

EXACT SOLUTION PROCEDURES FOR
THE CIRCULAR LAYOUT PROBLEM

Yavuz A. Bozer
Suk-Chul Rim

Technical Report #89-33

December 18, 1989

The University of Michigan
Department of
Industrial and Operations Engineering
1205 Beal Avenue
Ann Arbor, MI 48109-2117

Abstract

In this paper we present exact solution procedures for a special discrete layout problem; namely, the *Circular Layout Problem* (CLP) where the "facilities" are arranged around a simple closed loop path and flows between "facilities" occur only in one direction (e.g., clockwise). Given a flow matrix and a distance matrix, the problem is a special case of the Quadratic Assignment Problem where the objective is to assign each "facility" to one of the predetermined sites such that the total "cost" (defined by flows times distances) is minimized. As described in the paper, many practical applications of the CLP, such as arranging workstations around an AGV or conveyor loop, can be found in modern manufacturing systems.

The CLP can be divided into four subproblems based on whether the sites are equally spaced around the loop or not, and whether flow is conserved at each "facility" or not. We present an LP relaxation which optimally solves three of the subproblems. For the fourth subproblem where the sites are arbitrarily spaced *and* flow is not conserved, we present a branch and bound scheme based on a tight lower bound which is obtained by taking advantage of the circularity of the distance matrix.

The problem of determining the optimum locations of a set of facilities, while satisfying certain constraints, is generally known as the *facility layout* (or *facility location*) problem. It has many applications in the design of manufacturing systems, service facilities, office space, etc. In those cases where only a finite number of predetermined sites are considered, it is called the *discrete* layout/location problem.

Given certain "interactions" between the facilities to be located, the discrete layout problem has often been modeled as the *Quadratic Assignment Problem* (QAP). Since the QAP has been shown to be NP-complete (see Sahni and Gonzalez 1976), obtaining an optimal solution generally requires excessively high computation times. For example, Finke, Burkard and Rendl (1985) have reported that a QAP with 15 facilities requires about 50 minutes of CPU time on a CDC Cyber 76. Hence, more efficient solution procedures are required to optimally solve discrete layout problems of moderate size.

In this paper we will present exact solution procedures for a special discrete layout problem; namely, the *Circular Layout Problem* (CLP) where the facilities are arranged around a simple closed loop path and flows between "facilities" are handled only in one direction (e.g., clockwise). As motivated in the following paragraphs, such a configuration has many applications in manufacturing. To facilitate problem description we will use a manufacturing system as an example throughout the paper, although all the results would apply in other contexts.

1. PROBLEM DESCRIPTION AND APPLICATIONS

Consider a manufacturing system with automated guided vehicles (AGVs) or similar devices serving as the material handling system. Suppose the vehicles (or the devices) operate on a simple closed loop path with no "shortcuts" as shown in Figure 1, and that

there are n predetermined sites and n stations to be assigned to each site. Each station represents either an Input/Output (I/O) station or a "processor" station.

Jobs that arrive from outside the system are received through one of the I/O stations. Likewise, jobs that require no further processing are delivered to an I/O station through which they exit the system. A processor station, on the other hand, represents a machine (or a group of machines, or a cell) where jobs are processed. At a processor station, jobs are picked up from the input buffer; when processing is completed they are placed on the output buffer (see Figure 1) where they are eventually picked up by a vehicle and delivered to the input buffer of the next station. The example shown in Figure 1 contains one I/O station and 11 processor stations. Vehicles are assumed to move the jobs one at a time, and travel *unidirectionally* around the loop. Throughout the paper, without loss of generality, the vehicles are assumed to travel clockwise.

Figure 1

Let $\mathbf{F}=(f_{ij})$ denote the *flow matrix*, where $f_{ij} \geq 0$ represents the average number of jobs to be moved from *station* i to *station* j over a given length of time, for $i=1, \dots, n, j=1, \dots, n$, and $f_{ij}=0$ for all i . Also, let $\mathbf{R}=(r_{uv})$ denote the *site distance matrix*, where $r_{uv} > 0$ is the *clockwise* distance from *site* u to *site* v , for $u=1, \dots, n, v=1, \dots, n$, and $r_{uu}=0$ for all u . The most important characteristic of the CLP is that, for any pair of sites u and v , the distance from site u to site v and the distance from site v to site u sum up to the total loop length, C . That is, $r_{uv}+r_{vu}=C$ for all u and $v, u \neq v$. Such a distance matrix is called a *circular distance matrix*. Lastly, let $\mathbf{S}=(s(1), s(2), \dots, s(n))$ denote a layout of n stations, where $s(i)$ denotes the site occupied by station i .

Conceptually, any closed loop path with no shortcuts can be represented as a circle, around which there are n predetermined sites with the same distances as in the original closed loop

path. As shown in Figure 2, in general, $r_{uv} \neq r_{vu}$, except for those pairs of sites that are symmetrically located around the circle. Note that those problems that contain shortcuts cannot be modeled as a CLP.

Figure 2

We will next define the *material handling requirement* in a system as the total quantity of material to be moved over a given length of time, multiplied by the corresponding travel distance, i.e., $\sum_{i=1}^n \sum_{j=1}^n f_{ij} r_{s(i)s(j)}$, for a given flow matrix F , site distance matrix R , and layout S . In the above example based on AGVs, the material handling requirement is equal to the total *loaded* vehicle travel distance over a given length of time.

We want to find the layout S^* which minimizes the material handling requirement for a given flow matrix F and site distance matrix R for the following reasons: if the required throughput is high and the number of vehicles is fixed, then only the optimal layout S^* may meet the throughput, while other layouts may not. If required throughput is not so high, then several alternative layouts may meet the throughput. However, S^* would still be the preferred layout because it will lower the average vehicle utilization so that (1) a higher throughput requirement in the future can be met without purchasing additional vehicles; or (2) more stations can be added to the system with the same number of vehicles; or (3) capital investment can be reduced by using a smaller number of vehicles and/or less expensive, slower vehicles.

Hence, for a given flow matrix $F=(f_{ij})$ and a circular site distance matrix $R=(r_{uv})$, the CLP is defined as assigning n stations to n predetermined sites around a circle so as to minimize the material handling requirement given by $Z = \sum_{i=1}^n \sum_{j=1}^n f_{ij} r_{s(i)s(j)}$. Throughout the paper, the

fixed cost of assigning a station to any site is assumed to be negligible, or nearly equal among the sites.

An increasing number of problems arising in modern manufacturing applications can be modeled as a CLP. Depending on the type of application, the material handling requirement may be interpreted in different ways. For example, consider the case where material flow between individual stations is supported by a closed loop conveyor, as shown in Figure 3(a). In such an application the site distance matrix is clearly a circular one. Furthermore, the material handling requirement can be defined as the total distance all jobs have to travel on the conveyor. Hence, given a uniform conveyor speed around the loop, minimizing the material handling requirement effectively minimizes the total time jobs spend on the conveyor (which would minimize congestion on the loop).

Figure 3

Other examples include a station layout for a robot arm rotating in one direction only (clockwise or counterclockwise) to serve stations located around a circle as shown in Figure 3(b); and tool layout on a turret (in flexible manufacturing systems) to minimize the total turret rotation time for switching the tools (during which time the machine is idle), provided that the slots are of equal size and the turret rotates in only one direction as shown in Figure 3(c).

2. LITERATURE REVIEW

The CLP has been studied independently by several researchers over the last few years under various names. In formulating the CLP two types of objective functions have been used in the literature: (1) minimizing the total flows times distances per unit time; and (2) minimizing the total number of parts that cross the I/O station per unit time. The former has been used in the classical QAP formulation (Koopmans and Beckmann 1957) since it best represents the total interaction (or material handling requirement). A few studies have used this objective function in formulating the CLP. Kiran and Karabati (1988) develop a

polynomial time algorithm for a special case of the CLP where material flow is allowed only between a refixturing station and all other stations. They also present a branch and bound algorithm with dominance rules for the general case. Assuming that flow is conserved, Kiran, Unal and Karabati (1989) formulate the problem as a station sequencing problem and present an integer programming formulation. Based on empirical results, the authors report that the LP relaxation of the IP formulation yields integer optimal solutions for over 3,600 randomly generated problems. However, they were unable to prove that one would always obtain integer solutions.

The objective function of the second type defined above has been used in a few studies. Afentakis (1989) first formulates the problem as an integer programming problem. A heuristic algorithm using block interchanges as opposed to pairwise interchanges is also presented. Kouvelis and Kim (1989) present dominance rules, three constructive heuristic algorithms, and a branch and bound algorithm with decomposition principles for large flow matrices. Both Afentakis (1989) and Kouvelis and Kim (1989) show that the problem with the objective function of the second type is still NP-complete.

Lastly, Bartholdi and Platzman (1989) and Bozer and Srinivasan (1989) analyzed the throughput capacity of an AGV loop such as the one shown in Figure 1. Neither study, however, addressed the layout problem.

3. THE CIRCULAR LAYOUT PROBLEM

The CLP can be formulated as a quadratic assignment problem (QAP). Letting I_n denote the set of first n positive integers, the QAP formulation of the CLP can be given as follows:

$$\text{Min. } \sum_{i=1}^n \sum_{j=1}^n \sum_{u=1}^n \sum_{v=1}^n f_{ij} r_{uv} x_{iu} x_{jv} \quad (1)$$

$$\text{(QAP) s.t. } \sum_{i=1}^n x_{iu} = 1, \quad \text{for all } u \text{ in } I_n, \quad (2)$$

$$\sum_{u=1}^n x_{iu} = 1, \quad \text{for all } i \text{ in } I_n, \quad (3)$$

$$x_{iu} = 0 \text{ or } 1, \quad \text{for all } i \text{ and } u \text{ in } I_n, \quad (4)$$

where $x_{iu}=1$ indicates that station i is located at site u . In order to develop an efficient solution procedure for the CLP, we will first present a special case, namely, the Equi-Distance CLP. Subsequently, we will extend the results to more general cases of the CLP.

3.1. The Equi-Distance Circular Layout Problem (ED-CLP)

Consider n *equally spaced* predetermined sites arranged around a circle as shown in Figure 4(a). Without loss of generality, we will assume that adjacent sites are unit length apart, and that the length of the circumference is equal to n . Recall that $\mathbf{R}=(r_{uv})$ is a (circular) *site* distance matrix, where r_{uv} denotes the clockwise distance from *site* u to *site* v . Let us define $\mathbf{D}=(d_{ij})$ as a matrix of *decision* variables for the CLP, where d_{ij} denotes the clockwise distance from *station* i to *station* j , $i \neq j$. For example, given the set of sites shown in Figure 4(a), we have $r_{12}=1$, $r_{13}=2$, and so on. However, given a corresponding layout shown in Figure 4(b), we have $d_{52}=1$, $d_{56}=2$, and so on. Note that $\mathbf{D}=(d_{ij})$ is also a circular distance matrix, that is, $d_{ij}+d_{ji}=n$ for all i and j in I_n , $i \neq j$.

Figure 4

The flow matrix $\mathbf{F}=(f_{ij})$ can be calculated as follows: suppose there are K types of jobs, each of which has its own routing, that is, the sequence of stations they visit. Let Q_k denote the average number of jobs of type k ($k=1, \dots, K$) to be processed over a given length of time and $V_{ij}(k)$ be the total number of times that job type k is moved from station i to station j before it leaves the system. Then the flow matrix is obtained as

$$\mathbf{F}=(f_{ij}), \quad \text{where } f_{ij} = \sum_{k=1}^K Q_k V_{ij}(k), \quad \text{for all } i \text{ and } j \text{ in } I_n, \text{ and } f_{ii}=0 \text{ for all } i \text{ in } I_n. \quad (5)$$

Given the special structure of the problem, we will attempt to formulate the ED-CLP as a Linear Programming (LP) problem. Although the CLP is defined for $n \geq 3$, for notational convenience, we will consider CLPs only with $n \geq 4$ in the remainder of the paper. (It is straightforward to obtain the optimal solution for $n=3$.) As shown in Figure 4(b), regardless of the sequence of stations, the distances from station i to all other stations sum up to a constant. That is,

$$\sum_{\substack{j=1 \\ j \neq i}}^n d_{ij} = 1 + 2 + \dots + (n-1) = \frac{n(n-1)}{2}, \quad \text{for all } i \text{ in } I_n. \quad (6)$$

By the same argument, the distances to station j from all other stations sum up to the same constant. That is,

$$\sum_{\substack{i=1 \\ i \neq j}}^n d_{ij} = 1 + 2 + \dots + (n-1) = \frac{n(n-1)}{2}, \quad \text{for all } j \text{ in } I_n. \quad (7)$$

Furthermore, since $D=(d_{ij})$ is a circular distance matrix,

$$d_{ij} + d_{ji} = n, \quad \text{for all } i \text{ and } j \text{ in } I_n, i \neq j. \quad (8)$$

The next set of constraints proved to be critical for our study. These constraints are concerned with the precedence relationship of any three stations around a circle. As shown in Figure 5, starting at station i , either j precedes k (i.e., $d_{ij}+d_{jk}=d_{ik}$), or k precedes j (i.e., $d_{ij}+d_{jk}=d_{ik}+n$). Hence,

$$d_{ij} + d_{jk} = (d_{ik} \text{ or } d_{ik}+n), \quad \text{for all distinct } i, j, \text{ and } k \text{ in } I_n. \quad (9)$$

Figure 5

Unfortunately, if we attempt to employ (9) as a constraint set, the formulation would have to include 0–1 integer variables. Hence, the following relaxed inequality will be used instead of (9):

$$d_{ik} \leq d_{ij} + d_{jk} \leq d_{ik} + n, \quad \text{for all distinct } i, j, \text{ and } k \text{ in } I_n. \quad (10)$$

As defined earlier, the objective is to minimize the material handling requirement. The decision variables are d_{ij} 's, where d_{ij} denotes the clockwise distance from station i to station j , $i \neq j$. Hence, the LP relaxation for the ED-CLP is constructed as follows:

$$\text{Min. } \sum_i \sum_{j \neq i} f_{ij} d_{ij} \quad (11)$$

$$(\text{LP}^0) \quad \text{s.t. } \sum_{j \neq i} d_{ij} = \frac{n(n-1)}{2}, \quad \text{for all } i \text{ in } I_n, \quad (12)$$

$$\sum_{i \neq j} d_{ij} = \frac{n(n-1)}{2}, \quad \text{for all } j \text{ in } I_n, \quad (13)$$

$$d_{ij} + d_{ji} = n, \quad \text{for all } i \text{ and } j \text{ in } I_n, i \neq j, \quad (14)$$

$$d_{ij} + d_{jk} \leq d_{ik} + n, \quad \text{for all distinct } i, j \text{ and } k \text{ in } I_n, \quad (15)$$

$$d_{ij} + d_{jk} \geq d_{ik}, \quad \text{for all distinct } i, j \text{ and } k \text{ in } I_n, \quad (16)$$

$$d_{ij} \geq 0, \quad \text{for all } i \text{ and } j \text{ in } I_n, i \neq j. \quad (17)$$

Note that the above model represents a LP relaxation because constraints (15) and (16) are used instead of (9), and d_{ij} 's are *not* restricted to integer values.

An important characteristic of a feasible solution $D=(d_{ij})$ to LP^0 is that all d_{ij} 's implicitly have the same upper and lower bounds. Consider an arbitrary d_{ij} ($1 \leq i < j \leq n$) in D and a set of precedence constraints which contain d_{ij} as listed in (18) below:

$$\begin{array}{lll} d_{i1} & \leq d_{ij} + d_{j1} & \leq d_{i1} + n \\ \vdots & \vdots & \vdots \\ d_{i,i-1} & \leq d_{ij} + d_{j,i-1} & \leq d_{i,i-1} + n \\ d_{i,i+1} & \leq d_{ij} + d_{j,i+1} & \leq d_{i,i+1} + n \\ \vdots & \vdots & \vdots \\ d_{i,j-1} & \leq d_{ij} + d_{j,j-1} & \leq d_{i,j-1} + n \\ d_{i,j+1} & \leq d_{ij} + d_{j,j+1} & \leq d_{i,j+1} + n \\ \vdots & \vdots & \vdots \\ d_{in} & \leq d_{ij} + d_{jn} & \leq d_{in} + n \end{array} \quad (18)$$

Summing up the above $(n-2)$ constraints, we obtain:

$$\sum_{k \neq i,j} d_{ik} \leq (n-2)d_{ij} + \sum_{k \neq i,j} d_{jk} \leq \sum_{k \neq i,j} d_{ik} + n(n-2). \quad (19)$$

Theorem 1. In any feasible solution $D=(d_{ij})$ to LP^0 , $1 \leq d_{ij} \leq n-1$ for all $1 \leq i \neq j \leq n$.

Proof. Since $\sum_{k \neq i,j} d_{ik} = n(n-1)/2 - d_{ij}$ by (12), constraint (19) can be rewritten as:

$$n(n-1)/2 - d_{ij} \leq (n-2)d_{ij} + n(n-1)/2 - d_{ji} \leq n(n-1)/2 - d_{ij} + n(n-2).$$

By (14), the above constraint can be simplified to $1 \leq d_{ij} \leq n-1$ for all $1 \leq i < j \leq n$. Also, since $d_{ji} = n - d_{ij}$, we have $1 \leq d_{ij} \leq n-1$ for all $1 \leq j < i \leq n$ as well, which completes the proof. ■

To facilitate the following proofs, we will simplify LP^0 to obtain an equivalent formulation, say, LP^* as follows: 1. constraints (13) can be eliminated since they are redundant due to (12) and (14); 2. only those indices i, j , and k such that $1 \leq i < j < k \leq n$ need to be considered in (15) and (16) since precedence constraints with other indices are redundant due to (14); 3. d_{ji} in (11) and (12) can be replaced by $n - d_{ij}$ for $1 \leq i < j \leq n$, and (14) can be eliminated; and 4. any one row among the rows given by (12) can be eliminated since, after replacing d_{ji} in (11) and (12) with $n - d_{ij}$, both sides of the summation of all constraints given by (12) are equal to zero. This implies that the n rows in question are linearly dependent. Without loss of generality, we will eliminate the last row. Consequently, the resulting LP relaxation for the ED-CLP is given by:

$$\text{Min. } \sum_{i=1}^{n-1} \sum_{j=i+1}^n f'_{ij} d_{ij} + C \quad (20)$$

$$(LP^*) \text{ s.t. } -\sum_{j=1}^{i-1} d_{ji} + \sum_{j=i+1}^n d_{ij} = \frac{n(n-1)}{2} - (i-1)n, \quad \text{for } 1 \leq i \leq n-1, \quad (21)$$

$$d_{ij} + d_{jk} \leq d_{ik} + n, \quad \text{for } 1 \leq i < j < k \leq n, \quad (22)$$

$$d_{ij} + d_{jk} \geq d_{ik}, \quad \text{for } 1 \leq i < j < k \leq n, \quad (23)$$

$$d_{ij} \geq 0, \quad \text{for } 1 \leq i < j \leq n, \quad (24)$$

where $f'_{ij} = f_{ij} - f_{ji}$, and $C = n \sum_{i=1}^{n-1} \sum_{j=i+1}^n f_{ji}$.

Let \mathbf{B} denote a *basis* of LP^* . As shown in Figure 6(a), \mathbf{B} can be divided into four matrices labeled \mathbf{P} , \mathbf{C} , $\mathbf{0}$, and the identity matrix \mathbf{I} , where \mathbf{P} is further divided into \mathbf{E} and $-\mathbf{G}$, and \mathbf{C} is further divided into \mathbf{H} and \mathbf{G} . For each triplet (i,j,k) where $1 \leq i < j < k \leq n$, there is a pair of rows in \mathbf{B} given by (22) and (23). In LP^* with n stations, there are $n(n-1)/2$ columns (or decision variables), $(n-1)$ rows in \mathbf{E} given by (21), and $2\binom{n}{3} = n(n-1)(n-2)/3$ rows in $-\mathbf{G}$, \mathbf{H} , and \mathbf{G} given by (22) and (23). Since all the d_{ij} 's must be basic variables in any basis by Theorem 1, only $(n-1)(n-2)(2n-3)/6$ out of the $n(n-1)(n-2)/3$ slack variables can be selected as basic variables in \mathbf{B} .

Let \mathbf{H} denote the matrix which consists of the pairs of rows whose slack variables are *both* selected as basic variables, and let \mathbf{G} denote the matrix which consists of the rows for which only one slack (out of a pair) is selected as a basic variable. Let $r(\mathbf{M})$ denote the number of rows in a matrix \mathbf{M} . Since \mathbf{P} is a square matrix, $r(-\mathbf{G}) = n(n-1)/2 - (n-1) = (n-1)(n-2)/2$. Hence, $r(\mathbf{H}) = n(n-1)(n-2)/3 - 2r(-\mathbf{G}) = (n-1)(n-2)(n-3)/3$. Within the submatrices \mathbf{H} and \mathbf{G} , respectively, the rows are arranged in the lexicographically increasing order of the indices; that is, $(1,2,3)$, $(1,2,4)$, \dots , $(1,2,n)$, $(1,3,4)$, and so on.

Figure 6

Recall that, $\mathbf{P} = \begin{pmatrix} \mathbf{E} \\ -\mathbf{G} \end{pmatrix}$ and $\mathbf{C} = \begin{pmatrix} \mathbf{H} \\ \mathbf{G} \end{pmatrix}$. It is known that (Bazaraa and Jarvis 1977, p.52):

$$\mathbf{B}^{-1} = \left(\begin{array}{c|c} \mathbf{P} & \mathbf{0} \\ \hline \mathbf{C} & \mathbf{I} \end{array} \right)^{-1} = \left(\begin{array}{c|c} \mathbf{P}^{-1} & \mathbf{0} \\ \hline -\mathbf{C}\mathbf{P}^{-1} & \mathbf{I} \end{array} \right), \quad (25)$$

where

$$-\mathbf{C}\mathbf{P}^{-1} = \left(\begin{array}{c|c} \mathbf{U} & \mathbf{W} \\ \hline \mathbf{V} & \mathbf{X} \end{array} \right). \quad (26)$$

In order to prove that LP^* solves the ED-CLP, we will show that the optimal solution of LP^* actually yields an *equi-distance layout*. However, we will first show (using the following four lemmas) that $-CP^{-1}$ consists of $U, V, W,$ and X as shown in Figure 6(b), where $U=V=0, X=I,$ and W is composed of 0 or ± 1 's. Subsequently, we will show that for any basic feasible solution of LP^* , exactly one of the two constraints given by (22) and (23) holds at equality for all $1 \leq i < j < k \leq n$; that is, a pair of slack variables associated with (22) and (23) must assume the values $(0,n)$ or $(n,0)$ in any basic feasible solution of LP^* .

Recall that, for each triplet (i,j,k) where $1 \leq i < j < k \leq n$, there is a pair of rows (given by (22) and (23)) in B . Let the first and second row in the pair associated with the triplet (i,j,k) be labelled $r^+(i,j,k)$ and $r^-(i,j,k)$, respectively. In other words, let $r^+(i,j,k)$ denote the row from (22) in which the coefficients of $d_{ij}, d_{ik},$ and d_{jk} are 1, -1, and 1, respectively, and let $r^-(i,j,k)$ denote the matching row from (23); that is, $r^-(i,j,k) = -r^+(i,j,k)$. Also, let $r^0(i,j,k)$ denote either $r^+(i,j,k)$ or $r^-(i,j,k)$.

Consider next a quadruplet (i,j,k,l) where $1 \leq i < j < k < l \leq n$. There are exactly four triplets associated with this quadruplet: $(i,j,k), (i,j,l), (i,k,l),$ and (j,k,l) . Let $r_m^0(i,j,k,l)$ denote one of the two rows associated with the m th triplet of the above four; that is, $r_1^0(i,j,k,l) = r^+(i,j,k)$ or $r^-(i,j,k), r_2^0(i,j,k,l) = r^+(i,j,l)$ or $r^-(i,j,l),$ and so on.

Lemma 1. For any quadruplet (i,j,k,l) where $1 \leq i < j < k < l \leq n$, there exists $q=(q_1,q_2,q_3,q_4)$ such that $\sum_{m=1}^4 q_m r_m^0(i,j,k,l) = 0$ if and only if $q_m = \pm \omega$ for some $\omega > 0$ ($1 \leq m \leq 4$).

Proof. See Appendix.

Lemma 2. Every row of H can be represented as a linear combination of the rows of G .

Proof. Since G is included in C , it is sufficient to show that $\text{rank}(G) = \text{rank}(C)$. Recall that, for the ED-CLP with n stations, $r(G) = (n-1)(n-2)/2$. Also, all the rows of G are linearly independent. (Otherwise, due to $-G$, matrix B would not represent a basis.)

Hence, $\text{rank}(\mathbf{G}) = r(\mathbf{G})$. On the other hand, consider a quadruplet (i^*, j, k, l) where $1 \leq i^* < j < k < l \leq n$. By Lemma 1, $r^0(j, k, l)$ can be expressed as a linear combination of $r^0(i^*, j, k)$, $r^0(i^*, j, l)$, and $r^0(i^*, k, l)$. Hence, the maximum number of independent rows in \mathbf{C} is given by $\binom{n-1}{2} = (n-1)(n-2)/2$ when $i^* = 1$. That is, $\text{rank}(\mathbf{C}) \leq (n-1)(n-2)/2 = \text{rank}(\mathbf{G})$. However, since \mathbf{C} includes \mathbf{G} , $\text{rank}(\mathbf{C}) \geq \text{rank}(\mathbf{G})$. Therefore, $\text{rank}(\mathbf{C}) = \text{rank}(\mathbf{G})$. ■

Lemma 3. Let \mathbf{R}_{rs}^1 denote the matrix \mathbf{P} in which the s th row of \mathbf{E} is replaced by the r th row of \mathbf{H} . Then, $\det \mathbf{R}_{rs}^1 = 0$ for $1 \leq r \leq (n-1)(n-2)(n-3)/3$ and $1 \leq s \leq n-1$.

Proof. See Appendix.

Lemma 4. Let \mathbf{R}_{rs}^2 denote the matrix \mathbf{P} in which the s th row of \mathbf{E} is replaced by the r th row of \mathbf{G} . Then, $\det \mathbf{R}_{rs}^2 = 0$ for $1 \leq r \leq (n-1)(n-2)/2$ and $1 \leq s \leq n-1$.

Proof. Since \mathbf{R}_{rs}^2 contains two identical rows with opposite signs, $\det \mathbf{R}_{rs}^2 = 0$ for all r and s . ■

Lemma 5. Let \mathbf{R}_{rs}^3 denote the matrix \mathbf{P} in which the s th row of $-\mathbf{G}$ is replaced by the r th row of \mathbf{H} . Then, $\det \mathbf{R}_{rs}^3 = 0$ or $\pm \det \mathbf{P}$ for $1 \leq r \leq (n-1)(n-2)(n-3)/3$ and $1 \leq s \leq (n-1)(n-2)/2$.

Proof. See Appendix.

Lemma 6. Let \mathbf{R}_{rs}^4 denote the matrix \mathbf{P} in which the s th row of $-\mathbf{G}$ is replaced by the r th row of \mathbf{G} . Then, $\det \mathbf{R}_{rs}^4$ is equal to $-\det \mathbf{P}$ if $r=s$, and 0 otherwise, for $1 \leq r, s \leq (n-1)(n-2)/2$.

Proof. Straightforward.

Using the last four lemmas (Lemmas 3 through 6), we will next prove a theorem which is a critical one for the paper.

Theorem 2. Consider a triplet (i,j,k) where $1 \leq i < j < k \leq n$ and the corresponding pair of precedence constraints given by (22) and (23). In any *basic* feasible solution of LP^* , exactly one of these two constraints will hold at equality.

Proof. Obviously, in any feasible solution of LP^* , constraints (22) and (23) cannot both hold at equality. However, both constraints may hold as strict inequalities in some feasible solutions. We will show that, in a *basic* feasible solution, both constraints cannot hold as strict inequalities; that is, a pair of slack variables associated with (22) and (23) must be equal to either $(0,n)$ or $(n,0)$.

Let the g,h minor of P , denoted by P_{gh} , be the matrix obtained by eliminating the g th row and the h th column of P . Also, let $M(i,j)$ denote the element in the i th row and the j th column of a matrix M . Then, by (26), we have:

$$U(i,j) = \sum_{k=1}^p -C(i,k)P^{-1}(k,j). \quad (27a)$$

That is,

$$\begin{aligned} U(i,j) &= \sum_{k=1}^p -C(i,k) \frac{(-1)^{k+j} \det P_{jk}}{\det P} \\ &= \frac{(-1)}{\det P} \left[\sum_{k=1}^p (-1)^{k+j} C(i,k) \det P_{jk} \right], \end{aligned} \quad (27b)$$

where $p = n(n-1)/2$. Note that the expression within the brackets defines the determinant of a matrix obtained by replacing the j th row of P with the i th row of C . Such a matrix was defined earlier as R_{ij}^1 in Lemma 3 where it was shown that $\det R_{ij}^1 = 0$ for all i and j . Hence, $U(i,j) = \frac{(-1)}{\det P} \det R_{ij}^1 = 0$ for all i and j .

Likewise, $V(i,j) = \frac{(-1)}{\det P} \det R_{ij}^2$, where R_{ij}^2 is as defined in Lemma 4. By Lemma 4, $\det R_{ij}^2 = 0$ for all i and j . Therefore, $V = 0$. Following a similar approach, $W(i,j) = \frac{(-1)}{\det P}$

$\det \mathbf{R}_{ij}^3$, where \mathbf{R}_{ij}^3 is as defined in Lemma 5. Thus, by Lemma 5, $W(i,j)$ is equal to 0 or ± 1 for all i and j .

Consider next the matrix \mathbf{H} . Recall that \mathbf{H} is composed of pairs of rows. Let \mathbf{H}_i and \mathbf{H}_i' denote two rows in a pair. Then, $\mathbf{H}_i' = -\mathbf{H}_i$ by definition. Using an expression similar to (27a), one may obtain a row in \mathbf{W} , say, \mathbf{W}_i by using $H(i,k)$ in place of $C(i,k)$. Likewise, one can obtain another row in \mathbf{W} , say, \mathbf{W}_i' by using $H'(i,k)$ in place of $C(i,k)$ in (27a). Since $H'(i,k) = -H(i,k)$ for all i and k , it is straightforward to show that $\mathbf{W}_i' = -\mathbf{W}_i$ for all i . Thus, \mathbf{W} also is composed of pairs of rows with opposite signs. Lastly, $X(i,j) = \frac{(-1)}{\det \mathbf{P}} \det \mathbf{R}_{ij}^4$, where \mathbf{R}_{ij}^4 is as defined in Lemma 6. Thus, by Lemma 6, $\mathbf{X}=\mathbf{I}$.

Let \mathbf{b}^1 , \mathbf{b}^2 , \mathbf{b}^3 , and \mathbf{b}^4 denote the subvectors of \mathbf{b} associated with \mathbf{E} , $-\mathbf{G}$, \mathbf{H} , and \mathbf{G} , respectively, as shown earlier in Figure 6(c). Then, $b^1(i) = n(n-1)/2 - (i-1)n$ for $1 \leq i \leq n-1$, and $\mathbf{b}^3 = (n, 0, n, 0, \dots, n, 0)$. Also, $b^2(i)$ and $b^4(i)$ is either 0 or n , where $b^2(i) + b^4(i) = n$ for all i . Also, let $\mathbf{Y} = \{(Y_{m1}, Y_{m2}) \mid m=1, \dots, (n-1)(n-2)(n-3)/6\}$ denote the set of pairs of slack variables associated with \mathbf{H} . Recall that a basic feasible solution of LP^* is equal to $\mathbf{B}^{-1}\mathbf{b}$. Let $Y(i)$ denote the i th slack variable in \mathbf{Y} . Then, $Y(i) = \mathbf{U}_i \mathbf{b}^1 + \mathbf{W}_i \mathbf{b}^2 + \mathbf{I}_i \mathbf{b}^3$. Hence, $Y_{m1} = kn + n$ and $Y_{m2} = -kn$ for all m , where k is an integer unrestricted in sign. Since all slack variables must be nonnegative, we have $(k+1)n \geq 0$ and $-kn \geq 0$. Thus, $k = -1$ or 0 , which implies that $(Y_{m1}, Y_{m2}) = (0, n)$ or $(n, 0)$ for all m . Similarly, let $\mathbf{T} = \{(T_{m1}, T_{m2}) \mid m=1, \dots, (n-1)(n-2)/2\}$ denote the set of pairs of slack variables associated with \mathbf{G} and $-\mathbf{G}$. Since the slack variables associated with $-\mathbf{G}$ are equal to 0 by definition, $(T_{m1}, T_{m2}) = (0, n)$ or $(n, 0)$ for all m . Hence, in any basic feasible solution of LP^* , exactly one of the two precedence constraints in a given pair will hold at equality. ■

Recall that a solution $\mathbf{D}=(d_{ij})$ of LP^* yields an upper-right triangular matrix since only those d_{ij} 's with $1 \leq i < j \leq n$ are considered in LP^* . However, since the elements in the lower-left triangular matrix can be uniquely determined by (14), in the remainder of the paper, "a

solution of LP^* refers to the *complete* solution matrix. Using the following lemma and a corollary, we will next show that any *basic* feasible solution of LP^* is an *integer solution*.

Lemma 7. Let D be a feasible solution of LP^* . If there exists at least one fractional value in D , then there exists at least one pair of precedence constraints given by (22) and (23), both of which hold as strict inequalities.

Proof. Let $d_{ij} = \alpha$ be a fractional value in D . Suppose there is no pair of constraints both of which hold as strict inequalities. That is, every constraint in (18) holds at equality on one side. Further suppose that m out of $n-2$ constraints in (18) hold at equality on the right side, and the remaining $(n-2-m)$ constraints hold at equality on the left side of the constraint ($0 \leq m \leq n-2$). Then, regardless of which m constraints hold at equality on the right side, by summing up $(n-2)$ equalities, we obtain $(n-2)d_{ij} + \sum_{k \neq i,j} d_{jk} = \sum_{k \neq i,j} d_{ik} + mn$, or equivalently, $(n-2)\alpha + n(n-1)/2 - (n-\alpha) = n(n-1)/2 - \alpha + mn$. Simplifying the above equation, we obtain $\alpha = m+1$, which is contradictory. Hence, there exists at least one pair of precedence constraints both of which hold as strict inequalities. ■

Corollary 1. Let D be a feasible solution of LP^* . If either (22) or (23) holds at equality for all $1 \leq i < j < k \leq n$, then D is an integer solution.

Proof. The above corollary is the contra positive of Lemma 7.

Theorem 3. Any *basic* feasible solution of LP^* is an integer solution.

Proof. By Theorem 2, we showed that in any *basic* feasible solution of LP^* exactly one of the two constraints given by (22) and (23) will hold at equality for all triplets (i,j,k) where $1 \leq i < j < k \leq n$. Furthermore, it was shown in Corollary 1 that in a feasible solution of LP^* , if exactly one of the two precedence constraints in a given pair holds as an equality for all $1 \leq i < j < k \leq n$, then the solution is an integer solution. Hence, any *basic* feasible solution of LP^* is an integer solution. ■

It is straightforward to show that the coefficient matrix of LP^* is *not* unimodular. However, as shown above, due to the special structure of LP^* , all its extreme points are integer.

Lastly, we will prove that if a feasible solution D of LP^* is an integer matrix, and either (22) or (23) holds at equality for all $1 \leq i < j < k \leq n$, then D actually yields an equi-distance layout. To do so, let us define an *element matrix* $\bar{D} = (\bar{d}_{ij})$ of size n as a square matrix, where all diagonal elements are 0, and each element of $I_{n-1} = \{1, 2, \dots, n-1\}$ appears once and only once in each row and each column of \bar{D} . An example of an element matrix is shown in Table I(a). An element matrix $\bar{D} = (\bar{d}_{ij})$ of size n is called a *circular element matrix* if $\bar{d}_{ij} + \bar{d}_{ji} = n$, for all i and j , $i \neq j$. An example of a circular element matrix is shown in Table I(b). Let us next define the *standard matrix* $D^0 = (d_{ij}^0)$ of size n as a square matrix where $d_{ij}^0 = j - i$ for $1 \leq i < j \leq n$, and $n - (i - j)$ for $1 \leq j < i \leq n$, and 0 for $1 \leq i = j \leq n$ as shown in Table I(c). Note that a standard matrix is also a circular element matrix.

Table I

Theorem 4. Any *basic* feasible solution D of LP^* is an element matrix.

Proof. Suppose D is not an element matrix. Then either there exists at least one row or a column of D in which a value K appears more than once, or there exists at least one row or a column of D in which a value K does not appear, for some integer K ($1 \leq K \leq n-1$).

1) Without loss of generality, suppose that K appears more than once in the i th row of D . Let $d_{ij} = d_{ik} = K$. Since either (22) or (23) holds at equality by Theorem 2, d_{jk} is equal to 0 or n ; which contradicts Theorem 1.

2) Without loss of generality, suppose that K does not appear in the i th row of D . Recall that (21) is equivalent to (12) due to (14). Since $\sum_{k=1}^{n-1} k = n(n-1)/2$, either the row cannot

sum up to $n(n-1)/2$, or some other integer M ($1 \leq M \leq n-1$) has to appear more than once in the row. From 1) above, this also contradicts Theorem 1. ■

Using the following two lemmas and a corollary, we will finally show that LP^* optimally solves the ED-CLP. Recall that, in the station distance matrix $D=(d_{ij})$, d_{ij} denotes the clockwise distance from station i to station j . By "interchanging stations i and j " in D , we mean interchanging columns i and j of D and, at the same time, interchanging rows i and j of D .

Lemma 8. Consider an element matrix \bar{D} . If any pair of stations in \bar{D} are interchanged, the resulting matrix \bar{D}' is also an element matrix.

Proof. See Appendix.

Corollary 2. If any pair of stations in a circular matrix are interchanged, the resulting matrix is also a circular matrix.

Proof. The proof is straightforward by the proof of Lemma 8.

Lemma 9. Consider a feasible solution D of LP^* . If D is an element matrix, then the standard matrix can be obtained by a series of pairwise interchanges of stations in D .

Proof. Since D is an element matrix, the value 1 appears once and only once in each column and each row of D . Suppose we perform a series of pairwise interchanges of stations in D to obtain $D'=(d'_{ij})$ where $d'_{i,i+1}=1$, for $i=1,2,\dots,n-1$, and consequently $d'_{n,1}=1$. Since D' is also an element matrix by Lemma 8, and we know that $d'_{23}=1$, d'_{13} must assume one of the values in $\{2,3,\dots,n-1\}$. By (22) and (23), $d'_{13} \leq d'_{12} + d'_{23} \leq d'_{13} + n$. Since $d'_{12} + d'_{23} = 2$, d'_{13} must assume the value 2. By the same argument, $d'_{i,i+2} = 2$, for $i=1,2,\dots,n-2$. Similarly, since we know that $d'_{34}=1$ and $d'_{24}=2$, d'_{14} must assume one of the values in $\{3,4,\dots,n-1\}$. By (22) and (23), $d'_{14} \leq d'_{12} + d'_{24} \leq d'_{14} + n$. Since $d'_{12} + d'_{24} = 3$, d'_{14} must assume the value 3. By the same argument, $d'_{i,i+3} = 3$, for

$i=1,2,\dots,n-3$. Repeating this procedure, the upper-right triangular matrix of D' can be obtained as given in Table I(c). Since D is a circular matrix by (14), D' is also a circular matrix by Corollary 2. Hence, the lower left triangular matrix of D' can be also obtained as given in Table I(c), which is the standard matrix. ■

Theorem 5. LP^* optimally solves the Equi-Distance Circular Layout Problem.

Proof. By Theorem 4, we showed that any *basic* feasible solution of LP^* is an element matrix. Furthermore, it is shown in Lemma 9 that if a feasible solution D of LP^* is an element matrix, then the standard matrix can be obtained by a series of pairwise interchanges of stations in D . It is straightforward to show that if D is a standard matrix, then D yields a layout in which the stations are equally spaced along a circle in the same sequence as in D . Therefore, LP^* optimally solves the ED-CLP. ■

3.2. The Non-Equi-Distance, Conserved-Flow CLP

Consider a CLP in which flow is conserved at each station, but adjacent sites are not equally spaced around a circle (namely, the NED-CF CLP). Flow is said to be *conserved* at station k if total flow entering station k is equal to total flow leaving station k , i.e., if

$$\sum_{i=1}^n f_{ik} = \sum_{j=1}^n f_{kj}. \text{ A flow matrix is called a } \textit{conserved flow} \text{ matrix if flow is conserved at}$$

each station. Note that flow may not be conserved at some stations if there are more than one I/O stations, or if defective jobs are scrapped at some stations.

In LP^* (presented in Section 3.1), the flow values do not appear in any of the constraints. Hence, the flow values do not affect the feasible space of LP^* . Therefore, LP^* solves the ED-CLP, regardless of whether flow is conserved or not at each station. However, flow conservation plays an important role in analyzing the Non-Equi-Distance CLP as shown below. The Equi-Distance CLP with conserved flow is clearly a special case of the NED-

CF CLP. Although LP^* cannot be directly applied to the NED-CF CLP, it is shown by Theorem 6 that the optimal solution of the NED-CF CLP can be obtained by solving LP^* .

Theorem 6. LP^* yields the optimal solution for the NED-CF CLP.

Proof. Consider an arbitrary non-equi-distance layout with conserved flow. Let $\lambda(i) = \sum_{k=1}^n f_{ki}$ and $\mu(i) = \sum_{k=1}^n f_{ik}$ denote the total in-flow and total out-flow of station i , respectively.

Suppose station i is moved clockwise a small distance, say, $\delta > 0$ units along the circumference. As a result, the material handling requirement is increased by $\delta\lambda(i)$, while it is decreased by $\delta\mu(i)$. Since flow is conserved at station i , that is, $\lambda(i) = \mu(i)$, the above increase and decrease in the material handling requirement cancel each other out, and the objective function value does not change. The same argument holds if station i is moved counterclockwise.

Moreover, the result holds for any $\delta > 0$ as long as station i remains between the two adjacent stations j and k . (If station i is moved *beyond* either j or k , then the objective function value may change due to a new sequence of stations. This issue will be further mentioned in Theorem 7.) Note that the above result holds for any station, independent of the locations of the other stations. Therefore, the objective function value does not change even if all the stations were moved in either direction as long as the sequence of stations remains unchanged.

Therefore, for any NED-CF layout, there exists an Equi-Distance CF layout with the same sequence of stations, and hence, the same objective function value. Let $S^*(ED)$ denote an optimal layout for an ED-CLP obtained from LP^* . Since LP^* optimally solves the ED-CLP, all layouts having the same sequence of stations as in $S^*(ED)$ are optimal layouts for the original NED-CF CLP, regardless of which site the sequence of stations starts from. ■

Corollary 3. If flow is conserved at each station, then the sequence of stations in a circular layout completely determines the objective function value, regardless of the exact site locations.

Proof. The proof is straightforward by the proof of Theorem 6.

In proving Theorem 6, station i was not allowed to move beyond either one of the two adjacent stations. However, under a certain condition stated below, this restriction can be relaxed.

Theorem 7. Consider a pair of adjacent stations i and j in an arbitrary circular layout with conserved flow. If $f_{ij}=f_{ji}$, then interchanging stations i and j does not change the objective function value of the layout.

Proof. See Appendix.

In particular, if there is a pair of adjacent stations i and j such that $f_{ij}=f_{ji}$ in an optimal layout S^* with conserved flow, then, by Theorem 7, an alternative optimal layout can be obtained by interchanging stations i and j in S^* .

Theorem 8. If the flow matrix is symmetric, then any layout is optimal for the CLP.

Proof. The proof is straightforward by the proof of Theorem 7.

3.3. The Non-Equi-Distance, Non-Conserved-Flow CLP

For the Circular Layout Problem in which neither the sites are equally spaced nor the flow is conserved at each station, the LP relaxation shown in Section 3.1 does not apply. Hence, the problem must be formulated as a QAP. Presently, the most successful exact solution procedure for the QAP is a branch and bound technique based on matrix reductions and Gilmore-Lawler bounds (Finke, Burkard and Rendl 1987). Note that the NED-NCF CLP still has a circular distance matrix. By taking advantage of the circularity, we can

obtain a new lower bound which is at least as tight as the Gilmore-Lawler bound. For details on the Gilmore-Lawler lower bound, the reader may refer to Burkard and Derigs (1980) or Bazaraa and Elshafei (1979).

Consider the flow between stations i and j . Due to the circularity given by (8), *regardless of the locations of stations i and j* , a certain amount of flow, which is given by the minimum of f_{ij} and f_{ji} , has to travel around the circle once. Given a flow matrix $F=(f_{ij})$, let $H=(h_{ij})$ denote the *net flow matrix*, where $h_{ij} = f_{ij} - \min(f_{ij}, f_{ji})$. Let W denote the sum of smaller flows in each pair (f_{ij}, f_{ji}) , that is, $W = \sum_i \sum_{j>i} \min(f_{ij}, f_{ji})$. Also, let K denote the

lower bound obtained by applying the Gilmore-Lawler lower bound scheme to the net flow matrix H . Then, given that the total length of the circle is equal to n , i.e., the number of stations, the *net flow lower bound* is equal to $K + nW$.

Theorem 9. The net flow bound is at least as tight as the Gilmore-Lawler bound.

Proof. Let $\Phi = \{(i,j) \mid f_{ij} \geq f_{ji}\}$. Also, let $S = \{s(i), i=1, \dots, n\}$ be the layout vector where $s(i)$ denote the site occupied by station i . Recall that $R=(r_{uv})$ denotes the site distance matrix as defined in Section 1. Then, the objective function value Z given by (11) can be written as follows:

$$\begin{aligned}
Z &= \sum_{i=1}^n \sum_{j=1}^n f_{ij} r_{s(i)s(j)} \\
&= \sum_{(i,j) \in \Phi} f_{ij} r_{s(i)s(j)} + \sum_{(i,j) \notin \Phi} f_{ij} r_{s(i)s(j)} \\
&= \sum_{(i,j) \in \Phi} f_{ij} r_{s(i)s(j)} + \sum_{(j,i) \in \Phi} f_{ij} (n - r_{s(j)s(i)}) \\
&= \sum_{(i,j) \in \Phi} (f_{ij} - f_{ji}) r_{s(i)s(j)} + n \sum_{(j,i) \in \Phi} f_{ij} \\
&= \sum_i \sum_j h_{ij} r_{s(i)s(j)} + nW, \tag{28}
\end{aligned}$$

where nW represents the fixed "cost" (i.e., flows times distances) which is independent of the layout. Since a nonnegative constant is subtracted prior to computing the lower bound, the net flow lower bound will be at least as tight as the Gilmore-Lawler bound. If W is equal to zero, that is, if at least one element in each pair (f_{ij}, f_{ji}) is equal to zero, then the net flow bound is identical to the Gilmore-Lawler bound. ■

To evaluate the performance of the net flow bound relative to the Gilmore-Lawler bound, we compared the CPU times that each corresponding branch and bound code required to solve a set of randomly generated problems. The parameters considered in generating the random problems are: 1. the number of stations ($N = 5, 8, \text{ or } 12$); 2. the density of the flow matrix: dense (no 0's), medium (20% 0's), or sparse (40% 0's); 3. symmetry of the flow matrix: random or close to symmetric (in the latter case f_{ji} is generated from a uniform distribution over the range $[\cdot 8f_{ij}, 1.2f_{ij}]$); and 4. distribution of the site locations around the circle: even, medium, or uneven. Unlike the other parameters listed above, it is difficult to control the evenness of the site locations. Hence, instead of directly controlling the evenness, we will choose a *relatively* even, medium and uneven distance matrix among 50 randomly generated distance matrices using the following scheme.

Recall that the total length of the circle is n for a problem with n stations, and $\mathbf{R}=(r_{uv})$ is the site distance matrix. Let $Q = \sum_{u=1}^n |r_{u,u+1}-1|$, where $r_{n,n+1} = r_{n,1}$, denote the degree of unevenness of the site distance matrix \mathbf{R} . (Note that in a perfectly even distance matrix, $r_{u,u+1}=1$ for all $u=1, \dots, n$.) Then, the distance matrix whose Q value is the smallest, 25th, and the largest among the 50 randomly generated distance matrices is taken as the even, medium, and uneven distance matrix, respectively. Flow matrices are also randomly generated where each f_{ij} is a random integer between 0 and 100. For each parameter set, three problems with random flows are solved by the branch and bound code developed by Burkard and Derigs (1980) using the net flow bound and the Gilmore-Lawler bound.

Comparison of the CPU times between the two bounding schemes is summarized in Table II.

Table II

Although the net flow bound is not a fundamentally new bound, the branch and bound code with the net flow bound clearly outperforms the same code with the Gilmore-Lawler bound over all the parameter values tested. For problems with 12 stations, a dramatic improvement has been achieved: most problems took less than a minute with the net flow bound whereas all the problems exceeded the one hour time limit with the Gilmore-Lawler bound. Note that, as the flow matrix approaches a symmetric one, a larger amount of fixed "cost" can be subtracted a priori; this yields a tighter lower bound and results in a larger improvement over the Gilmore-Lawler bound. As an extreme case, if the flow matrix is perfectly symmetric, then any layout is an optimal layout (as stated earlier in Theorem 8) and the net flow lower bound is equal to the optimum objective function value. Also, based on Table II, neither the distribution of site locations nor the flow density appears to be a critical parameter.

4. CONCLUSIONS

In this paper, we studied a special case of the Quadratic Assignment Problem, namely, the Circular Layout Problem in which the "sites" are located around a simple closed loop path with no shortcuts. Assuming that all the flows occur in one direction only (i.e., clockwise or counterclockwise), we showed that, if the sites are equally spaced and/or flow is conserved at each "facility", then the problem can be solved as a linear programming problem (which can be implemented as a polynomial time algorithm). Although the sites are not likely to be equally spaced in most cases, flow is usually conserved at each "facility" in manufacturing applications. For those cases where neither condition holds, however, we derived a lower bound by modifying the well-known Gilmore-Lawler bound. The

resulting bounds were reasonably strong and they become tighter as the flow matrix approaches a symmetric one.

Currently, we are in the process of extending the above results to the *bidirectional* Circular Layout Problem. Although conveyor-based systems are unidirectional, some automated guided vehicle systems and robot arm applications can be bidirectional.

REFERENCES

- Afentakis, P. (1989) "A Loop Layout Design Problem for Flexible Manufacturing Systems," *The International Journal of Flexible Manufacturing Systems*, 1, pp.175-196.
- Bartholdi, III, J. J. and Platzman, L. K. (1989) "Decentralized Control of Automated Guided Vehicles on a Simple Loop," *IIE Transactions*, 21, pp.76-81.
- Bazaraa, M.S. and Elshafei, A. N. (1979) "An Exact Branch and Bound Procedure for Quadratic Assignment Problem," *Naval Research Logistics Quarterly*, 26, pp.109-121.
- Bazaraa, M. S. and Jarvis, J. J. (1977) *Linear Programming and Network Flows*, John Wiley and Sons, NY.
- Bozer, Y. A. and Srinivasan, M. M. (1989) "Tandem Configurations for Automated Guided Vehicle Systems and the Analysis of Single Vehicle Loops," to appear in *IIE Transactions*.
- Burkard, R.E. and Derigs, U. (1980) *Assignment and Matching Problems: Solution Methods with FORTRAN-Programs*, Lecture Notes in Economics and Mathematical Systems #184, Springer-Verlag, NY.
- Finkbeiner, II, D. T. (1966) *Introduction to Matrices and Linear Transformations*, 2nd edition, W. H. Freeman and Company, San Francisco.
- Finke, G., Burkard, R. E. and Rendl, F. (1985) "Quadratic Assignment Problems," Working Paper, Department of Applied Mathematics, Technical University of Nova Scotia, Halifax, NS, Canada.
- Finke, G., Burkard, R. E. and Rendl, F. (1987) "Quadratic Assignment Problems," *Annals of Discrete Mathematics*, 31, pp.61-82.
- Kiran, A. S. and Karabati, S. (1988) "The Station Location Problem on Unicyclic Material Handling Networks," Working Paper, University of Southern California, Department of Industrial and Systems Engineering, Los Angeles.
- Kiran, A. S., Unal, A. T. and Karabati, S. (1989) "A Location Problem on Unicyclic Networks: Balanced Case," Working Paper 89-11, University of Southern California, Department of Industrial and Systems Engineering, Los Angeles.
- Koopmans, T. C. and Beckman, M. (1957) "Assignment Problems and the Location of Economic Activities," *Econometrica*, 25, pp.53-76.
- Kouvelis, P. and Kim, M. W. (1989) "Unidirectional Loop Network Layout Problem in Automated Manufacturing System," Working Paper 88/89-4-5, Department of Management, The University of Texas at Austin.
- Sahni, S. and Gonzalez, T. (1976) "P-Complete Approximation Problem," *Journal of Associated Computing Machinery*, 23, pp.555-565.
- Tompkins, J. A. and White, J. A. (1984) *Facilities Planning*, John Wiley and Sons, NY.

Appendix

Proof of Lemma 1. (\Leftarrow) Consider $\mathbf{q}=(\omega,-\omega,\omega,-\omega)$ and $\mathbf{A} = (a_{ij}) = \sum_{m=1}^4 q_m \mathbf{r}_m^0(i,j,k,l) = \omega \mathbf{r}^+(i,j,k) - \omega \mathbf{r}^+(i,j,l) + \omega \mathbf{r}^+(i,k,l) - \omega \mathbf{r}^+(j,k,l)$ where a_{ij} is the coefficient associated with d_{ij} . Then, as shown in Table III, $a_{ij} = a_{ik} = a_{il} = a_{jk} = a_{jl} = a_{kl} = 0$, and consequently $\mathbf{A} = \mathbf{0}$. The proof can be generalized by noting that, if any one of the above four terms of \mathbf{A} , say, $\mathbf{r}^+(i,j,k)$ is replaced by $\mathbf{r}^-(i,j,k)$, then we can maintain \mathbf{A} equal to $\mathbf{0}$ simply by reversing the sign of q_1 .

(\Rightarrow) Suppose there exists $\mathbf{q}=(q_1,q_2,q_3,q_4)$ such that $\mathbf{A} = (a_{ij}) = \sum_{m=1}^4 q_m \mathbf{r}_m^0(i,j,k,l) = \mathbf{0}$.

Since each column in Table III contains exactly two nonzero elements, they must be of equal magnitude with opposite signs. Therefore, $q_m = \pm\omega$ for some $\omega>0$ ($1 \leq m \leq 4$). ■

Table III
Coefficients of \mathbf{A} .

	d _{ij}	d _{ik}	d _{il}	d _{jk}	d _{jl}	d _{kl}
$\omega \mathbf{r}^+(i,j,k)$	ω	$-\omega$		ω		
$-\omega \mathbf{r}^+(i,j,l)$	$-\omega$		ω		$-\omega$	
$\omega \mathbf{r}^+(i,k,l)$		ω	$-\omega$			ω
$-\omega \mathbf{r}^+(j,k,l)$				$-\omega$	ω	$-\omega$
A	0	0	0	0	0	0

Proof of Lemma 3. By Lemma 2, the r th row of \mathbf{H} can be represented as a linear combination of the rows of $-\mathbf{G}$ as well. Thus, $\det \mathbf{R}_{rs}^1 = 0$ for all r and s . ■

Proof of Lemma 5. Let \mathbf{M}_i denote the i th row of a matrix \mathbf{M} . Also, let $\Psi(r)$ be the set

of row numbers in G such that $H_r = \sum_{k \in \Psi(r)} \alpha_k G_k$, where $\alpha_k \neq 0$. (Note that $\Psi(r)$ cannot be an empty set due to Lemma 2.) Recall that H_r and G_k , $k \in \Psi(r)$, are both composed of 0 or ± 1 . Furthermore, neither H_r nor G_k , $k \in \Psi(r)$, is equal to $\mathbf{0}$, and by Lemma 1, all the α_k 's are equal in magnitude. Hence, α_k is equal to ± 1 for all $k \in \Psi(r)$.

Consider the i th row of a matrix M . As shown in Finkbeiner (1966, p.102), if M_i can be expressed as the sum of two vectors, say, R_1 and R_2 , then $\det M = \det M^1 + \det M^2$ where M^1 (M^2) is obtained by replacing the i th row of M by R_1 (R_2). Consequently, $\det R_{rs}^3 = \sum_{k \in \Psi(r)} \det R_{ks}$, where R_{ks} denotes the matrix P in which the s th row of $-G$ is replaced by $\alpha_k G_k$, $k \in \Psi(r)$. Hence, if $s \notin \Psi(r)$, then $\det R_{ks} = 0$ for all $k \in \Psi(r)$, which implies that $\det R_{rs}^3 = 0$. However, if $s \in \Psi(r)$, then $\det R_{ks} = -\alpha_k \det P$ if $k=s$, and 0 otherwise. Thus, $\det R_{rs}^3 = -\alpha_s \det P = \pm \det P$. Therefore, $\det R_{rs}^3 = 0$ or $\pm \det P$ for all r and s . ■

Proof of Lemma 8. Consider an element matrix $\bar{D}=(\bar{d}_{ij})$. Suppose stations i and j in \bar{D} are interchanged to obtain $\bar{D}'=(\bar{d}'_{ij})$. Let us define the *element column (row)* of size n as the column (row) in which each element of $\{0,1,2,\dots,n-1\}$ appears once and only once. For all k except $k=i$ or j , column k in \bar{D}' remains an element column, since only d_{ik} and d_{jk} are interchanged within column k . The elements in columns i and j in \bar{D} move to columns j and i in \bar{D}' , respectively, although the positions of d_{ij} and d_{ji} are interchanged. Two diagonal elements d_{ii} and d_{jj} become d'_{jj} and d'_{ii} , respectively, and hence maintain all diagonal elements in \bar{D}' as 0. Hence, columns i and j in \bar{D}' are also element columns. Therefore, all columns in \bar{D}' are element columns. By the same argument, all rows in \bar{D}' are element rows. Furthermore, all diagonal elements in \bar{D}' remain 0. Therefore, \bar{D}' is also an element matrix. ■

Proof of Theorem 7. Let

$S(i,j)$ be a layout in which station i immediately precedes station j in the clockwise direction,

$S(j,i)$ be the same layout as $S(i,j)$ except that the station i and j are interchanged,

$Z[S]$ denote the material handling requirement of layout S ,

$D=(d_{ij})$ be the station distance matrix determined by $S(i,j)$,

Θ denote the set of all stations except stations i and j , and

C denote the length of circumference. Then,

$$Z[S(i,j)] = f_{ij}d_{ij} + \sum_{k \in \Theta} f_{ik}d_{ik} + \sum_{k \in \Theta} f_{jk}d_{jk} + f_{ji}(C-d_{ij}) + \sum_{k \in \Theta} f_{ki}d_{ki} + \sum_{k \in \Theta} f_{kj}d_{kj} + \sum_{k \in \Theta} \sum_{m \in \Theta} f_{km}d_{km}.$$

$$Z[S(j,i)] = f_{ij}(C-d_{ij}) + \sum_{k \in \Theta} f_{ik}(d_{ik}-d_{ij}) + \sum_{k \in \Theta} f_{jk}(d_{jk}+d_{ij}) + f_{ji}d_{ij} + \sum_{k \in \Theta} f_{ki}(d_{ki}+d_{ij}) + \sum_{k \in \Theta} f_{kj}(d_{kj}-d_{ij}) + \sum_{k \in \Theta} \sum_{m \in \Theta} f_{km}d_{km}.$$

Hence, $Z[S(i,j)] - Z[S(j,i)]$

$$\begin{aligned} &= 2f_{ij}d_{ij} - Cf_{ij} + \sum_{k \in \Theta} f_{ik}d_{ij} - \sum_{k \in \Theta} f_{jk}d_{ij} + Cf_{ji} - 2f_{ji}d_{ij} - \sum_{k \in \Theta} f_{ki}d_{ij} + \sum_{k \in \Theta} f_{kj}d_{ij} \\ &= 2d_{ij}(f_{ij}-f_{ji}) - C(f_{ij}-f_{ji}) + d_{ij} \left\{ \sum_{k \in \Theta} f_{ik} - \sum_{k \in \Theta} f_{jk} - \sum_{k \in \Theta} f_{ki} + \sum_{k \in \Theta} f_{kj} \right\} \\ &= (f_{ij}-f_{ji})(2d_{ij}-C) + d_{ij} \{ \mu(i) - \mu(j) - \lambda(i) + \lambda(j) - 2(f_{ij}-f_{ji}) \} \\ &= -C(f_{ij}-f_{ji}) = 0, \text{ if } f_{ij} = f_{ji}, \end{aligned}$$

$$\text{where } \lambda(i) = \sum_{k=1}^n f_{ki}, \text{ and } \mu(i) = \sum_{k=1}^n f_{ik}. \quad \blacksquare$$

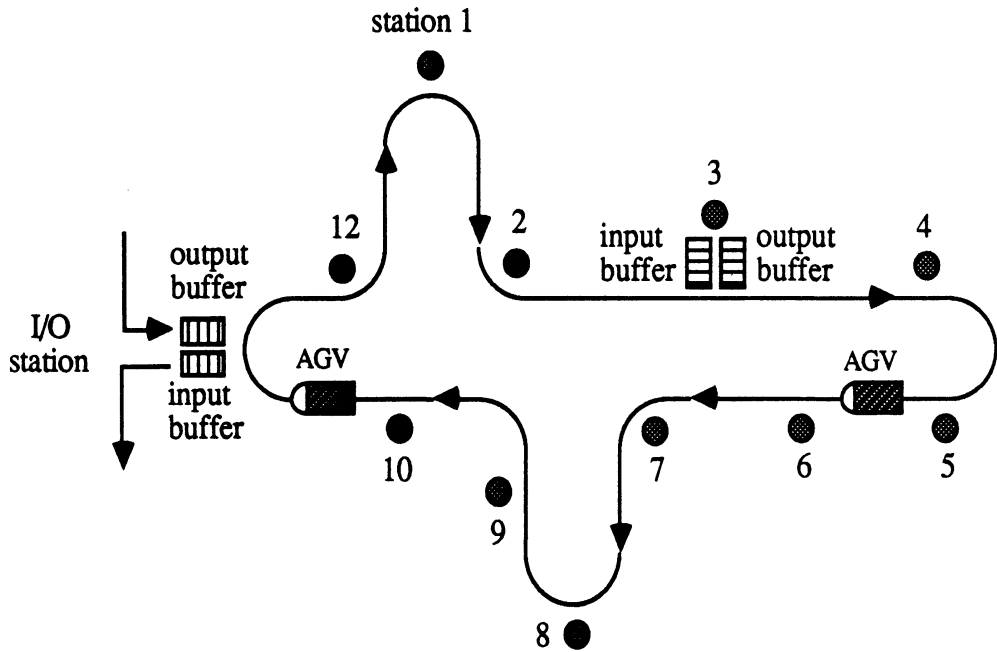


Figure 1. A closed loop manufacturing system with two vehicles and 12 stations.

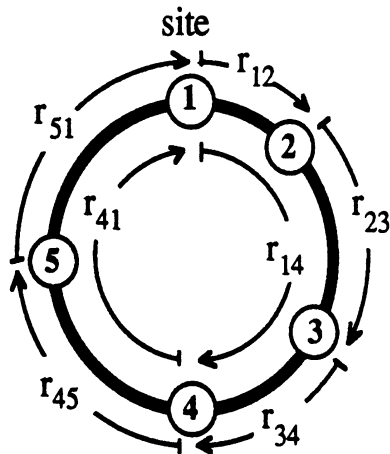
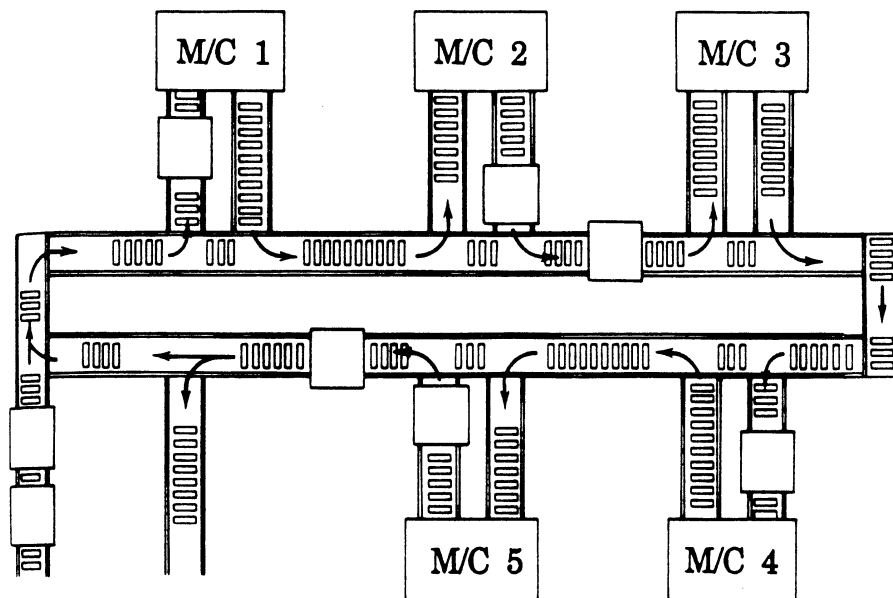
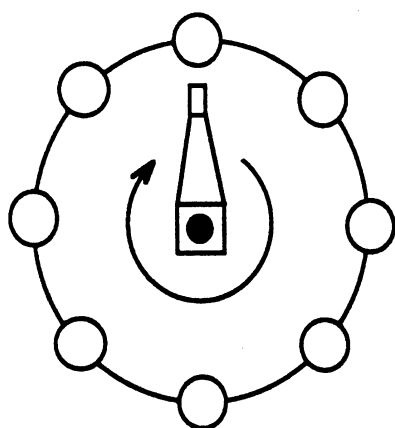


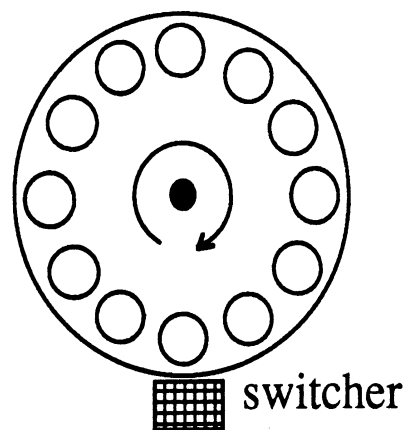
Figure 2. The circular representation of a CLP with five sites, where r_{14} is equal to r_{41} , since site 1 and site 4 are symmetrically located around the circle.



(a) Machine layout in a conveyor-supported manufacturing system.
 (Adapted from Tompkins and White 1984, p.407.)



(b) Machine layout for a unidirectional robot arm.



(c) Tool layout on a turret.

Figure 3. Applications of the circular layout problem.

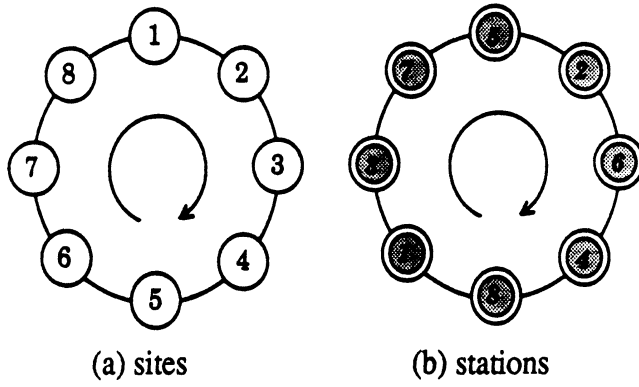


Figure 4. A layout example in the ED-CLP with eight stations.

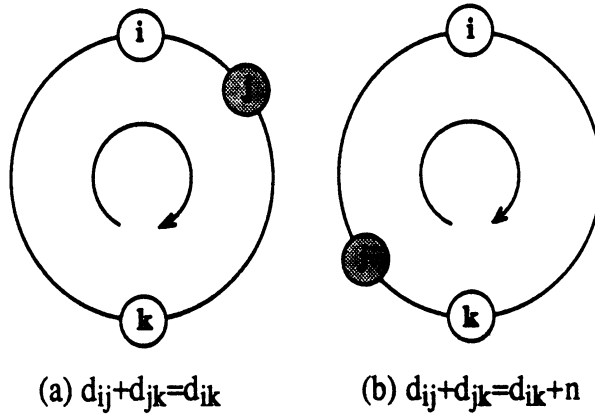


Figure 5. Precedence relationships of three stations around a circle.

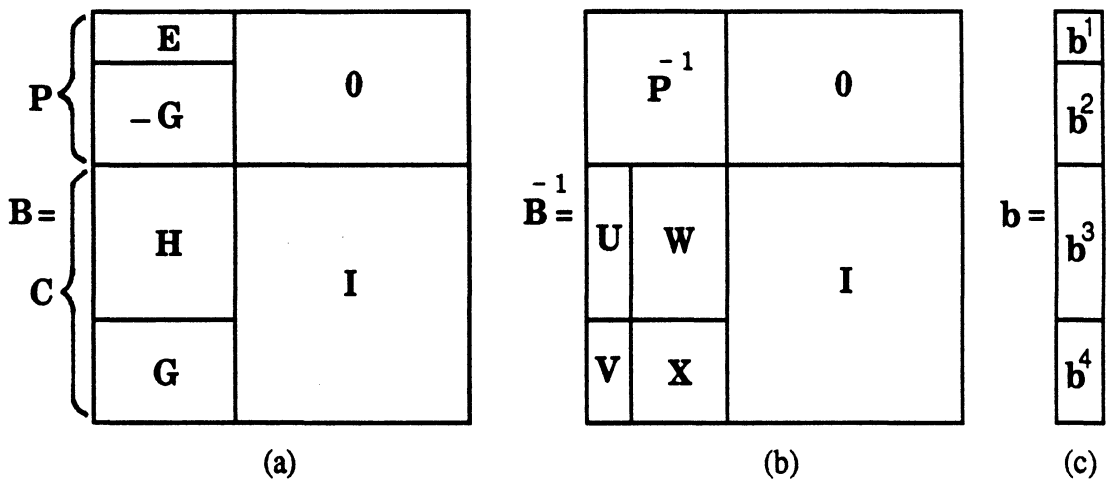


Figure 6. The structure of the basis of LP*.

Table I
Examples for three types of matrices.

0 1 2 3 4	0 2 4 3 1	0 1 2 3 4
2 0 3 4 1	3 0 2 1 4	4 0 1 2 3
4 2 0 1 3	1 3 0 4 2	3 4 0 1 2
3 4 1 0 2	2 4 1 0 3	2 3 4 0 1
1 3 4 2 0	4 1 3 2 0	1 2 3 4 0

(a) an element matrix (b) a circular element matrix (c) a standard matrix

Table II
Comparison of CPU seconds between Net Flow (NF) bound
and Gilmore-Lawler (GL) bound.

		Symmetric Flow				Random Flow							
		Even		Medium		Uneven		Even		Medium		Uneven	
		NF	GL	NF	GL	NF	GL	NF	GL	NF	GL	NF	GL
(N=5)	S ⁽¹⁾	.006	.019	.009	.010	.006	.015	.005	.005	.006	.006	.006	.006
		.008	.024	.005	.023	.004	.019	.002	.002	.002	.002	.003	.003
		.002	.021	.002	.019	.002	.018	.005	.005	.002	.004	.002	.003
	M	.003	.021	.005	.023	.005	.017	.002	.006	.003	.005	.004	.007
		.004	.017	.004	.020	.002	.017	.008	.016	.006	.014	.006	.011
		.004	.023	.002	.023	.002	.025	.007	.014	.004	.010	.002	.008
	D	.006	.024	.004	.022	.002	.017	.006	.012	.005	.011	.004	.008
		.004	.017	.003	.010	.004	.013	.003	.006	.003	.003	.002	.003
		.008	.017	.004	.013	.004	.012	.006	.008	.004	.006	.002	.005
(N=8)	S	.285	5.621	.146	5.037	.152	4.945	.075	.204	.070	.182	.044	.107
		.071	5.781	.075	5.625	.109	5.188	.209	.406	.183	.258	.127	.336
		.128	6.860	.052	6.447	.069	5.800	.089	.165	.099	.192	.065	.085
	M	.086	5.429	.081	5.222	.068	4.404	.339	.747	.256	.482	.225	.490
		.146	5.161	.186	5.400	.073	5.057	.114	.415	.102	.402	.068	.365
		.200	6.236	.170	5.859	.074	4.979	.344	.722	.138	.469	.133	.350
	D	.069	4.446	.029	4.398	.040	4.209	.292	.642	.304	.421	.161	.370
		.150	4.861	.124	4.945	.064	4.465	.155	.460	.090	.322	.054	.238
		.148	4.639	.088	4.612	.056	4.041	.196	.489	.209	.402	.163	.420
(N=12)	S	11.7	NA ⁽²⁾	5.4	NA	15.7	NA	18.8	142.8	11.0	111.9	15.0	156.3
		29.4		19.6		9.3		23.9	145.0	27.0	182.0	11.6	129.5
		41.2		33.8		18.8		109.5	626.4	46.0	363.4	94.2	391.7
	M	38.8	NA	46.2	NA	28.3	NA	27.3	242.8	33.4	235.8	26.3	355.3
		18.6		32.3		8.5		62.3	311.7	45.1	215.0	33.7	270.4
		23.4		20.4		17.7		68.4	402.1	54.2	317.2	35.6	113.9
	D	80.3	NA	50.1	NA	52.9	NA	56.3	344.2	64.9	249.5	26.6	151.7
		43.7		38.5		28.0		45.9	307.0	32.0	238.1	22.0	110.8
		27.9		24.9		27.1		44.2	280.9	31.6	231.2	19.3	223.4

(1) Flow Density: S = sparse, M = medium, D = dense.

(2) exceeds one hour time limit.