

AD-A053 202

STANFORD UNIV CALIF DEPT OF OPERATIONS RESEARCH  
CIRCULAR ONES AND CYCLIC STAFFING.(U)  
NOV 77 J J BARTHOLDI, J B ORLIN, H D RATLIFF

F/G 15/5

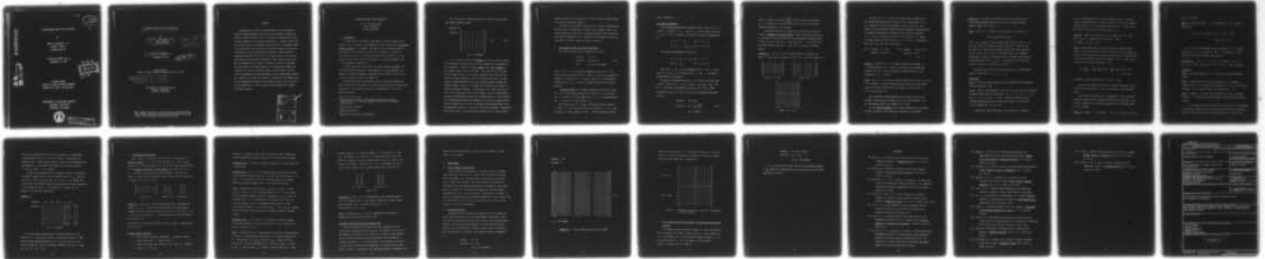
N00014-75-C-0493

UNCLASSIFIED

TR-31

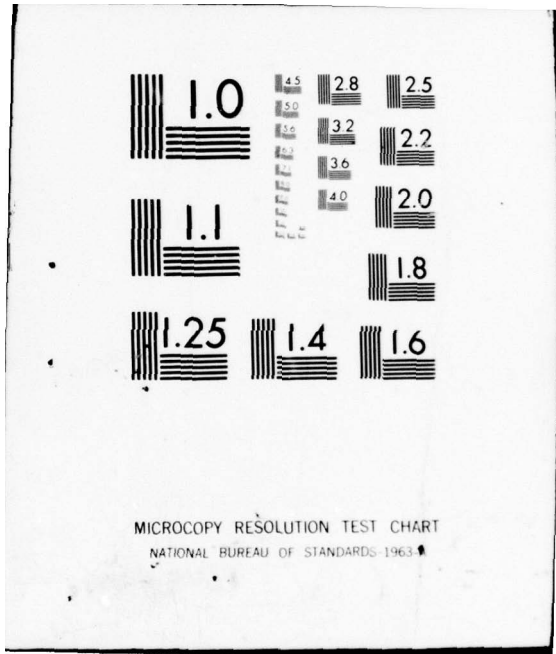
NL

1 of 1  
AD  
A053202



END  
DATE  
FILMED  
6 - 78

DDC



MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

AD A 053202

AD No.             
DDC FILE COPY

12  
R

CIRCULAR ONES AND CYCLIC STAFFING

BY

JOHN J. BARTHOLDI, III  
JAMES B. ORLIN  
H. DONALD RATLIFF

TECHNICAL REPORT NO. 31  
NOVEMBER 20, 1977

DDC  
APR 26 1978  
F

PREPARED UNDER  
OFFICE OF NAVAL RESEARCH CONTRACT  
N00014-75-C-0493 (NR-042-264)

DEPARTMENT OF OPERATIONS RESEARCH  
STANFORD UNIVERSITY  
STANFORD, CALIFORNIA



This document has been approved  
for public release and sale; its  
distribution is unlimited.

6 CIRCULAR ONES AND CYCLIC STAFFING,

By

10 John J./Bartholdi, III,  
James B./Orlin  
H. Donald/Ratliff

71 20 Nov 77

72 27p

9 Technical Report, No. 31  
November 20, 1977

14 TR-31

Prepared Under  
Office of Naval Research Contract N00014-75-C-0493\*

15 N00014-75-C-0493,  
N00014-76-C-0096

Department of Operations Research  
Stanford University  
Stanford, California

\* Also partially supported by National Science Foundation Grant  
ENG-76-12266, Army Research Office Grant DAHCO4-75-G-0150 and  
Office of Naval Research Grant N00014-76-C-0096.

ABSTRACT

A large class of cyclic staffing problems, when formulated as (linear) integer programs, possess zero-one constraint matrices for which the ones in each row occur in consecutive components (the first and last components are considered consecutive). Included within this class is the problem of minimizing the linear cost of assigning workers to a multi-period cyclic schedule such that the demand in each period is satisfied, and each person works a shift of a common number of consecutive periods and is idle for the other periods (the first and last periods are considered consecutive). Any problem in this class may be transformed via a change of variables so that the resulting constraint matrix is, after deletion of a distinguished column, the transpose of a node-arc incidence matrix. The problem can then be solved in polynomial time parametrically in the distinguished variable as a sequence of network flow problems. Alternately, the optimal value of the distinguished variable can be found to within integer roundoff as its optimal value in the associated linear program with integer constraints ignored.

ACCESSION for	
BTIS	White Section <input checked="" type="checkbox"/>
BDC	Buff Section <input type="checkbox"/>
UNANNOUNCED	<input type="checkbox"/>
JUSTIFICATION	
BY	
DISTRIBUTION/AVAILABILITY NOTES	
Dr	SPECIAL
A	

## CIRCULAR ONES AND CYCLIC STAFFING\*\*

John J. Bartholdi, III\*

James B. Orlin†

H. Donald Ratliff@

### 1. Introduction

In recent years much interest has focused on various problems that arise in cyclic staffing. One fundamental problem, called the (k,m)-cyclic staffing problem, is to minimize the linear cost of assigning persons to an m-period cyclic schedule so that (1) the demand  $d_j$  in period j is satisfied for each j, and (2) each person works a shift of k consecutive periods and is idle for the other m-k periods (periods 1 and m are considered consecutive).

In 1955, Gross [11] solved the (2,m)-cyclic staffing problem. The (5,7)-cyclic staffing problem (5-day workers in a weekly schedule) was studied by Tibrewala, Philippe, and Brown [16] and others [3,5,14]. For a survey of cyclic staffing, see Baker [2].

The related problem of minimizing costs of assigning workshifts in which schedules are not assumed to be cyclic has been solved by Veinott and Wagner [18]. They solved the problem by transforming it into a network flow problem.

---

\*\* This work was supported by ONR contracts N0014-76-C-0096 and N0014-75-C-0493, by ARO grant DAHCO4-75-G-0150 and by NSF grant ENG 76-12266.

\* University of Michigan.

† Stanford University.

@ University of Florida at Gainesville.

The (5,7)-cyclic staffing problem may be stated as the following (linear) integer program:

Minimize  $ax$

$$\text{Subject to } \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} x \geq d \quad (1.1)$$

$x \geq 0$  and integer.

A 0-1 vector is said to be circular if its 1's occur in consecutive components, where the first and last components are considered to be consecutive. A matrix is called column (resp., row) circular if its columns (resp., rows) are circular. The matrix in (1.1) is both row and column circular. The column circularity stems from the fact that each column represents a shift of consecutive periods. For this reason, constraint matrices for cyclic staffing problems usually are column circular. However, it is their row circularity that leads to an efficient solution technique. In particular, it is shown in Section 2 that if the constraint matrix is row circular, then the problem may be transformed unimodularly so that the resulting matrix is, after deletion of a distinguished column, the transpose of a node-arc incidence matrix. The problem may then be solved in polynomial time parametrically in the distinguished variable as a series of network flow problems. Alternately, the optimal value of the distinguished variable may be found to within

integer roundoff as its optimal value in the associated linear program with integer constraints ignored.

In Section 3, we present a large class of column circular matrices that are, up to a permutation of columns, row circular as well. Included within this class are the constraint matrices derived from the  $(k,m)$ -cyclic staffing problem. Finally, several examples to which the algorithms of Section 2 apply are given in Section 4.

## 2. The Integer Program and Solution Techniques

The integer program that we consider has the form:

$$\begin{aligned} &\text{Minimize} && ax + by \\ &\text{Subject to} && Ax + By \geq d && (2.1) \\ &\text{and} && x, y \geq 0 \text{ and integer,} \end{aligned}$$

where  $a$  and  $b$  are non-negative integral vectors, and  $d$  is a strictly positive integral vector. (There is no loss of generality in assuming each component of  $d$  positive, since in the following any constraint with  $d_j \leq 0$  would be trivially satisfied and could be deleted.)

A staffing problem is an integer program of the form (2.1) with  $A$  an  $m \times n$  row-circular matrix with at least one 1 in every row, and  $B$  a matrix of the form  $\begin{bmatrix} I \\ 0 \end{bmatrix}$  with  $I$  a  $p \times p$  identity matrix and  $0$  an  $(m-p) \times p$  matrix of all 0's.

The vector of all 1's (resp., 0's) will be denoted simply as  $\underline{1}$  (resp.,  $\underline{0}$ ). If a circular vector is not equal to  $\underline{0}$  or  $\underline{1}$ , the its first 1 (resp., last 1) is that 1 which immediately follows

(resp., precedes) a 0.

### The Change of Variables

For a staffing problem with constraint matrix  $[A \ B]$ , let  $l_j = n$  if row  $j$  of matrix  $A$  is 1 and let  $l_j$  be the index of the last 1 in row  $j$  of  $A$  otherwise. Consider the change of variables given by

$$\begin{aligned} v_j &= x_1 + \cdots + x_j && \text{for } j = 1, \dots, n \\ w_j &= x_1 + \cdots + x_{l_j} + y_j && \text{for } j = 1, \dots, p. \end{aligned}$$

The inverse transformation is given by

$$\begin{aligned} x_1 &= v_1; \quad x_j = v_j - v_{j-1} && \text{for } j = 2, \dots, n \\ y_j &= w_j - v_{l_j} && \text{for } j = 1, \dots, p. \end{aligned}$$

Note that  $v_j$  and  $w_i$  are integral for all  $i$  and  $j$  if and only if  $x_j$  and  $y_i$  are integral for all  $i$  and  $j$ . (The above transformation is unimodular.)

We want to distinguish variable  $v_n$  for reasons that will later be clear. Thus we let  $u = (u_i) = (v_1, v_2, \dots, v_{n-1}, w_1, \dots, w_p)$ , and let  $T$  denote the transformation such that  $\begin{pmatrix} x \\ y \end{pmatrix} = T \begin{pmatrix} u \\ v_n \end{pmatrix}$ . Then substituting  $T \begin{pmatrix} u \\ v_n \end{pmatrix}$  for  $\begin{pmatrix} x \\ y \end{pmatrix}$  in (2.1), the staffing problem takes the form:

$$\begin{aligned} \text{Minimize} \quad & \bar{a}u + \bar{a}_n v_n \\ \text{Subject to} \quad & \bar{A}u + cv_n \geq \begin{pmatrix} d \\ 0 \\ 0 \end{pmatrix} \\ & u, v_n \text{ integer,} \end{aligned} \tag{2.2}$$

where  $\bar{A}$  denotes the matrix  $\begin{bmatrix} A & B \\ I & O \\ O & I \end{bmatrix}^T$  with the last column (denoted by  $c$ ) deleted, and  $\bar{a}$  denotes the vector  $[a \ b]^T$  with the last component (denoted by  $\bar{a}_n$ ) deleted.

A staffing problem that has been put in the form (2.2) will be called a transformed staffing problem. The key observation that leads to efficient solution techniques for the transformed staffing problem is that  $\bar{A}$  has at most one 1 and at most one -1 in any row, with all other elements equal to 0. Thus  $\bar{A}^t$  is a node-arc incidence matrix.

Example 2.1. Let  $[A \ B]$  be the constraint matrix of Figure 2.1a.

Then  $T$  is the matrix of Figure 2.1b and  $[\bar{A} \ c]$  is that in Figure 2.1c.

$$\left[ \begin{array}{cccc|ccc} 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array} \right]$$

Figure 2.3a.  $[A \ B]$

$$\left[ \begin{array}{ccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right]$$

Figure 2.3b.  $T$

$$\left[ \begin{array}{cccccc|c} 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right]$$

Figure 2.3c.  $[\bar{A} \ c]$

The fact that  $\bar{A}$  is a node-arc incidence matrix suggests that the transformed staffing problem may be solved parametrically in  $v_n$  as a sequence of network-flow problems. To this end we let  $D(v)$  denote the linear program obtained from the transformed staffing problem by dropping the integrality constraints and adding the constraint  $v_n = v$ . Let  $P(v)$  denote the dual to  $D(v)$  and let  $z(v)$  denote the common objective value (where  $z(v) = \infty$  if  $D(v)$  is infeasible). Then  $D(v)$  and  $P(v)$  may be written as

$$\begin{array}{ll}
 D(v): \text{ minimize } & \bar{a}u + \bar{a}_n v \\
 & \text{subject to } \bar{A}u \geq d - vc \\
 \\
 P(v): \text{ maximize } & \lambda(d-vc) + \bar{a}_n v \\
 & \text{subject to } \lambda \bar{A} = \bar{a} \\
 & \lambda \geq 0.
 \end{array}$$

Lemma 2.1. Problem  $P(v)$  is a (primal) network flow problem. If  $v$  is integral and  $z(v)$  is finite, then there exist optimal dual variables for  $P(v)$  that are integral and are optimal for  $D(v)$ . Furthermore,  $z(\cdot)$  is convex.

Proof. Since  $\bar{A}^t$  is a node-arc incidence matrix and  $\bar{a}$ ,  $\bar{a}_n$ ,  $d$ ,  $v$ , and  $c$  are integral, it follows from Dantzig [6] that there exist optimal (primal and) dual variables that are integral. The convexity of  $z(\cdot)$  is well known [10].

The transformed staffing problem has thus been reduced to the problem of determining an integral value  $v_n^*$  which minimizes  $z(v)$ , and determining optimal dual variables for  $P(v_n^*)$ .

Let the linear relaxation of an integer program denote the linear program that results from dropping the integrality constraints.

Theorem 2.2. If  $(\begin{smallmatrix} u' \\ v_n' \end{smallmatrix})$  is optimal for the linear relaxation of a transformed staffing problem, then there is an optimal integral solution  $(\begin{smallmatrix} u^* \\ v_n^* \end{smallmatrix})$  with  $|v_n^* - v_n'| < 1$ .

Proof. Since  $z(\cdot)$  is convex, it follows that for integral  $v_n^*$

$$z(v_n^*) \geq \min(z(\lceil v_n' \rceil), z(\lfloor v_n' \rfloor)) ,$$

where  $\lceil v_n' \rceil$  denotes the least integer greater than or equal to  $v_n'$ , and  $\lfloor v_n' \rfloor$  denotes the greatest integer less than or equal to  $v_n'$ . By Lemma 2.1, if there exist optimal solutions that yield objective values  $z(\lceil v_n' \rceil)$  and  $z(\lfloor v_n' \rfloor)$ , they can be taken to be integral. Furthermore,  $D(\lceil v_n' \rceil)$  is feasible so that an optimal solution exists yielding objective value  $z(\lceil v_n' \rceil)$ . It is possible that no solution exists for  $D(\lfloor v_n' \rfloor)$  in which case  $z(\lfloor v_n' \rfloor) = \infty$ .

The above proof reveals an efficient algorithm for the staffing problem. Recall that  $v_n = x_1 + \dots + x_n$ .

Algorithm A

Step 1. Solve the linear relaxation of the staffing problem obtaining an optimal solution  $(\begin{smallmatrix} x' \\ y' \end{smallmatrix})$ .

Step 2. Form two linear programs LP1 and LP2 from the above relaxation by adding respectively the constraints  $x_1 + \dots + x_n = \lceil x_1' + \dots + x_n' \rceil$  and  $x_1 + \dots + x_n = \lfloor x_1' + \dots + x_n' \rfloor$ . An optimal solution for the staffing problem (which can be taken to be integral) is the better of the solutions for LP1 and LP2.

Although the simplex algorithm is very efficient in practice,

there is no guarantee that it runs in polynomial time. In order to obtain a formally efficient algorithm the key is to obtain an optimal integral value  $v_n^*$  in a polynomial number of steps. The next result is useful to this end in providing bounds on  $v_n^*$ .

Lemma 2.3. There is an integer value  $v_n^*$  minimizing  $z(v)$  such that  $v_n^* \leq ld$ . Furthermore  $z(v_n^*) \leq M \cdot (ld)$ , where  $M = \max(a_1, \dots, a_n, b_1, \dots, b_p)$ .

Proof. We may assume that every nonzero optimal integer variable  $x_i^*, y_j^*$  in (2.1) appears with a nonzero coefficient in some tight constraint, since otherwise that variable could be reduced, feasibility maintained and the objective value not increased. Let  $S$  denote the set of indices of the tight constraints. Then

$$\begin{aligned} ld \geq \sum_{i \in S} d_i &= \sum_{i \in S} \left( \left[ \sum_{j=1}^n a_{ij} x_j^* \right] + \left[ \sum_{j=1}^p b_{ij} y_j^* \right] \right) \geq lx^* + ly^* \\ &\geq lx^* = v_n^* . \end{aligned}$$

Furthermore,  $M(ld) \geq M(lx^* + ly^*) \geq ax^* + by^* = z(v_n^*)$ .

There are many ways to search for an integer minimizing a convex function over a bounded set. One is to use Kiefer's [13] fibonacci search.

A more efficient method, when a "subgradient" of the function is available at each point, is binary search (interval bisection). This method is developed for the staffing problem in the following lemma and its corollary.

Lemma 2.4. Suppose  $\lambda'$  is optimal for  $P(v')$ . If  $(\bar{a}_n - \lambda'c)(v - v') \geq 0$ ,

then  $z(v') \leq z(v)$ .

Proof. For any real number  $v$ ,  $\lambda'$  is feasible for  $P(v)$ . then by hypothesis,

$$\begin{aligned} z(v) - z(v') &\geq \lambda'(d-v) + \bar{a}_n v - \lambda'(d-v') + \bar{a}_n v' \\ &= (\bar{a}_n - \lambda'c)(v-v') \geq 0 . \end{aligned}$$

Let  $v_n^*$  denote an integral value that minimizes  $z(v)$ . Suppose  $v_n^* \in [j_1, j_2]$ . Let  $j_3 = \lfloor (j_1 + j_2) / 2 \rfloor$ . Let  $\lambda'$  be optimal for  $P(j_3)$ . Then the following is immediate from Lemma 2.4.

Corollary 2.5. If  $\bar{a}_n - \lambda'c = 0$ , then  $j_3$  minimizes  $z(v)$ . If  $\bar{a}_n - \lambda'c > 0$ , then  $v_n^* \in [j_1, j_3]$ . If  $\bar{a}_n - \lambda'c < 0$ , then  $v_n^* \in [j_3, j_2]$ .

#### Algorithm B

Step 1. Use transformation  $T^{-1}$  to derive the transformed staffing problem.

Step 2. Using binary search based on Corollary 2.5, determine an integral value  $v_n^*$  in  $[0, ld]$  which minimizes  $z(v)$ . At each iteration determine  $z(v)$  by using a network flow algorithm on  $P(v)$ .

Step 3. Transform  $v_n^*$  and the dual variables for  $P(v_n^*)$  via transformation  $T$  to obtain the optimal integral solution for the staffing problem.

It is not the case that any network flow algorithm runs in polynomial time; however, Edmonds and Karp [8] demonstrated that the out-of-kilter-algorithm with scaling does run in polynomial time. In order to use

that algorithm it is necessary to have upper bounds on the arc flows.

Lemma 2.6. Without loss of generality, we may add to problem  $P(v)$  the constraint  $\lambda_j \leq M \cdot (ld) + 1$ , for each  $j$ , where  $M = \max(a_1, \dots, a_n, b_1, \dots, b_p)$ .

Proof. By assumption,  $d \geq 1$ . Thus if for fixed  $v$  any new constraint is tight, then  $z(v) > M \cdot ld$ . Then by Lemma 2.3,  $z(v)$  is not minimum.

In the following theorem we assume that the constraints  $\lambda_j \leq M^*$  have been added to  $P(v)$  where  $M^* = 1 + M \cdot ld$ .

Theorem 2.7. If the out-of-kilter algorithm is used in Step 2 of Algorithm B, then the execution time is  $O((n+m)^2 (\log M^*) (\log ld))$  steps.

Proof. The transformations for steps 1 and 3 may be carried out in  $O((n+m)^2)$  steps. Each iteration using out-of-kilter with scaling requires  $O(e^2 \log q)$  steps, where  $e$  is the number of edges in the network, and  $q$  is the maximum flow allowed in any edge. Since the number of edges is  $m+n+p$ , and  $p \leq n$ , the number of steps in each iteration is  $O((n+m)^2 \log M^*)$ . The number of iterations using binary search is  $\log(ld)$ . Thus the total number of steps is  $O((n+m)^2 (\log M^*) (\log ld))$ .

#### A Special Objective Function

Of special importance for staffing problems is the case for which the cost coefficients are equal to 1. In applications, minimizing such an objective function usually corresponds to minimizing the number of workers.

Let  $S$  denote a staffing problem with constraint matrix  $A$  and

cost coefficients  $a$  with  $a = 1$ . Let  $S'$  be the transformed scheduling problem.

Proposition 2.8. If  $v^*$  is an optimal solution vector for the linear relaxation of  $S'$ , then an optimal solution for  $S'$  is

$$[v^*] = ([v_1^*], \dots, [v_n^*]).$$

Proof. The objective function for  $S'$  is equal to  $v_n$ . Thus the minimum objective value for  $S'$  is at least  $[v_n^*]$ . It suffices to show that  $[v^*]$  is feasible. By construction of  $T$ , there are at most 3 types of linear constraints for  $S'$ .

- i)  $v_j - v_\ell \geq b_i$ ,
- ii)  $v_j - v_\ell + v_n \geq b_i$ ,
- iii)  $v_j \geq b_i$ .

Noting that for all real numbers  $r$  and  $s$ ,  $[r] - [s] \geq [r-s]$ , we see that

$$[v_j^*] - [v_\ell^*] \geq [v_j^* - v_\ell^*] \geq b_i,$$

$$\begin{aligned} [v_j^*] - [v_\ell^*] + [v_n^*] &\geq [v_j^* + v_n^*] - [v_\ell^*] \\ &\geq [v_j^* - v_\ell^* + v_n^*] \geq b_i, \end{aligned}$$

and  $[v_j^*] \geq v_j^* \geq b_i$ .

Remark 2.9. If  $a = 1$ , then  $\bar{a} \geq 0$ . For  $\bar{a} \geq 0$  problem  $D(v_n)$  satisfies the conditions assumed by Dorsey, Hodgeson, and Ratliff [7]. Then  $D(v_n)$  may be solved by their simple recursive substitution scheme.

Furthermore, in the special case of  $a = 1$ , it is possible to solve the problem via parametric programming in  $O(m^2 \log m)$  steps.

Orlin, in an unpublished work, bases this algorithm on determining an optimal basis for  $P(v)$  for each  $v \in [0, 1d]$ . In practice, for problems with  $m$  taking values between 24 and 500, this algorithm takes approximately  $m$  iterations which results in an execution time of  $c \cdot m \log m$ , where  $c$  is a constant.

It might seem plausible that the rounding procedure of Proposition 2.8 might be generalized, or that the optimal value for  $v'_n$  should be rounded up. Such is not the case. If  $v^*$  and  $v'$  are optimal for respectively a transformed scheduling problem and its linear relaxations, it is possible for some  $j$  that  $|v_j^* - v_j'| \geq 1$  and that  $v_n^* < v_n'$ . Example 2.2 illustrates both points.

Example 2.2

$$\begin{array}{l}
 \text{Minimize} \quad 21x_1 + 15x_2 + 15x_3 + \dots + 15x_8 \\
 \text{Subject to} \quad \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_8 \end{bmatrix} \geq \begin{bmatrix} 2 \\ \vdots \\ 2 \end{bmatrix} \\
 x \geq 0 \text{ and integer.}
 \end{array}$$

The unique optimal solution for the linear relaxation of the problem is  $x' = [0, 2/3, 2/3, \dots, 2/3]$ , with objective value 70. The unique optimal integral solution is  $x^* = [2, 0, 0, 0, 2, 0, 0, 0]$  with objective value 72. After the change of variables  $v_1^* - v_1' = 2$ , and  $v_n^* - v_n' = 1x^* - 1x' = -2/3$ .

### 3. Recognizing Circular Rows

Let  $x$  and  $y$  be vectors with circular 1's. We say that  $x$  properly contains  $y$  if  $x \geq y$  and the vector  $x-y$  does not have circular 1's. Following Tucker [17], we say that column circular matrix  $A$  has properly compatible circular columns if no column of  $A$  is 0 or 1 and if no column properly contains another. The matrix in Figure 3.1a has properly compatible circular columns while the matrices in Figures 3.1b and 3.1c do not.

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Figure 3.1a

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}$$

Figure 3.1b

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Figure 3.1c

Remark 3.1. Suppose in a cyclic staffing problem the workshifts are labeled  $s_1, \dots, s_n$ . Then the constraint matrix will have properly compatible circular columns if and only if there does not exist  $i$  and  $j$  such that the persons in  $s_i$  start working before and finish after the persons in  $s_j$ .

For matrices with properly compatible circular columns, a natural order suggests itself.

#### Column Ordering Algorithm

- 1) Order columns in groups, where group  $i$  consists of those columns whose first 1 appears in row  $i$ .
- 2) Within each group, order columns so that column  $A_j$  precedes column  $A_\ell$  if  $A_j \leq A_\ell$ .

Henceforth, we assume, without loss of generality that a matrix with properly compatible circular columns has its columns ordered as above.

Observation 3.2. A matrix with properly compatible circular columns has circular rows.

Proposition 3.3. Let  $P$  be an integer program of the form (2.1) with column circular constraint matrix  $A$  and cost coefficient vector  $a = 1$ . Then  $P$  may be written so that the constraint matrix has properly compatible circular columns. Thus  $P$  is a staffing problem.

Proof. Suppose there are columns  $A_i$  and  $A_j$  in matrix  $A$  with  $A_i \leq A_j$ . Then if  $x^*$  is an optimal solution for  $P$ , it is possible to form (another) optimal solution by replacing  $x_j^*$  by  $x_j^* + x_i^*$  and replacing  $x_i^*$  by 0. This replacement will neither affect cost nor alter feasibility. Thus we may eliminate any columns  $A_i$  (and corresponding variable  $x_i$ ) if there exists  $A_j$  with  $A_i \leq A_j$ . Repeating this procedure leaves a matrix with properly compatible circular columns.

Proposition 3.4. If  $A$  is an  $m \times n$  matrix with distinct columns and properly compatible circular columns, then  $n \leq 2m$ . Moreover, the bound is sharp for  $m \geq 3$ .

Proof. Consider the set  $S$  consisting of the  $m(m-1)$  circular columns of length  $m$ , excluding the column of all 0's and the column of all 1's. Define sets  $S_1, \dots, S_{2m}$  such that  $S_{2j-1}$  consists of those columns in  $S$  with an odd number of 1's, and whose middle 1 occurs in row  $j$ , and such that  $S_{2j}$  consists of those columns in  $S$  with

an even number of 1's and whose middle 1's occur in rows  $j$  and  $j+1$ . (See Figure 3.2.) Then set  $S$  is the union of the  $S_j$ 's and matrix  $A$  contains at most one column from each set  $S_j$ . Thus  $A$  has at most  $2m$  columns. The bound of  $n = 2m$  is obtainable for  $m \geq 3$ , since  $A$  may consist of all circular columns with one or two 1's.

$$\begin{array}{cc}
 \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \\
 S_3 \text{ for } m = 5 & S_{10} \text{ for } m = 5
 \end{array}$$

Figure 3.2.

Corollary 3.5. Let  $S$  be a staffing problem with constraint matrix  $[A \ B]$  for which matrix  $A$  has properly compatible circular columns. Then Algorithm B finds an optimal solution for  $S$  in  $O(m^2(\log M^*)(\log ld))$  steps.

Proof. By Proposition 3.4,  $n \leq 2m$ . Applying this inequality to Proposition 2.7 yields the desired result.

#### Recognizing Other Matrices with Circular Rows

There are matrices with circular rows and columns that do not have properly compatible circular columns. (See Figures 3.1b and 3.1c.) Furthermore, many matrices have circular rows without circular columns. Thus it would be desirable to have an algorithm which efficiently determines if matrix  $A$  is row circular up to a permutation of columns. In [17], Tucker presents such an algorithm. The algorithm which takes  $O(m^2n)$  steps, is a variation of the algorithm presented by Fulkerson and

Gross in [9] for determining if a matrix has the property of consecutive 1's in columns.

#### 4. Applications

##### A. Cyclic Staffing with Overtime

A basic staffing problem involves a facility such as a hospital that operates 24 hours each day. Assume that there are fixed hourly staff requirements  $d_i$ , and that there are three basic work shifts, each of eight hours duration: 0700-1500, 1500-2300, and 2300-0700. Overtime of up to an additional eight hours is possible for each shift. What is the least cost assignment of personnel such that all staff requirements are met? This problem may be formulated as in Figure 4.1, where the constraint matrix has properly compatible circular columns. Thus the problem may be solved either by a bounded series of network flow problems or by linear programming.

##### B. Days-off Scheduling

A problem studied by Brownell and Lowerre [4] is to minimize the total workforce necessary to meet daily staffing requirements, where each worker is guaranteed two days off each week, including every other weekend. One variant of the problem is to add the restriction that the days off in each week are consecutive, and to allow the hiring of part time employees for any day. The problem may then be formulated as

$$\begin{aligned} \text{Minimize} \quad & ax + by \\ \text{Subject to} \quad & Ax + y \geq d \\ & x, y \geq 0 \text{ and integer,} \end{aligned}$$

Minimize  $ax$

Subject to

07	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 1 1 1 1 1 1 1 1 1
08	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 1 1 1 1 1 1 1 1
09	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 0 1 1 1 1 1 1 1
10	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 0 0 1 1 1 1 1 1
11	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 1 1 1 1 1
12	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1 1 1
13	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 1 1
14	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 1
15	0 1 1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
16	0 0 1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
17	0 0 0 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
18	0 0 0 0 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
19	0 0 0 0 0 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
20	0 0 0 0 0 0 1 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
21	0 0 0 0 0 0 0 1 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
22	0 0 0 0 0 0 0 0 1 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
23	0 0 0 0 0 0 0 0 0 1	1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0
24	0 0 0 0 0 0 0 0 0 0	0 1 1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1
01	0 0 0 0 0 0 0 0 0 0	0 0 1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1
02	0 0 0 0 0 0 0 0 0 0	0 0 0 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1
03	0 0 0 0 0 0 0 0 0 0	0 0 0 0 1 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1
04	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 1 1 1 1 1	1 1 1 1 1 1 1 1 1 1
05	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1 1 1	1 1 1 1 1 1 1 1 1 1
06	0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 1 1	1 1 1 1 1 1 1 1 1 1

$x$   
IV  
 $a$

$x \geq 0$  and integer.

Figure 4.1. A cyclic staffing problem with overtime.

where row circular matrix  $A$  is the matrix in Figure 4.2. Thus the problem is a staffing problem as described in Section 2, and may be solved by either Algorithm A or Algorithm B.

First week	1 1 1 1 1 1	1 1 1 1 1 0
	1 1 1 1 1 1	1 1 1 1 0 0
	1 1 1 1 1 1	1 1 1 0 0 1
	1 1 1 1 1 1	1 1 0 0 1 1
	1 1 1 1 1 1	1 0 0 1 1 1
	0 0 0 0 0 0	0 0 1 1 1 1
	0 0 0 0 0 0	0 1 1 1 1 1
Second week	0 1 1 1 1 1	1 1 1 1 1 1
	0 0 1 1 1 1	1 1 1 1 1 1
	1 0 0 1 1 1	1 1 1 1 1 1
	1 1 0 0 1 1	1 1 1 1 1 1
	1 1 1 0 0 0	1 1 1 1 1 1
	1 1 1 1 0 0	0 0 0 0 0 0
	1 1 1 1 1 0	0 0 0 0 0 0

Figure 4.2. Constraint matrix for a variant of the Brownell and Lowerre problem.

C. Cyclic Staffing with Linear Penalties for Understaffing and Overstaffing

One staffing problem considered by Baker [2] is one for which the period demands are not rigid. Instead there is a linear penalty  $b_i$  for understaffing, and a linear (possibly negative) penalty  $c_i$  for overstaffing period  $i$ . Since the amount of overstaffing is  $d - Ax - y$ , the problem may be stated as

Minimize  $ax + by + c(d-Ax-y)$

Subject to  $Ax + y \geq d,$

$x, y \geq 0$  and integer.

If  $A$  is the matrix desired from the  $(k,m)$ -staffing problem or any other row circular matrix, the problem may be solved by the algorithms of Section 2.

#### REFERENCES

- [1] Baker, K. R., "Scheduling a Full-Time Workforce to Meet Cyclic Staffing Requirements," Management Science, Vol. 20, No. 12 (1974), pp. 1561-8.
- [2] Baker, K. R., "Workforce Allocation in Cyclical Scheduling Problems," Operational Research Quarterly, Vol. 27, No. 1 (1976), pp. 155-167.
- [3] Bartholdi, J. J. and H. D. Ratliff, "Unnetworks, with Applications to Idle Time Scheduling," Research Report No. 77-4, Industrial and Systems Engineering Department, University of Florida, Gainesville, Florida.
- [4] Brownell, W. S. and J. M. Lowerre, "Scheduling of Workforce Required for Continuous Operation Under Alternative Labor Policies," Management Science, Vol. 22, No. 5 (1976), pp. 597-605.
- [5] Chen, D. S., "A Simple Manpower Scheduling Algorithm for Two Consecutive Days Off," presented at the ORSA/TIMS Joint National Meeting, Miami, Florida, 1976.
- [6] Dantzig, G. B., "An Application of the Simplex Method to the Transportation Problem," in T. C. Koopmans (ed.): Activity Analysis of Production and Allocation, J. Wiley and Sons, New York, (1951).
- [7] Dorsey, R. C., T. J. Hodgeson, H. D. Ratliff, "An Efficient Integer Programming Algorithm for a Multi-facility, Multi-production Production Scheduling Problem," Research Report No. 73-8, January 1973, Industrial and Systems Engineering Department, University of Florida, Gainesville, Florida.

- [8] Edmonds, J. and R. M. Karp, "Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems," Journal for the Association of Computing Machinery, Vol. 19, No. 2, (1972), pp. 248-64.
- [9] Fulkerson, D. R., and O. A. Gross, "Incidence Matrices and Interval Graphs," Pacific Journal of Mathematics, Vol. 15, (1965), pp. 835-855.
- [10] Gaddum, J. W., A. J. Hoffman, and D. Sokolowsky, "On the Solution of the Caterer Problem," Naval Research Logistics Quarterly, Vol. 1, No. 3 (1954), pp. 223-229.
- [11] Gross, O. A., "A Simple Linear Programming Problem Explicitly Solvable in Integers," Project Rand Document D-3129 (1955).
- [12] Iri, M., "A Criterion for the Reducibility of a Linear Programming Problem to a Linear Network Flow Problem," RAAG Research Notes, Third Series, No. 98, February, 1966.
- [13] Kiefer, J., "Sequential Minimax Search for a Maximum," Proceedings of the American Mathematical Society, Vol. 4, (1953), pp. 502-506.
- [14] Orlin, J. B., "Quick Optimal Weekly Scheduling with Two Consecutive Days Off," Technical Report 77-1, Department of Operations Research, Stanford University, Stanford, Calif. (1977).
- [15] Segal, M., "The Operator Scheduling Problem: A Network Flow Approach," Operations Research, Vol. 22, No. 4 (1974), pp. 803-24.
- [16] Tibrewala, R., D. Philipee, and J. Browne, "Optimal Scheduling of Two Idle Periods," Management Science, Vol. 19, No. 1 (1972), pp. 71-5.

- [17] Tucker, A., "Matrix Characterizations of Circular Arc Graphs,"  
Pacific Journal of Mathematics, Vol. 39, No. 2 (1971),  
pp. 535-45.
- [18] Veinott, A. F., Jr. and H. M. Wagner, "Optimal Capacity  
Scheduling - I and II," Operations Research, Vol. 10, No. 4  
(1962), pp. 518-46.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 31 <sup>v</sup>	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) CIRCULAR ONES AND CYCLIC STAFFING		5. TYPE OF REPORT & PERIOD COVERED Technical Report
7. AUTHOR(s) John J. Bartholdi, III, James B. Orlin, and H. Donald Ratliff		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Operations Research Stanford University Stanford, California 94305		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0493 <sup>v</sup>
11. CONTROLLING OFFICE NAME AND ADDRESS Logistics and Mathematical Statistics Branch Office of Naval Research Arlington, Virginia		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (NR-042-264)
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE November 20, 1977
		13. NUMBER OF PAGES 22
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) This document has been approved for public release and sale. Its distribution is unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) Also partially supported by National Science Foundation Grant ENG-76-12266, Army Research Office Grant DAHC04-75-G-0150, and Office of Naval Research Grant N00014-76-C-0096.		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Cyclic staffing Circular ones Integer programming Network flows Parametric programming		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  <div style="text-align: center; border: 1px solid black; border-radius: 50%; padding: 10px; width: fit-content; margin: 0 auto;">SEE REVERSE SIDE</div>		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE  
S/N 0102-014-6601


UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)


UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract



A large class of cyclic staffing problems, when formulated as (linear) integer programs, possess zero-one constraint matrices for which the ones in each row occur in consecutive components (the first and last components are considered consecutive). Included within this class is the problem of minimizing the linear cost of assigning workers to a multi-period cyclic schedule such that the demand in each period is satisfied, and each person works a shift of a common number of consecutive periods and is idle for the other periods (the first and last periods are considered consecutive). Any problem in this class may be transformed via a change of variables so that the resulting constraint matrix is, after deletion of a distinguished column, the transpose of a node-arc incidence matrix. The problem can then be solved in polynomial time parametrically in the distinguished variable as a sequence of network flow problems. Alternately, the optimal value of the distinguished variable can be found to within integer roundoff as its optimal value in the associated linear program with integer constraints ignored.



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)