs restrict the total number of

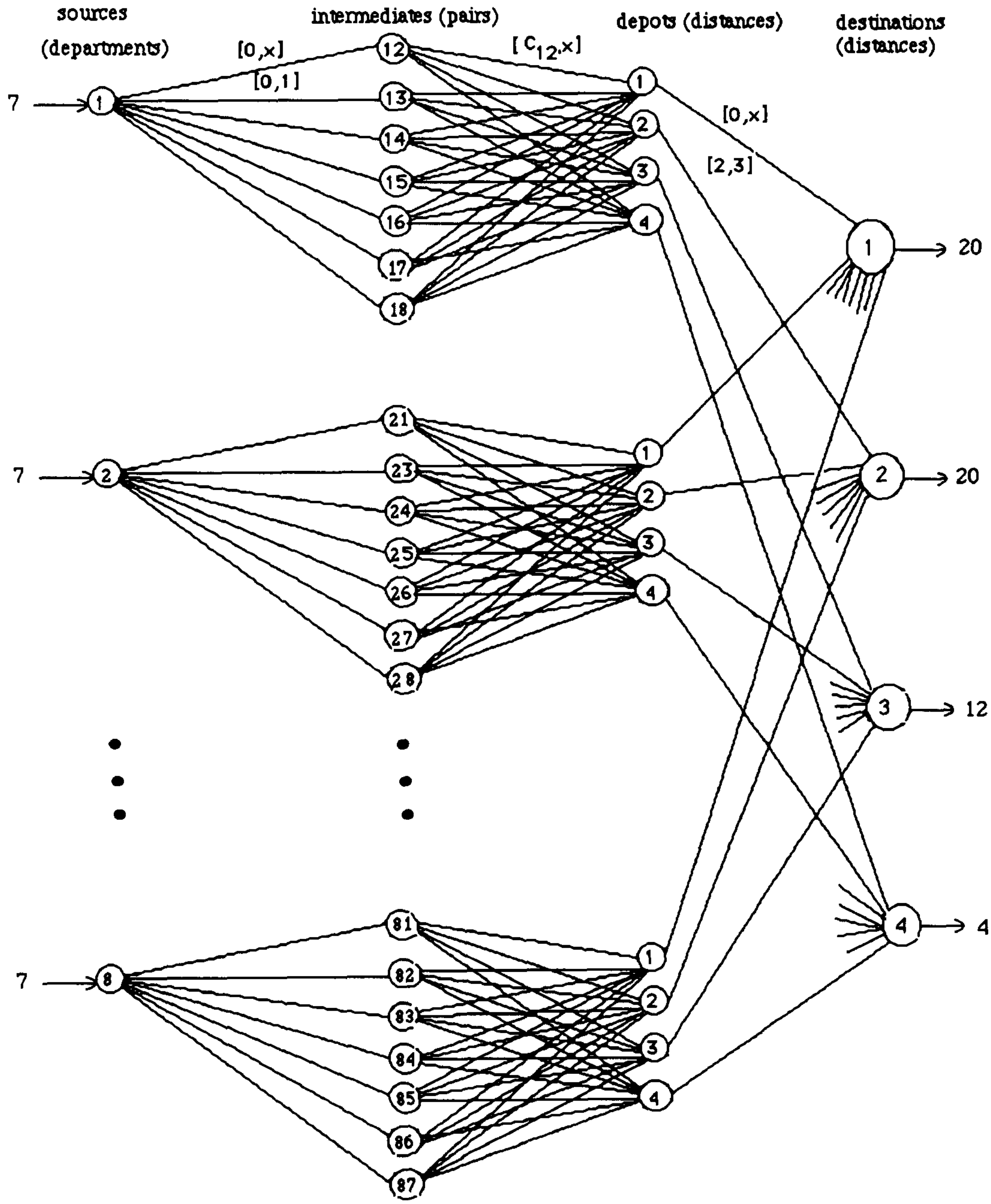


Figure 1. Network representation

distances of any length d measured from one department to be bounded from above and below by parameters which are determined by the layout configuration. In the 2×4 example there should be at least 2 and no more than 3 distances of length 1 measured from any department (site). While in the graphs described by Foulds (1983), each node represents a physical department, here, some of the nodes represent abstract notions of *pairs* and *distances*.

The network depicted in Figure 1 differs from the QAP in two aspects. First, while in the QAP it is implicitly assumed that $x_{ikd} = x_{kid}$ for every pair (i, k) and every distance d, the network does not include such a restriction. The latter constraints can be added to the formulation turning it into a network with side constraints. The benefit from the additional constraints – stronger lower bounds – should be compared with the disadvantages caused by the side constraints – loss of the pure network properties (including the integrality property) and significant increase in computational effort. In this study it was decided not to employ these side constraints.

Second, this network formulation does not represent all the 'structural' constraints required to obtain a feasible assignment of departments to sites. For example, it accumulates only the 'forward flow' (from i to k where $i < k$) but it does not restrict the 'backward flow' from k to i. Again, adding such structural (side) constraints is possible, but, at the cost discussed above.

The algorithm

The algorithm is divided into three phases. In the first phase, the distance matrix is analyzed and distance parameters depending on the specific configuration of the layout are determined. In the second phase, the network model, formulated as a generalized assignment problem is solved and optimal distances between the various pairs of facilities are obtained. In the last phase, the 'optimal distance allocation' is transformed into an assignment of facilities to sites. A heuristic algorithm is used in this phase since there is no guarantee that there exists a consistent (feasible) transformation of the distance assignment into a facilities-to-sites assignment.

Throughout this study the distance between sites is measured from the center of each site, assuming a rectilinear distance. This is justified by assuming rectangular shape of a site. In this study it is further assumed that each site is a square with the same area. This last assumption can be relaxed to make the algorithm more applicable to practical situations, see Hillier and Connors (1966).

Phase 1: Preprocessing the distance matrix

For each possible distance d, $d = 1, \dots, D$, the following measures are computed:
t_d = Number of times distance d is measured.
u_d = Maximum number of times distance d is measured from any site.
v_d = Minimum number of times distance d is measured from any site.

As an illustration, consider again the 2×4 plant structure described earlier. It is assumed that the width of each site (squared-shaped) is one unit, and the numbers represent site identification numbers.

1	2	3	4
5	6	7	8

The distance matrix for this example is given in Table 1 (only the upper triangular portion is given due to symmetry).

Table 1
The distance matrix

Site	Site 1	2	3	4	5	6	7	8
1	0	1	2	3	1	2	3	4
2		0	1	2	2	1	2	3
3			0	1	3	2	1	2
4				0	4	3	2	1
5					0	1	2	3
6						0	1	2
7							0	1
8								0

Table 2
Preprocessing the distance matrix

d	t_d	u_d	v_d
1	20	3	2
2	20	3	2
3	12	2	1
4	4	1	0

Table 2 presents the values of t_d, u_d and v_d for each distance d.

The maximum distance from any site is 4, i.e. $D = 4$. In order to further understand Table 2, the first line is explained as:

'There are 20 distances of length 1 (notice that t_d is based on twice the count in the upper half of the matrix in Table 1); the maximum number of distances of one unit length measured from a single site is 3 and the minimum is 2.

Finally, the output of this preprocessing stage (t_d, u_d, v_d) can easily be generated using the site-to-site distance matrix in $O(n^2)$-effort.

Phase 2: Solving the network problem

In this phase the n facilities define $n(n-1)$ distances which need to be assigned into D categories. A (network) integer linear formulation of the distance assignment problem (DAP) is solved.

$$\min \quad z = \sum_{i=1}^{n} \sum_{\substack{k=1 \\ k \neq i}}^{n} f_{ik} \sum_{d=1}^{D} d \cdot x_{ikd} \tag{1}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} \sum_{\substack{k=1 \\ k \neq i}}^{n} x_{ikd} = t_d, \quad d = 1, \ldots, D, \tag{2}$$

$$\sum_{d=1}^{D} x_{ikd} = 1, \quad i, k = 1, \ldots, n, \ k \neq i, \tag{3}$$

$$v_d \leq \sum_{k=1}^{n} x_{ikd} \leq u_d, \quad i = 1, \ldots, n, \ i \neq k, \ d = 1, \ldots, D, \tag{4}$$

$$x_{ikd} \in \{0, 1\}, \quad i, k = 1, \ldots, n, \ k \neq i, \ d = 1, \ldots, D. \tag{5}$$

The objective function minimizes the total material handling costs as measured by the product of the inter-departmental flow and the distance between departments. Constraint (2) guarantees that exactly t_d distances of type d for all pairs (i, k) will result. This corresponds to the outgoing flow from the *destination* nodes in the network. Constraint (3) restricts each pair of departments to be assigned to exactly one type of distance. This accounts for the flow between the *pair-to-distance* nodes. Constraint (4) puts an upper and lower limits on distances of type d, measured from any department, at the values of u_d and v_d, respectively. This corresponds to the arc capacities on the *distance-to-destination* nodes. Finally, the zero–one restriction on the decision variables corresponds to the arc capacities on the *pair-to-distance* nodes in the network.

Due to the integrality property, it is possible to relax the zero–one restriction on the x_{ikd} variables in (5), solve the DAP by any LP code and integer solutions are guaranteed. (A similar approach was taken by Montreuil et al. (1987) to solve a different network representation of the QAP.)

Phase 3

This phase is composed of two parts. In the first part, an initial assignment of a department to a site is determined. The second part uses a branch-and-bound procedure where each node corresponds to an assignment of an additional department to a site.

Part I. Assignment of the initial department

1. Construct a matrix $S = (s_{jd})$ whose rows (j) correspond to sites, and columns (d) correspond to distances, where s_{jd} = two times the number of sites with distance d from site j (see Table 6). (Each distance is counted twice, once from j to m, and once from m to j.)

2. Construct a matrix $G = (g_{kd})$ whose rows correspond to departments, while its columns correspond to the distance assignment in the optimal solution of DAP, i.e.

$$g_{kd} = \sum_{\substack{i=1 \\ i \neq k}}^{n} x_{ikd} + \sum_{\substack{i=1 \\ i \neq k}}^{n} x_{kid}.$$

3. Construct a matrix $A = (a_{jk})$ whose entries are the sum of squared differences between the respective entries in matrices G and S, i.e. a_{jk} represents how 'close' is department k to site j across all relevant distances:

$$a_{jk} = \sum_{d=1}^{D} (s_{jd} - g_{kd})^2.$$

4. Choose the entry a_{jk} with the smallest value as the 'core' site and the appropriate department as the initial department to be assigned. In case of a tie, consider all the entries with the minimal value as candidates for initial assignment ('core' sites) in the branch-and-bound tree.

Part II. The branch-and-bound procedure

1. *Ordering the sites for arrangement*

– For each site, its 'centrality' measure σ_j is defined as the sum of distances to all other sites, i.e.

$$\sigma_j = \sum_{d=1}^{D} d \cdot s_{jd}.$$

– For the 'core' site selected in Part I, define the following sets: I_d = {All sites with a distance d from the 'core' site}.

Thus, in the above 2×4 configuration, if, for example, site 1 is selected first, then

$$I_1 = \{2, 5\}, \quad I_2 = \{3, 6\}, \quad I_3 = \{4, 7\} \text{ and } I_4 = \{8\}.$$

– Order the sites according to increasing order of the sets I_d, where the internal order within each set is determined by increasing order of σ_j. Ties are broken by sequencing first sites which are closer to sites which have already been sequenced. In the example above, starting with site 1 determines the following assignment order of sites: {1, 2, 5, 6, 3, 7, 4, 8}.

2. *Assigning a department to a selected site*

– Let Ω be the set of departments which have already been assigned. Given the current site to be assigned, compute the penalties associated with assigning each candidate department i ($i \notin \Omega$) to this site. This penalty cost, r_{ik}, is the sum of the 'reduced cost' associated with the variables x_{ikd} and x_{kid} (obtained from the optimal solution to DAP) where d is the distance between department k ($k \in \Omega$) and the candidate department i.

– For each candidate department i compute the sum of its penalties, P_i, where

$$P_i = \sum_{k \in \Omega} r_{ik}.$$

– For each department $k \in \Omega$, check whether the assignment of department i into the current site will exhaust its distances of length d (where d is the distance from k to the current site). If this happens, impose a 'pseudo' penalty, r'_{ik}, which is the sum of the reduced cost associated with the variables x_{ikl} and x_{kil} in DAP, where l is the next available distance between the pair (i, k). The value of l is determined by first checking whether the distance $l = d + 1$ is available, then $l = d + 2$, etc. If no distance $d < l \leq D$ is available, then the search continues by the sequence $l = 1, 2, \ldots, d - 1$.

– For each candidate department i, compute the sum of its pseudo-penalties, R_i, where

$$R_i = \sum_{k \in \Omega} r'_{ik}.$$

– Select the department with the minimal 'modified' penalty, MP_i, where $\text{MP}_i = P_i - R_i$. This department is assigned to the currently available site. Note that MP_i can be negative.

3. *Branching*

– Branching occurs in two cases of ties:

1) More than one department can be assigned first, i.e. tie in the minimal values of a_{jk}.

2) There are two or more departments with the same minimal value of MP_i at a given iteration.

– Branching is done according to the newest-bound rule, i.e. the most recently created subproblem is the one selected for branching.

3a. *Optional branching rule*

Assume that the last assigned department is q and its location is p, i.e. $x_{qp} = 1$. Try to exchange department q with each of the departments in Ω. The exchange which yields the best (positive) cost savings is adopted as a new node in the branch-and-bound tree. The cost savings is calculated as follows. Each $i \in \Omega$ is located in site $j(i)$, i.e. $x_{i,j(i)} = 1$. Then, exchange department q with department k^* $(k^* \in \Omega)$ if $\Delta_{k^*} > 0$ where:

$$\Delta_k = \sum_{i \in \Omega} f_{ik}\left[g_{j(i),j(k)} - g_{j(i),p}\right] + f_{qi}\left[g_{p,j(i)} - g_{j(k),j(i)}\right],$$

$$\Delta_{k^*} = \max_{k \in \Omega} \Delta_k.$$

4. *Bounding*

At every iteration of the branch-and-bound procedure one department is assigned to a site. The cost of every partial assignment serves as a lower bound for its relevant branch in the tree.

5. *Fathoming*

Fathoming occurs when the current lower bound is greater than or equal to the current upper bound (incumbent solution).

Discussion

It was observed that the first assignment of a facility to a site has a major impact on the overall performance of the algorithm. The first part of Phase 3 is intended to address this issue by creating an 'assignment matrix', A. This matrix is constructed by comparing the actual number of distances from each site to all other sites with the optimal allocation of distances to departments. The algorithm attempts to find one or more departments which can be viewed as natural candidates to be assigned to their respective sites. If no 'perfect match' of a department to a site exists, i.e. $a_{jk} \neq 0$ for all 'department-to-site' pairs, the pair selected for the initial assignment is the one with the smallest sum of squared errors. The reason for using the sum of squared errors (rather than other alternatives such as the least absolute value) is to penalize for large deviations (in a similar manner to its use in forecasting). Ties may occur whenever there are several departments/sites with the same minimal value. In such cases, all the possible initial assignments are explored as 'roots' in the branch-and-bound tree. However, due to the symmetry of various site configurations, there is no need to explore all the alternatives of assigning departments into symmetric sites.

It should be noted that this technique does not guarantee optimality, i.e. the initial assignment may be misleading in some cases. Since the algorithm is not repetitive, namely, once a department is assigned to a site it remains there for all further nodes on the same branch of the tree, a mistake in the initial assignment may be crucial to obtaining optimality.

The algorithm uses a branch-and-bound scheme where in each node a department is assigned to a site. The sequence in which sites in the layout are selected is according to ever-increasing 'circles' starting with the first ('core') site selected for assignment. This idea is somewhat similar to the CORELAP procedure for solving plant layout problems (see Lee and Moore, 1967, and Moore, 1971).

The assignment of facilities to sites is based on the reduced costs of the DAP solution which serve here as penalties. If the distance between an assigned department and a candidate department is awarded a positive value by DAP (i.e. it is a basic variable), its reduced cost would be zero and no penalty is added. If, however, another distance is selected by DAP as optimal, then forcing the current distance between these departments to that level amounts to making the corresponding variables basic, with the associated increase in the objective value given by the reduced costs. Two types of penalties are considered. The first, P_i, is a result of the actual assignment of a potential department to a site. The other is an estimation of additional penalties which would occur in the next steps as a result of the current assignment. This 'pseudo-penalty', R_i, is basically a 'look-ahead' feature added to the heuristic to prevent it from making the wrong assignment by ignoring information available only one step ahead. For each department, the value P_i is the sum of penalties with respect to all departments previously assigned.

The idea of the pseudo-penalty is slightly more complicated. Consider, for example, the case where, by assigning a potential department i to a site which is adjacent (i.e. $d = 1$) to an assigned department k, we exhaust all sites with distance $d = 1$ from department k. Then, if we do not assign department i to that site, its minimum distance from k will be 2 units of distance. Therefore, the sum of the reduced costs associated with x_{ik2} and x_{ki2} is taken as a 'pseudo'-penalty to be subtracted from the 'real' penalty computed for i. If, however, by assigning department i to a site with a distance d from an assigned department k, we do not exhaust all sites with the same distance d from department k, then the value of the pseudo-penalty, r'_{ik}, is zero. The same procedure needs to be applied to all departments previously assigned. Thus, the combined penalty cost MP_i associated with assigning a potential department i is defined as the difference between the penalty P_i, and the pseudo penalty R_i, namely, $\mathrm{MP}_i = P_i - R_i$. The department with the minimal value of MP_i is chosen as the next department to be assigned. It should be noted that other, more complicated, 'look-ahead' schemes may be applied at this stage. However, as these schemes become more sophisticated the computational effort grows drastically.

Step 3a of the branching procedure is introduced in order to overcome the 'irreversibility' nature of the assignment process. Without it, each time a department is assigned to a site, it remains in the same location throughout the relevant branch of the tree. As the assignment procedure is myopic, and obviously does not guarantee optimality, an exchange between assigned facilities provides an avenue for improvement.

The bound on each node is obtained by adding the penalty (ignoring the value of the pseudo-penalty) of the current selected pair (department–site assignment) to the previous bound. The pseudo-penalty does not play a role in determining bounds because it is only an estimate of the cost to be incurred in later assignment stages, and cannot be used as a valid penalty.

A numerical example

Data for this example is taken from Nugent et al. (1968). The three phases of the algorithm are demonstrated below for the 2×4 structure described earlier with the symmetric flow matrix, (f_{ik}), given in Table 3.

Table 3
The flow matrix

Department	Department							
	1	2	3	4	5	6	7	8
1	0							
2	5	0						
3	2	3	0					
4	4	0	0	0				
5	1	2	0	5	0			
6	0	2	0	2	10	0		
7	0	2	0	2	0	5	0	
8	6	0	5	10	0	1	10	0

Phase 1. The information given in Table 2 is generated.

Phase 2. The following DAP is formulated and solved.

$$\begin{aligned}
\min \quad & Z = 5(x_{121} + 2x_{122} + 3x_{123} + 4x_{124}) \\
& \quad + 5(x_{211} + 2x_{212} + 3x_{213} + 4x_{214}) + 2(x_{131} + 2x_{132} + 3x_{133} + 4x_{134}) \\
& \quad + 2(x_{311} + 2x_{312} + 3x_{313} + 4x_{314}) + \cdots + 10(x_{781} + 2x_{782} + 3x_{783} + 4x_{784}) \\
& \quad + 10(x_{871} + 2x_{872} + 3x_{873} + 4x_{874}) \\
\text{s.t.} \quad & x_{121} + x_{211} + x_{131} + x_{311} + \cdots + x_{781} + x_{871} = 20, \\
& x_{122} + x_{212} + x_{132} + x_{312} + \cdots + x_{782} + x_{872} = 20, \\
& x_{123} + x_{213} + x_{133} + x_{313} + \cdots + x_{783} + x_{873} = 12, \\
& x_{124} + x_{214} + x_{134} + x_{314} + \cdots + x_{784} + x_{874} = 4, \\
& x_{121} + x_{122} + x_{123} + x_{124} = 1, \\
& x_{131} + x_{132} + x_{133} + x_{134} = 1, \\
& \quad \vdots \\
& x_{181} + x_{182} + x_{183} + x_{184} = 1, \\
& x_{211} + x_{212} + x_{213} + x_{214} = 1, \\
& x_{231} + x_{232} + x_{233} + x_{234} = 1, \\
& \quad \vdots \\
& x_{281} + x_{282} + x_{283} + x_{284} = 1, \\
& \quad \vdots \\
& x_{811} + x_{812} + x_{813} + x_{814} = 1, \\
& x_{821} + x_{822} + x_{823} + x_{824} = 1, \\
& \quad \vdots \\
& x_{871} + x_{872} + x_{873} + x_{874} = 1, \\
& 2 \le x_{121} + x_{131} + \cdots + x_{181} \le 3, \\
& 2 \le x_{211} + x_{231} + \cdots + x_{281} \le 3, \\
& \quad \vdots
\end{aligned}$$

Table 4
Optimal distance allocation values

Department	Department							
	1	2	3	4	5	6	7	8
1	–	1	2	1	2	3	3	1
2	1	–	1	3	2	1	2	3
3	2	1	–	3	2	3	4	1
4	1	2	3	–	1	2	2	1
5	2	2	4	1	–	1	2	3
6	3	2	4	2	1	–	1	2
7	3	2	3	2	4	1	–	1
8	1	2	2	1	3	2	1	–

$$2 \le x_{811} + x_{821} + \cdots + x_{871} \le 3,$$
$$2 \le x_{122} + x_{132} + \cdots + x_{182} \le 3,$$
$$2 \le x_{212} + x_{232} + \cdots + x_{282} \le 3,$$
$$\vdots$$
$$2 \le x_{812} + x_{822} + \cdots + x_{872} \le 3,$$
$$1 \le x_{123} + x_{133} + \cdots + x_{183} \le 2,$$
$$1 \le x_{213} + x_{233} + \cdots + x_{283} \le 2,$$
$$\vdots$$
$$1 \le x_{813} + x_{823} + \cdots + x_{873} \le 2,$$
$$0 \le x_{124} + x_{134} + \cdots + x_{184} \le 1,$$
$$0 \le x_{214} + x_{234} + \cdots + x_{284} \le 1,$$
$$\vdots$$
$$0 \le x_{814} + x_{824} + \cdots + x_{874} \le 1,$$
$$x_{ikd} = 0 \text{ or } 1, \quad i, k = 1, \ldots, 8, \ k \ne i, \ d = 1, 2, 3, 4.$$

The optimal solution is $Z^* = 185$. Table 4 gives the optimal distances between pairs of departments and Table 5 the corresponding (optimal) reduced cost values associated with each pair of departments. Each entry in Table 5 gives the sum of reduced costs for X_{ikd} and X_{kid}, respectively.

Phase 3

I. *Initial assignment*. Table 6 presents the number of distances of each type d measured from each site to all other sites ($\{s_{jd}\}$). Table 7 gives the optimal distance allocation to each department. Departments 2, 3, 4 and 7 are candidates to become the first site to be assigned ($\text{Min}\{a_{jk}\} = 2$). (See Part I.3.)
II. *The branch-and-bound procedure*. The sequence of assignments which corresponds to the branch leading to the minimal solution value obtained, without applying the optional branching rule 3a, is demonstrated below. Note that this is only one of the four branches that should be started for this example. At Phase I, department 3 is assigned to site 1.

Next,

3	?		

Candidate dep.(i)	1	2	4	5	6	7	8
$r_{3i}(X_{3i1}, X_{i31})$	0	0	4	4	4	4	0
P_i	0	0	4	4	4	4	0

Table 5
Reduced costs in the optimal solution

Pair	Distance			
	1	2	3	4
1-2	0	6	16	26
1-3	0	0	4	8
1-4	0	4	12	20
1-5	2	0	2	4
1-6	4	0	0	0
1-7	4	0	0	0
1-8	0	5	17	29
2-3	0	2	8	14
2-4	4	0	0	0
2-5	0	0	4	8
2-6	0	0	4	8
2-7	0	0	4	8
2-8	7	0	0	0
3-4	4	0	0	0
3-5	4	0	0	0
3-6	4	0	0	0
3-7	4	0	0	0
3-8	0	3	13	23
4-5	0	6	16	26
4-6	0	0	4	8
4-7	0	0	4	8
4-8	0	13	33	53
5-6	0	16	36	56
5-7	4	0	0	0
5-8	7	0	0	0
6-7	0	6	16	26
6-8	5	0	2	4
7-8	0	13	33	53

Comments:

* There is no pseudo-penalty since dep. 3 has another adjacent site (site 5).
* Three departments tie for the minimal penalty value, so, three branches are opened.
* Branch on 'dep. 8 into site 2'.

The result is:

3	8		
?			

Table 6
Site distance (S)

Site	Distance			
	1	2	3	4
1	4	4	4	2
2	6	6	2	0
3	6	6	2	0
4	4	4	4	2
5	4	4	4	2
6	6	6	2	0
7	6	6	2	0
8	4	4	4	2

Table 7
Optimal distances (G)

Department	Distance			
	1	2	3	4
1	6	4	4	0
2	5	7	2	0
3	3	4	4	3
4	6	5	3	0
5	4	6	2	2
6	5	5	3	1
7	4	5	3	2
8	7	4	3	0

Candidate dep.(i)	1	2	4	5	6	7
$r_{3i}(X_{3i1}, X_{i31})$	0	0	4	4	4	4
$r_{8i}(X_{8i2}, X_{i82})$	5	0	13	0	0	13
P_i	5	0	17	4	4	17
$r'_{3i}(X_{3i2}, X_{i32})$	0	2	0	0	0	0
R_i	0	2	0	0	0	0
MP_i	5	−2	17	4	4	17

Comments:
* A pseudo-penalty associated with X_{3i2} (X_{i32}) is applied, since dep. 3 has no other adjacent sites.
* Dep. 2 has the minimal modified penalty, $MP_2 = -2$.
* Assign 'dep. 2 into site 5'.

This step of the algorithm is repeated two more times before we reach the following situation:

3	8	4	
2	1	?	

Candidate dep.(i)	5	6	7
$r_{3i}(X_{3i3}, X_{i33})$	0	0	0
$r_{8i}(X_{8i2}, X_{i82})$	0	0	13
$r_{2i}(X_{2i2}, X_{i22})$	0	0	0
$r_{1i}(X_{1i1}, X_{i11})$	2	4	4
$r_{4i}(X_{4i1}, X_{i41})$	0	0	0
P_i	2	4	17
$r'_{2i}(X_{2i3}, X_{i23})$	4	4	4
$r'_{1i}(X_{1i2}, X_{i12})$	0	0	0
R_i	4	4	4
MP_i	−2	0	13

Comments:
* Pseudo-penalties associated with X_{2i3}, (X_{i22}) and X_{1i2}, (X_{i12}) are applied since dep. 2 has no more sites at a distance of $d = 2$ and dep. 1 has no other adjacent sites.

* Dep. 5 has the minimal modified penalty, $MP_5 = -2$.
* Assign 'dep. 5 into site 7'.

We get:

3	8	4	?
2	1	5	

Candidate dep.(i)	6	7
$r_{3i}(X_{3i3}, X_{i33})$	0	0
$r_{8i}(X_{8i2}, X_{i82})$	0	13
$r_{2i}(X_{2i4}, X_{i24})$	8	8
$r_{1i}(X_{1i3}, X_{i13})$	0	0
$r_{4i}(X_{4i1}, X_{i41})$	0	0
$r_{5i}(X_{5i2}, X_{i52})$	16	0
P_i	24	21
$r'_{3i}(X_{3i4}, X_{i34})$	0	0
$r'_{8i}(X_{8i3}, X_{i83})$	2	33
$r'_{2i}(X_{2i3}, X_{i23})$	4	4
$r'_{1i}(X_{1i2}, X_{i12})$	0	0
$r'_{4i}(X_{4i2}, X_{i42})$	0	0
$r'_{5i}(X_{5i1}, X_{i51})$	0	4
R_i	6	41
MP_i	18	−20

Comments:

* A pseudo-penalty associated with each of the assigned sites is applied since for all of them the assignment into site 4 exhausts their respective distance from this site.
* Notice that here is the first time where a smaller distance was used when checking for other available distances to apply the pseudo-penalty, e.g. a distance of 3 instead of 4 from dep. 2.
* Dep. 7 has the minimal modified penalty, $MP_7 = -20$.
* assign 'dep. 7 into site 4'.

Finally, therefore, we obtain:

3	8	4	7
2	1	5	6

Comments:

* The assignment of 'dep. 6 into site 8' is by default since this is the only pair left.
* Additional penalty for this assignment can be taken from the pseudo-penalty for dep. 6 in the previous stage. Layout cost = 109.
* The known optimal solution to this problem is 107. The optimal solution is indeed obtained when branching rule 3a is applied (see Table 9).

Computational experience

The set of problems offered by Nugent et al. (1968) became an acceptable benchmark for comparing different solution techniques. This set is selected to demonstrate the performance of the DAP algorithm. The results given below (see Table 9) are compared to eight other algorithms. The first four are taken from the original paper by Nugent et al. (1968), the fifth (FLAC) from Scriabin and Vergin (1985), the

Table 8
Computational results (DAPI)

n	LP solution	No. of iterations	No. of nodes in b&b tree
5	25	48	130
6	41	71	516
7	67	147	165
8	93	171	1022
12	243	485	814
15	479	879	910
20	1015	1917	6103
30	2251	5148	12705

sixth (DISCON) from Drezner (1987) and the last two (FATE and TAA) from Heragu and Kusiak (1988). Two versions of the results are shown, with and without applying the optional branching rule 3a. The computation was performed according to the algorithm's phases. The LP phase was done on a Prime mini-computer and the third phase on an IBM PC.

Since run times are incompatible to other computers we prefer to list the number of simplex iterations each problem took in the second phase, and the number of nodes visited in the branch-and-bound procedure in the third phase (see Table 8). Since at each node the algorithm performs simple tabular operations, the amount of CPU time on a mainframe computer required to complete a tour over the tree, even for the largest problem listed below, would be quite small. It is interesting to note that while the number of simplex iterations increases monotonously with the problem size, the number of nodes visited in the tree changes in both directions. In particular, odd-shaped layouts (e.g. $n = 5$, 7) are associated with relatively few nodes while 'even' layouts – where many sites are symmetric – are associated with many more nodes. This may suggest a relative advantage of this algorithm for such problems over comparable algorithms. Notice in Table 9 that the relative error for the four algorithms checked by Nugent et al. (1968) is larger for the odd-shaped layout problems than for the even layout problems. (Drezner, 1987, does not report results for these odd-shaped layouts.)

In order to compare the effectiveness of the various heuristic solutions, the average deviation (from best known results) of each one of them over the test problems is computed. For reasons explained above, the deviations of the DISCON heuristic are averaged over 6 problems, while for the other heuristics the average is over 8 problems. Table 10 reports the relative effectiveness of the two DAP algorithms with respect to all the compared heuristics.

Table 9
Comparison of the various algorithms

n	Best known	H63 [b]	HC63-66 [b]	CRAFT [b]	Biased sampling [b]	FLAC	DISCON [c]	FATE [c]	TAA	DAP I	DAP II [d]
5	25 [a]	27.6	29.4	28.2	26.8	25		26.0	25	25	25
6	43 [a]	44.2	44.2	44.2	43.6	43	47.5	50.6	43	43	43
7	74 [a]	78.8	78.4	79.6	74.8	74		78.0	74	75	74
8	107 [a]	114.4	110.2	113.4	107.0	107	118.8	126.7	116	109	107
12	289 [a]	317.4	310.2	296.2	293.0	289	322.2	326.2	314	302	291
15	575 [a]	632.6	600.2	606.0	580.2	585	630.8	660.8	596	616	589
20	1285	1400.4	1345.0	1339.0	1313.0	1303	1416.4	1436.3	1414	1404	1343
30	3064	3267.2	3206.8	3189.6	3124.8	3079	3436.4	3390.6	3326	3474	3247

[a] Known optimal values (Burkard, 1984; other values taken from Wilhelm and Ward, 1987).
[b] As reported in Nugent et al. (1968).
[c] Average values as reported by Drezner (1987) and Heragu and Kusiak (1988). In both cases, the best results achieved by the heuristics outperformed DAP. However, these results were achieved by using an improvement phase. Since DAP does not employ such a phase, comparison is made only to the average results of these heuristics.
[d] Including the optional branching rule 3a.

Table 10
Average relative errors for the various algorithms

Algorithm	H63	HC66	CRAFT	Biased sampling	FLAC	DISCON	FATE	TAA	DAP I	DAP II
Average deviation (%)	7.7	6.3	5.8	2.0	0.5	10.8	12.0	4.9	4.7	1.7

Table 11
Layout vectors corresponding to best DAP solutions

n	DAP II	Layout vector
5	25	4,5,1,2,3
6	43	1,2,3,4,5,6
7	74	3,7,2,1,6,5,4
8	107	3,8,7,6,2,1,4,5
12	291	12,9,7,3,4,11,8,1,5,6,10,2
15	589	11,9,13,2,1,8,7,3,14,4,12,5,15,6,10
20	1343	6,7,20,12,13,3,1,8,15,16,10,5,11,19,14,18,17,4,2,9
30	3247	18,22,27,30,20,15,3,23,11,16,14,4,19,8,7,9,29,21,17,25,10,13,28,5,1,24,26,12,6,2

Table 11 gives the layout solutions of DAP II to the 8 test problems. The elements in the layout vectors below are departments assigned to sites corresponding to the position in the vectors. For example, department 4 is assigned to site 1 in the first problem ($n = 5$).

Conclusions

This paper approaches the quadratic assignment problem, formulated for the facilities layout problem, from a different angle. Instead of solving a quadratic assignment problem in which facilities are assigned to sites, a distance assignment problem (DAP) formulation is used. Under the DAP formulation, a given distance (based on plant configuration) is assigned to each pair of facilities. The DAP is solved using an LP code. Then, a heuristic algorithm is applied, resulting in a feasible assignment of facilities to sites. The heuristic approach is based on penalty costs obtained from the reduced costs of the DAP solution. Indeed, the solution achieved using this algorithm is only a heuristic one, although, given our experience, its performance is quite satisfactory as shown by the examples given in the previous section. In addition to the solution procedure developed, the solution of the LP program can serve as a lower bound for the QAP. Also, due to the uni-modularity property, the solution obtained by solving the DAP using LP packages is integer.

Some features that are by-products of the algorithm are: its ability to easily handle nonsymmetric distances and nonrectilinear distances (e.g. Euclidean), and its ability to fix certain facilities to a corner/edge/center of the plant configuration. This is easily done by setting the relevant sum of x_{ikd} to values corresponding to the distance characteristic vector associated with the site. Similar types of restrictions may involve a greater effort in the original QAP formulation.

Acknowledgment

We thank Prof. U. Rothblum for his helpful comments and U. Levi and C. Grines for their help in performing the necessary computations.

References

Adams, W.P., and Sherali, H.D. (1986), "A tight linearization and an algorithm for zero-one quadratic programming problems", *Management Science* 32, 1274–1290.

Armour, G.C., and Buffa, E.S. (1963), "A heuristic algorithm and simulation approach to the relative location of facilities", *Management Science* 9, 294–309.

Bazaraa, M.S., and Sherali, H.D. (1980), "Benders' partitioning scheme applied to a new formulation of the quadratic assignment problem", *Naval Research Logistics Quarterly* 27, 29–41.

Beale, E.M.L., and Tomlin, J.A. (1972), "An integer programming approach to a class of combinatorial problems", *Mathematical Programming* 1, 339–344.

Burkard, R.E. (1984), "Quadratic assignment problems", *European Journal of Operational Research* 15, 283–289.

Burkard, R.E., and Startmann, K.H. (1978), "Numerical investigations on quadratic assignment problem", *Naval Research Logistics Quarterly* 25, 129–148.

Drezner, Z. (1980), "DISCON: A new method for the layout problem", *Operations Research* 28, 1375–1384.

Drezner, Z. (1987), "A heuristic procedure for the layout of a large number of facilities", *Management Science* 33, 907–915.

Fortenberry, J.C., and Cox, J.F. (1985), "Multiple criteria approach to the facilities layout problem", *International Journal of Production Research* 23, 773–781.

Foulds, L.R. (1983), "Techniques for facilities layout: Deciding which pairs of activities should be adjacent", *Management Science* 29, 1414–1426.

Francis, R.L., and White, J.A. (1974), *Facility Layout and Location: An Analytical Approach*, Prentice-Hall, Englewood Cliffs, NJ.

Gavett, J.W. (1968), *Production and Operation Management*, Harcourt, Brace and World, New York.

Gavett, J.W., and Plyter, N.V. (1966), "The optimal assignment of facilities to locations by branch and bound", *Operations Research* 14, 210–232.

Gilmore, P.C. (1962), "Optimal and suboptimal algorithms for the quadratic assignment problem", *Journal of the Society for Industrial and Applied Mathematics* 10, 305–313.

Heider, C.H. (1973), "An *N*-step, 2-variable search algorithm for the component placement problem", *Naval Research Logistics Quarterly* 20, 699–724.

Heragu, S.S., and Kusiak, A. (1988), "Machine layout problem in flexible manufacturing systems", *Operations Research* 36, 258–267.

Hillier, F.S. (1963), "Quantitative tools for plant layout analysis", *Journal of Industrial Engineering* 14, 33–40.

Hillier, F.S., and Connors, M.M. (1966), "Quadratic assignment problem algorithms and the location of indivisible facilities", *Management Science* 13, 42–57.

Kaufman, L., and Broeckx, F. (1978), "An algorithm for the quadratic assignment problem using Benders' decomposition", *European Journal of Operational Research* 2, 204–211.

Koopmans, T.C., and Beckmann, M. (1957), "Assignment problems and the location of economic activities", *Econometrica* 25, 53–76.

Lawler, E.L. (1963), "The quadratic assignment problem", *Management Science* 19, 586–590.

Lee, R.C., and Moore, J.M. (1967), "CORELAP – Computerized relationship layout planning", *Journal of Industrial Engineering* 18, 195–200.

Montreuil, B., and Ratliff, H.D. (1989), "Utilizing cut trees as design skeletons for facility layout", *IIE Transactions* 21, 136–143.

Montreuil, B., Ratliff, H.D., and Goetschalckx, M. (1987), "Matching based interactive facility layout", *IIE Transactions* 19, 271–279.

Moore, J.M. (1971), "Computer program evaluates plant layout alternatives", *Journal of Industrial Engineering* 22/8, 19–25.

Murty, K. (1976), *Linear and Combinatorial Programming*, Wiley, New York.

Nugent, C.E., Vollmann, T.E., and Ruml, J. (1968), "An experimental comparison of techniques for the assignment of facilities to locations", *Operations Research* 16, 150–173.

Pierce, J.F., and Crowston, W.B. (1971), "Tree-search algorithms for quadratic assignment problems", *Naval Research Logistics Quarterly* 18, 1–36.

Rosenblatt, M.J. (1979), "The facilities layout problem: A multi-goal approach", *International Journal of Production Research* 17, 323–332.

Rosenblatt, M.J. (1986), "The dynamics of plant layout", *Management Science* 32, 76–86.

Rosenblatt, M.J., and Kropp, D.H. (1990), "The single period stochastic plant layout problem", *IIE Transactions*, in print.

Rosenblatt, M.J., and Lee, H.L. (1987), "A robustness approach to facilities layout", *International Journal of Production Research* 25, 479–486.

Rosenblatt, M.J., and Sinuany-Stern, Z. (1986), "A discrete efficient frontier approach to the plant layout problem", *Material Flow* 3, 277–281.

Scriabin, M., and Vergin, R.C. (1985), "A cluster analysis approach to facility layout", *Management Science* 31, 33–49.

Seehof, J.M., and Evans, W.D. (1967), "Automated layout design program", *Journal of Industrial Engineering* 18, 690–695.

Tompkins, J.A., and White, J.A. (1984), *Facilities Planning*, Wiley, New York.

Urban, T.L. (1987), "A multiple criteria model for the facilities layout problem", *International Journal of Production Research* 25, 1805–1812.

Vollman, T.E., Nugent, C.E., and Zartler, R.L. (1968), "A computerized method for office layout", *Journal of Industrial Engineering* 19, 321–329.

Wilhelm, M.R., and Ward, T.L. (1987), "Solving quadratic assignment problems by 'simulated annealing'", *IIE Transactions* 19, 107–119.