

**TECHNICAL
RESEARCH
REPORT**

*Institute for
Systems
Research*

Reliability, Covering and Balanced Matrices

by M.O. Ball and F.L. Lin

*The Institute for Systems
Research is supported by the
National Science Foundation
Engineering Research Center
Program (NSFD CD 8803012),
Industry and the University*

TR 93-42

RELIABILITY, COVERING AND BALANCED MATRICES*

By

Michael O. Ball
University of North Carolina
and
University of Maryland at College Park

and

Feng L. Lin
University of Maryland at College Park

April, 1993

* The work was supported by National Science Foundation grant number NSFD CDR-8803012.

Abstract

This paper addresses a certain generalized covering integer program. The original application that motivated the study of this problem was an emergency services vehicle location problem. In this paper, we show that the same model can be applied to a general system reliability optimization problem. The main contribution of this paper is the definition and analysis of a reformulation strategy. Specifically, we show how the original generalized covering problem can be reformulated as a set covering problem. We then show that for a particular special case the associated constraint matrix is balanced. This in turn implies that the integer program can be efficiently solved using linear programming techniques. This result together with the good computational results reported in a previous paper constitute substantial evidence as to the overall effectiveness of the reformulation strategy. Furthermore, they indicate that the generalized covering model addressed can be effectively solved in a fairly wide range of cases.

1. Introduction

This paper addresses a class of generalized covering integer programs. While a set covering problem, $\text{Min } \{CX: AX \geq 1 \text{ for } 0 \leq X \leq 1 \text{ and integer}\}$, has a $(0,1)$ matrix, A , a generalized covering problem can have constraint coefficients and right hand side values that are any non-negative numbers. (See [11] for similar covering problems.) Although the problem we study has general applicability, we will show, in particular, that it can be used to represent certain reliability constrained integer programs. The main contribution of this paper is a reformulation strategy which converts the original generalized covering integer program (IP) to a set covering IP. This strategy makes use of concepts from blocking clutters. We show that under special conditions the constraint matrix is balanced, which in turn implies that the associated linear programming relaxation has all integer extreme points. We thus characterize a polynomially solvable case of the problem and also present theoretical evidence which explains, in part, the success of preprocessing techniques we used in [1] to solve integer programs of this type.

The specific generalized covering IP we consider can be described in terms of a bipartite graph, $G=(N_1, N_2, E)$, called the cover graph. Here, node set N_1 represents a set of customers, node set N_2 represents a set of potential service units (or stations) and edge set E describes customer/service unit compatibilities, by $E = \{(i,j): i \in N_1, j \in N_2 \text{ and unit } j \text{ can service customer } i\}$. A service unit j can be selected or not, and if it is selected, it must be assigned a capacity level k ranging from 1 to M_j . When service unit j is selected and assigned capacity level k , it generates a covering contribution a_{jk} and incurs a cost w_{jk} . A required covering level r_i is associated with each customer i . We denote by $\text{COV}(i) = \{j \in N_2: (i,j) \in E\}$, i.e., $\text{COV}(i)$ is the set of stations that can service customer i . The generalized covering IP we study is:

(CV-IP)

$$\text{Min} \quad \sum_{j \in J} \sum_{1 \leq k \leq M_j} w_{jk} x_{jk}$$

s.t.

$$\sum_{j \in \text{COV}(i)} \sum_{1 \leq k \leq M_j} a_{jk} x_{jk} \geq r_i \quad \text{for } i \in N_1, \quad (1)$$

$$\sum_{1 \leq k \leq M_j} x_{jk} \leq 1 \quad \text{for } j \in N_2, \quad (2)$$

$$x_{jk} \in \{0, 1\} \quad \text{for all } j, k.$$

Here, $x_{jk}=1$ implies that service unit j is chosen and assigned capacity level k and $x_{jk}=0$ implies that this option is not taken. We assume that a_{jk} , r_i , and w_{jk} are all positive real numbers. We can also assume without loss of generality that, for each j , $a_{jk} \leq a_{jh}$ and $w_{jk} \leq w_{jh}$ for $1 \leq k < h \leq M_j$.

A number of models closely related to this one have received considerable research attention. Of particular note are covering location problems defined on networks [8]. In this class of models both the customers and service units are defined as nodes on a network. The sets $\text{COV}(i)$ are defined as those service units within an prescribed network distance. Unlike most location models CV-IP does not have customer-to-service unit variables since we did not need wish to include costs that depend on the distances between customers and services units. On the other hand CV-IP has added complexity relative to the manner in which it handles multiple levels within a given service unit. Specifically, it does not assume that the marginal covering contributions, i.e. the $a_{jk+1} - a_{jk}$'s, are constant. This property is important in the applications we consider and led us to define multiple variables for a single service unit as opposed to having a single "cover level" variable for each service unit.

This model was used in [1] to address the problem of determining locations for

emergency service vehicles. In that context, stations corresponded to locations for emergency vehicles, customers to demand centroids, and the capacity level of a station to the number of vehicles housed at the station. Constraint (1) was obtained by taking logarithms of an expression which restricts an upper bound on the probability that an emergency service call arising at demand point i cannot be immediately serviced.

We now describe a model similar to the emergency services model, which addresses the problem of designing systems subject to reliability constraints. We refer the reader to reference [2] for background on reliability models of this type. We consider the general context of a coherent binary system, which is characterized by its component set, which we associate with N_2 , and some representation of system structure. Each component, j , has a failure probability q_j . There are many ways of representing the structure of the system. One, which in general is not the most efficient, is to produce a list of minimal cutsets. A cutset is a component subset, S , having the property: all elements in S fail implies the system fails. Thus, we have,

$$\Pr[\text{system fails}] = \Pr[\text{all elements in at least one minimal cutset fail}].$$

Now, let us consider a certain optimization model. With each component j , we associate a set of M_j "options". These options differ in terms of the associated component failure probabilities. Specifically, associated with each component j , we have a list of failure probabilities, q_{jk} for $k=1,2,\dots,M_j$ and corresponding costs, w_{jk} for $k=1,2,\dots,M_j$. A very general and important design problem would be to choose an option for each component so as to minimize total cost subject to a constraint on the system failure probability. While we do not have a method of addressing this very difficult problem directly, CV-IP can be used to solve the system design problem where a constraint on system failure probability is replaced with constraints on the

failure probability on individual minimal cutsets. This provides an approximate solution to the system design problem since upper bounds on the failure probability of minimal cutsets imply an upper bound on system failure probability. We derive the model as follows. First, we interpret $x_{jk}=1$ if option k is chosen for component j and $x_{jk}=0$ if it is not chosen. Let $C_i \subseteq N_2$ be any minimal cutset and β_i a desired limitation on the failure probability of C_i , then

$$\Pr[\text{all components in cutset } C_i \text{ fails}] = \prod_{j \in C_i} \prod_{1 \leq k \leq M_j} q_{jk} x_{jk} \text{ and}$$

$$\prod_{j \in C_i} \prod_{1 \leq k \leq M_j} q_{jk} x_{jk} \leq \beta_i \quad \text{if and only if (1) holds}$$

where $a_{jk} = -\log q_{jk}$ and $r_i = -\log \beta_i$. Thus, the problem of choosing a minimum cost set of options for the components of a system subject to constraints on the failure probabilities of certain minimal cutsets can be represented by CV-IP, where N_1 represents a set of minimal cutsets, N_2 represents the components and E indicates which components are contained in which cutsets.

There are a number of practical issues to be addressed in the application of this approach. Although obvious relationships exist between minimal cutset failure probability bounds and system reliability bounds, this topic has not received significant attention in the literature. Since in general the number of minimal cutsets grows exponentially with the number of components one would typically not include a constraint for all minimal cutsets (although one may wish to consider constraint generation approaches which did this implicitly). There are a number of strategies one might consider for choosing minimal cutsets and the right hand side values for the associated constraints. It seems quite clear, however, that since minimum cardinality cutsets have the greatest impact on the system reliability it would be most important to impose constraints on these. This consideration together with the observation that in most practical

settings of interest the size of a minimum cardinality cutset is two, indicates that instances of CV-IP in which a significant portion of the constraints of type (1) had $|\text{COV}(i)| = 2$ would be of practical interest.

In the remainder of this paper we present a reformulation strategy for CV-IP and derive some related properties. Section 2 treats the case where $|\text{COV}(i)| \leq 2$ for all constraints (1). It shows that under certain conditions the linear programming relaxation of the alternate formulation has all integer extreme points, which in turn implies that the associated IP can be solved efficiently using linear programming techniques. Section 3 extends the reformulation strategy to the general case.

We finish this section by characterizing the solvable case. Consider instances of CV-IP in which $|\text{COV}(i)| \leq 2$ for all constraints of type (1). As is illustrated by Figure 1, this is equivalent to assuming that all N_i nodes in the cover graph have degree at most two.

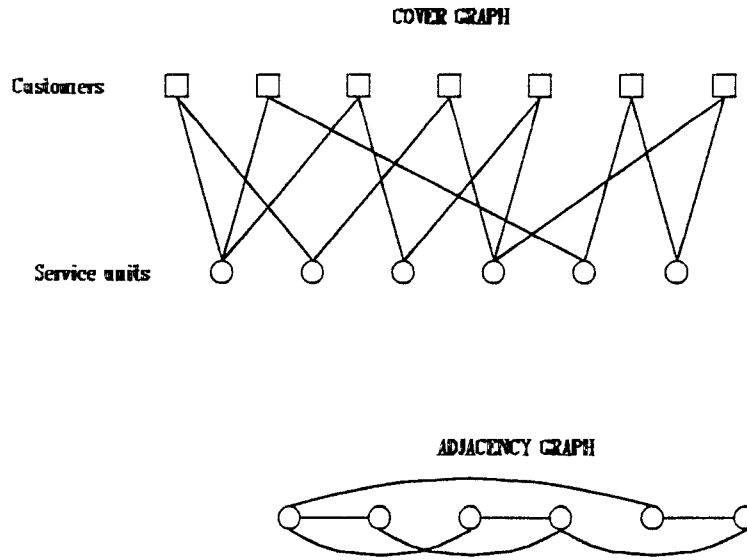


Figure 1: Cover graph and adjacency in 2-CV-IP case

We call this class of integer programs 2-Cover IPs (2-CV-IP). For any 2-CV-IP we can associate a graph, called the adjacency graph, which can be derived from the cover graph. The node set in this graph is N_2 , the set of service units/components. Two nodes, j_1 and j_2 , are adjacent in the adjacency graph if and only if there is a node from set N_1 in the cover graph adjacent to both j_1 and j_2 . Since all N_1 nodes in the cover graph have degree at most 2, the process of forming the adjacency graph is equivalent to making series reductions on the N_1 nodes and deleting pendant nodes. We can now characterize the solvable case.

Theorem 1. 2-CV-IP can be solved in polynomial time if the associated adjacency graph is bipartite.

This result follows from the analysis given in Section 2.

To put this result in perspective we note that 2-CV-IP can be defined in terms of the adjacency graph where the service units are nodes and the customers are edges. The problem of covering edges, multiple times, by nodes on a bipartite graph is the dual of a weighted b-matching problem and can be solved in polynomial time using matching theory or network flow theory. Our problem could be formulated in this way if $a_{jk+1} - a_{jk} = a^*$ all j and k . In this case the covering requirement for each edge would be $\lceil r_i/a^* \rceil$ where $\lceil d \rceil$ is the smallest integer greater than or equal to d . Thus, Theorem 1 applies to the more general setting when $a_{jk+1} - a_{jk}$ can vary in an arbitrary way. It is also interesting to note that reference [13] describes certain solvable cases of problems closely related to CV-IP based on the application of balanced matrices. Although we also use balanced matrices our paper and theirs address different classes of problems.

2. Substitution For 2-Cover Constraints

We refer the reader to [10] and [12] for background related to the results given in the remaining sections and to [9] for more details. It is well-known (see, for example, [4]) that if any $a_{jk} > r_i$ in constraint (1), we can replace a_{jk} by r_i , where the replacement will not affect the set of integer feasible solutions and will eliminate some fractional solutions. Hence, in this paper we always assume that $a_{jk} \leq r_i$ for each constraint in (1).

We now derive certain valid inequalities for 2-CV-IP, which, in fact, will lead to a completely new formulation. To start with, we describe a single 2-cover constraint as (3), and its two associated constraints in (2) as (4) and (5).

$$\sum_{j \in U} a_j x_j + \sum_{j \in V} b_j y_j \geq c \quad (3)$$

$$\sum_{j \in U} x_j \leq 1 \quad (4)$$

$$\sum_{j \in V} y_j \leq 1 \quad (5)$$

As noted earlier, both a_j 's and b_j 's are nondecreasing with j and $a_j, b_j \leq c$ for all j . We first observe that (3), (4) and (5) together with integrality implies:

For any $k \in U$, $x_k = 0$, for all $j > k$, implies $y_j = 1$ for some j with $b_j + a_k \geq c$.

The implication can be expressed as:

$$\sum_{b_j + a_k \geq c} y_j \geq 1 - \sum_{j > k} x_j. \quad (*)$$

The above constraint remains valid even if $x_j = 1$ for some $j > k$. We now let $U_c = \{j \in U: a_j = c\}$ and $V_c = \{j \in V: b_j = c\}$. Rearranging (*), we thus obtain the following valid inequalities.

$$\sum_{\{j \in U: j > k\}} x_j + \sum_{\{j \in V: b_j + a_k \geq c\}} y_j \geq 1 \quad k \in U \setminus U_c \quad (6)$$

$$\sum_{\{j \in V: j > k\}} y_j + \sum_{\{j \in U: a_j + b_k \geq c\}} x_j \geq 1 \quad k \in V \setminus V_c \quad (7)$$

We thus have Proposition 1.

Proposition 1. (6) and (7) are valid inequalities for the convex hull defined by (3), (4) and (5) together with variable integrality.

Define R_+ to be the set of non-negative real numbers and $B_+ = \{0, 1\}$. Consider the two polyhedra P_1 and P_2 as below:

$$P_1 = \{x \in R_+^{|U|+|V|}: (3) - (5)\}$$

$$P_2 = \{x \in R_+^{|U|+|V|}: (4) - (7)\}$$

Proposition 2 shows that P_2 can be used to replace P_1 in an integer programming formulation. Proposition 3 shows that this substitution is worthwhile since P_2 only has integer extreme points.

Proposition 2. $P_1 \cap \{X \in B_+^{|U|+|V|}\} = P_2 \cap \{X \in B_+^{|U|+|V|}\}.$

Proof: The binary solutions contained in P_1 have at most one x_j equal to 1 and at most one y_j equal to 1. Furthermore, for the x_j and y_j set to 1, it must be that $a_j + b_j \geq c$. This is exactly the set of conditions insured by (4) - (7). ||

Proposition 3. All extreme points of P_2 are integral.

Proof: We prove the result by showing that P_2 can be described as $\{X \in R_+^n: AX \geq b\}$ where A is a totally unimodular (TU) matrix. Consider a constraint in (7) for $k \in V \setminus V_c$. Since the a_j 's and b_j 's are ordered by increasing value of a and b respectively, the sets $\{j \in U: a_j + b_k \geq c\}$ and $\{j \in V: j > k\}$ both consist of consecutive j 's. Thus, if the columns are arranged in the order of $(x_1, \dots, x_{|U|-1}, x_{|U|}, y_{|V|}, y_{|V|-1}, \dots, y_1)$, the 1's in each constraint will appear in consecutive columns. The same holds for a constraint in (6). Clearly, constraints (4) and (5) also have the consecutive 1's property. Thus, this constraint matrix is TU (see [10] pg 544). ||

Example 1. Consider an example of (3) - (5):

$$0.6x_1 + 1.6x_2 + 3.0x_3 + 3.5x_4 + 0.4y_1 + 1.2y_2 + 2.3y_3 + 3.5y_4 \geq 3.5$$

$$x_1 + x_2 + x_3 + x_4 \leq 1$$

$$y_1 + y_2 + y_3 + y_4 \leq 1$$

Thus, the constraints in (4) - (7) are displayed as follows:

$$\begin{array}{cccccccc}
 & & & x_4 + & & y_2 + & y_3 + & y_4 & \geq & 1 \\
 & & & x_3 + & x_4 + & & y_3 + & y_4 & \geq & 1 \\
 x_2 + & x_3 + & x_4 + & & & & & y_4 & \geq & 1 \\
 x_2 + & x_3 + & x_4 + & & & & & y_4 & \geq & 1 \\
 & & x_3 + & x_4 + & & & y_3 + & y_4 & \geq & 1 \\
 & & & x_4 + & & y_2 + & y_3 + & y_4 & \geq & 1 \\
 x_1 & x_2 & x_3 & x_4 & & & & & \leq & 1 \\
 & & & & y_1 & y_2 & y_3 & y_4 & \leq & 1
 \end{array}$$

After reversing the order of the columns associated with the y variables, the constraint matrix becomes:

$$\begin{array}{cccccccc}
 x_1 & x_2 & x_3 & x_4 & y_4 & y_3 & y_2 & y_1 \\
 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\
 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\
 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1
 \end{array}$$

We now wish to address what occurs when all instances of (1) are replaced by inequalities of the form given by (6) and (7). Two questions now arise. First, what properties does the constraint matrix formed by replacing each instance of (1) by (6) and (7) have, i.e. what is the effect of intersecting several instances of (6) and (7). Second, can the overall formulation be modified to form a "pure" set covering problem (this is currently not the case

due to the presence of (2)).

We first note that the answer to the second question is yes. Specifically, (6) and (7) alone are sufficient to replace a constraint system (3) - (5); namely, (4) and (5) becomes superfluous. This is true since any optimal solution will satisfy: $\sum_{1 \leq k \leq M_j} x_{jk} \leq 1$ for all $j \in N_2$. Suppose that X satisfies (6) and (7) with $\sum_{1 \leq k \leq M_j} x_{jk} \geq 2$ for some $j \in N_2$. Let x be such that $x_{jk1} = x_{jk2} = 1$ with $1 \leq k_1 < k_2 \leq M_j$ for j . It is obvious that the new solution with $x_{jk1} = 0$ but $x_{jk2} = 1$ still satisfies (6) and (7) yet has a smaller weight since all $w_{jk} > 0$. Hence, we can reformulate CV-IP by dropping (2) and replacing each instance of (1) with (6) and (7). The result becomes a set covering formulation. That is, the constraint set has the form $AX \geq 1$ where A is a 0/1 matrix. Note that it is not possible to drop (2) from the original formulation without imposing stronger conditions on the a_{jk} and/or w_{jk} .

We now consider the first question. Specifically, we will derive conditions under which the new formulation will have all integer extreme points. For any 0/1 matrix B , we say that B has the Row Strong Consecutive Ones (RSCO) property, if for all rows i , $b_{ij} = 1$ implies $b_{ik} = 1$ for all $k \geq j$. In replacing a constraint (3), we see that blocks associated with (6) or (7) have the RSCO property.

We now review certain concepts related to balanced matrices which are fundamental to our results. It is well known that a graph is bipartite if and only if it has no odd cycles. Any undirected graph can be represented by an edge-node incidence matrix $D = [d_{ij}]$ where $d_{ij} = 1$ if edge i is incident to node j and $d_{ij} = 0$ if otherwise. The edges of any odd cycle on $k \geq 3$ nodes can be ordered so that the edge-node incidence matrix has the following form.

$$\begin{array}{ccccccc}
1 & 1 & 0 & 0 & . & 0 & 0 \\
0 & 1 & 1 & 0 & . & 0 & 0 \\
0 & 0 & 1 & 1 & . & 0 & 0 \\
& . & . & . & . & 1 & 0 \\
& . & . & . & . & 1 & 1 \\
1 & 0 & . & . & . & 0 & 1
\end{array}$$

For $k \geq 3$ odd, we call any square $k \times k$ 0/1 matrix an odd-cycle matrix if it has the form given above after suitable row and column permutations. A 0/1 matrix is defined to be balanced if it does not contain an odd-cycle submatrix. Balanced matrices are important because if A is balanced then $\{x \in \mathbb{R}_+ : Ax \leq 1\}$ and $\{x \in \mathbb{R}_+ : Ax \geq 1\}$ are both integral [3], [7].

Prior to proving Theorem 1 we state and prove another theorem which characterizes a certain class of balanced matrices. First we define some notation. For an $n \times m$ matrix A and ordered pairs, (j_1, j_2) with $0 \leq j_1 < j_2 \leq n$ and (i_1, i_2) with $0 \leq i_1 < i_2 \leq m$, denote by $A[(i_1, i_2), (j_1, j_2)]$ the submatrix restricted to columns $j_1 + 1$ through j_2 and rows $i_1 + 1$ through i_2 .

Theorem 2. Let A be a 0/1 matrix whose rows and columns can be permuted so that there exist j_1 through j_r and i_1 through i_t with $0 = j_1 < j_2 < \dots < j_r = n$ and $0 = i_1 < i_2 < \dots < i_t = m$ such that,

- 1.) $A[(i_h, i_{h+1}), (j_k, j_{k+1})]$ has the RSCO property for all k and h ,
- 2.) for any h , $A[(i_h, i_{h+1}), (j_k, j_{k+1})]$ is non-zero for at most two different values of k ,
- 3.) the undirected graph whose node set is $\{1, 2, \dots, J\}$ and whose edge set is $\{(k_1, k_2) : \text{for some } h, \text{ both } A[(i_h, i_{h+1}), (j_{k_1}, j_{k_1+1})] \text{ and } A[(i_h, i_{h+1}), (j_{k_2}, j_{k_2+1})] \text{ are non-zero}\}$ is bipartite.

Then A is balanced.

Proof: We call each $A[(i_h, i_{h+1}), (j_k, j_{k+1})]$ a block of A and each block that contains at least one

non-zero elements a non-zero block. The theorem will be proved by showing that any square submatrix, B whose row and column sums are at least 2 cannot be an odd-cycle matrix. First, note that by condition 3.) if B has at most one 1 from each non-zero block then it cannot be an odd-cycle matrix since the existence of such a submatrix would imply the existence of an odd-cycle in the bipartite graph. Now, suppose that B contains columns j' and j'' and row i with $i_h < i \leq i_{h+1}$ and $j_h < j' < j'' \leq j_{h+1}$ with $a_{ij'} = a_{ij''} = 1$. B must contain another row i^* with $a_{i^*j'} = 1$. But, by the RSCO property, $a_{i^*j''} = 1$ which means the rows i, i^* and columns j', j'' form a square matrix of 1's implying B is not odd-cycle matrix. Alternatively, suppose that B contains rows i' and i'' and column j with $i_h < i' < i'' \leq i_{h+1}$ and $j_h < j \leq j_{h+1}$ with $a_{i'j} = a_{i''j} = 1$. There must exist columns j' and j'' with $a_{i'j'} = a_{i''j''} = 1$. By condition 2.), either both of these elements are in the same non-zero block or one of them is in the block containing $a_{i'j}$ and $a_{i''j}$. Either situation implies the submatrix has two non-zero column elements intersecting the same block, which has just been shown to imply that B contains a 2×2 submatrix of 1's. The proof is now complete.

||

The RSCO property implies both the row consecutive ones property and the row inclusion property (see [10] for definitions). However, Theorem 2 does not hold if either of these properties are substituted for the RSCO property.

Theorem 1 now follows as a direct consequence of Theorem 2.

Proof of Theorem 1:

Each instance of constraint (1) is replaced by two sets of constraints -- (6) and (7). As was stated earlier once this is done constraint set (2) can be deleted. Since each constraint matrix block associated with (6) or (7) has the RSCO property and since by assumption the adjacency

graph is bipartite, we are left with a set covering problem whose constraint matrix satisfies the conditions of Theorem 2. By Theorem 2 all extreme points are integer and we may solve the integer program by finding an extreme point solution to the linear program which can be accomplished in polynomial time. ||

4. Substitution For Multi-Cover Constraints

This section generalizes the concepts used in Section 2 so that they can be applied to generate constraint substitutions for multi-cover, i.e., $|\text{COV}(i)| \geq 3$, constraints in (1). Our analysis makes use of concepts from blocking clutters [5]. We note that we employ concepts similar to those used in the development of blocking and anti-blocking polyhedra [6]. Given a finite set N we define a clutter Ψ to be a family of subsets of N such that no member of Ψ is contained in any other member of Ψ . A blocking clutter (or blocker) of Ψ is a clutter denoted as $B(\Psi)$ whose members H satisfy: (i) Intersection: $H \cap F \neq \emptyset$ for all $F \in \Psi$; (ii) Minimality: If H' is strictly contained in H , then $H' \cap F = \emptyset$ for some $F \in \Psi$.

We start with considering an individual $|\text{COV}(i)| \geq 3$ constraint and its associated constraints in (2). A set $CP(i)$ for each $i \in I$ is defined as follows:

$$CP(i) = \{X \in B^n: \sum_{j \in \text{COV}(i)} \sum_{1 \leq k \leq M_j} a_{jk} x_{jk} \geq r_i \text{ and } \sum_{1 \leq k \leq M_j} x_{jk} \leq 1 \text{ for } j \in \text{COV}(i)\}.$$

Using ground set $N(i) = \{(j,k): \text{for all } j \in \text{COV}(i), 1 \leq k \leq M_j\}$ we define a clutter Ψ_i , by $F \in \Psi_i$ if F is a minimal set satisfying $X^F \in CP(i)$ where X^F represents the incidence vector of F . In order to describe the blocking clutter $B(\Psi_i)$, we define "cutsets" for $CP(i)$. These cutsets are conceptually similar to the cutsets defined in reliability literature in that if all elements of a cutset are removed or "fail to contribute", then $CP(i) = \emptyset$. A set H of indices $\{(j,k)\}$ is called a cutset relative to $CP(i)$ if it satisfies $\sum_{j \in \text{COV}(i)} \text{Max}_k \{a_{jk}: (j,k) \notin H\} < r_i$.

The following example illustrates the process of generating a minimal cutset.

Example 2. Consider the following constraint with $|\text{COV}(i)| = 3$.

$$2x_{11} + 3x_{12} + 5x_{13} + 7x_{14} + 3x_{21} + 4x_{22} + 8x_{23} + 9x_{24} + x_{31} + 5x_{32} + 7x_{33} + 10x_{34} \geq 18$$

Thus, $\text{CP}(i) = \{(1,1), (1,2), (1,3), (1,4), (2,1), (2,2), (2,3), (2,4), (3,1), (3,2), (3,3), (3,4)\}$

The process of generating a minimal cutset, H , can be illustrated as follows:

service unit 1:	2	3	5	7
service unit 2:	3	4	8	9
service unit 3:	1	5	7	10

We start by choosing $\text{Max}_k \{a_{jk} : (j,k) \notin H\}$. That is one (j,k) is chosen for each j , such that $\Sigma_{(j,k) \text{ chosen}} a_{jk} < b_i$. To insure that a minimal cutset is generated we require that if $a_{j,k+1}$ is substituted for any a_{jk} then this property does not hold. In this case the chosen set is $\{(1,2), (2,3), (3,2)\}$ and the corresponding a_{jk} values, illustrated in bold, are 3, 8 and 5. Note that $3+8+5 = 16 < 18$ but if $a_{j,k+1}$ is substituted for any a_{jk} then the sum will be greater than or equal to 18. The associated minimal cutset, H , is $\{(1,3), (1,4), (2,4), (3,3), (3,4)\}$.

We are now ready to establish in Proposition 4 that these cutsets constitute the blocking clutter of Ψ_i . We let $X^F \in B^n$ and $Y^H \in B^n$ represent the incident vectors to an F in Ψ_i and an H in $B(\Psi_i)$, respectively.

Proposition 4. $\{H: H \text{ is a minimal cutset for } \text{CP}(i)\} = B(\Psi_i)$.

Proof: We only need to show the intersection property holds for $B(\Psi_i)$, since the minimality condition holds for the minimal cutsets. In particular, we wish to show that a set H satisfies

the intersection property holds.

Let H be such that $\sum_{j \in \text{COV}(i)} \text{Max}_k \{a_{jk} : a_{jk} \notin H\} < r_i$. Suppose that $Y^H X^F = 0$ for some $F \in \Psi_i$. This means that the feasible solution X^F has all of its 1's incident to the $a_{jk} \notin H$. Thus, $r_i \leq \sum_{(j,k) \in F} a_{jk}$ since $X^F \in \text{CP}(i)$. As X^F takes at most one a_{jk} for each $j \in \text{COV}(i)$, we have $\sum_{(j,k) \in F} a_{jk} \leq \sum_{j \in \text{COV}(i)} \text{Max}_k \{a_{jk} : a_{jk} \notin H\}$. As a result, $r_i \leq \sum_{j \in \text{COV}(i)} \text{Max}_k \{a_{jk} : a_{jk} \notin H\}$, which is a contradiction. This shows that a minimal cutset satisfies the intersection property.

On the other hand, let H be such that $Y^H X^F \geq 1$ for all $F \in \Psi_i$. Suppose that $\sum_{j \in \text{COV}(i)} \text{Max}_k \{a_{jk} : a_{jk} \notin H\} \geq r_i$. That is, there exists some $F \in \Psi_i$ such that $H^c \supseteq F$. Hence, $Y^H X^F = 0$ for some $F \in \Psi_i$, which is a contradiction. ||

Clutters and their blockers are useful in deriving integer programming formulations since $Q^1 = \{X \in Z_+^n : Y^H X \geq 1 \text{ for all } H \in B(\Psi)\}$ and $Q^2 = \{X \in Z_+^n : X \geq X^F \text{ for some } F \in \Psi\}$ are identical. The extreme points of the convex hull of Q^2 ($\text{conv}(Q^2)$), are precisely the X^F 's. With $w > 0$, an X^F will be the optimal solution to $\text{Min } \{wX : X \in \text{conv}(Q^2)\}$, hence, the optimal solution to $\text{Min } \{wX : X \in Q^2 \cap B^n\}$, and, thus, the optimal solution to $\text{Min } \{wX : X \in Q^1 \cap B^n\}$. Thus, the inequalities $Y^H X \geq 1$ for all $H \in B(\Psi_i)$ can be used to replace constraint in (1) for $i \in I$. This result coincides with what we concluded for a $|\text{COV}(i)|=2$ constraint with the exception that the total unimodularity property is not necessarily satisfied here for a single replacement and the overall matrix is not necessarily balanced. Again, the constraints $\sum_{1 \leq k \leq M_j} x_{jk} \leq 1$ for all $j \in N_2$ can be dropped.

In general, the combinatorial optimization problem, $\text{Min } \{w(F) : F \in \Psi\}$, may be formulated as a set covering problem once the blocking clutter, $B(\Psi)$ is identified. The two major impediments to using this strategy are that $B(\Psi)$ may be difficult to identify or $B(\Psi)$

could be very large. However, the results from [1] indicate that if $|\text{COV}(i)|$ is relatively small, e.g. less than 6, then $B(\Psi_i)$ can be determined within a reasonable amount of computing time. We can then put the blockers $B(\Psi_i)$ for all $i \in N_1$ together and obtain a set covering formulation for CV-IP. This process "works" since $\cup_{i \in I} B(\Psi_i) \supseteq B(\Psi)$. That is, $\sum_{j,k:(j,k) \in H} x_{jk} \geq 1$ for all $H \in \cup_{i \in I} B(\Psi_i)$ can be used to replace (1) and (2). Of course, when taking the union minimality cannot be guaranteed so that redundant constraints could be generated. This would not affect the validity of the formulation. Redundant constraints could certainly be eliminated by preprocessing if desired.

As a result, if we can identify all members of $B(\Psi_i)$ for all $i \in N_1$, the constraint substitution reduces CV-IP into a set covering IP:

$$\begin{aligned}
 \text{Min} \quad & \sum_{j \in J} \sum_{1 \leq k \leq M_j} w_{jk} x_{jk} \\
 \text{s.t.} \quad & \\
 & \sum_{j,k:(j,k) \in H} x_{jk} \geq 1 \quad \text{for all } H \in B(\Psi_i), \text{ for all } i \in N_1 \\
 & x_{jk} \in \{0,1\} \quad \text{for all } j, k
 \end{aligned}$$

While the set covering IP may remain difficult to solve, this paper shows that under certain special conditions the reformulation strategy leads to a polynomially solvable case. Furthermore, the computational results in [1], indicate that this approach has general merit as long as $|\text{COV}(i)|$ is not "too large".

References

- [1.] M. Ball and F. Lin (1993). "A Reliability Model Applied To Emergency Service Vehicle Location," Operations Research, **41**, 1-15.
- [2.] M. Ball, J. Provan, and D. Shier (1991). "Reliability Covering Problems," Networks, **21**, 345-357.
- [3.] C. Berge (1972), "Balanced Matrices", Mathematical Programming, **2**, 19-31.
- [4.] H. Crowder, E. Johnson, and M. Padberg (1983). "Solving Large-Scale Zero-One Linear Programming Problems," Operations Research, **31**, 803-834.
- [5.] J. Edmonds and D. Fulkerson (1970). "Bottleneck Extrema," in Studies In Optimization, G. Dantzig and B. Eaves, Eds., The Mathematical Association Of America, 94-103.
- [6.] D. Fulkerson (1971). "Blocking And Antiblocking Pairs Of Polyhedra," Mathematical Programming, **1**, 168-194.
- [7.] D. Fulkerson, A. Hoffman and R. Oppenheim (1974), "On Balanced Matrices", Mathematical Programming Study, **1**, 120-132.
- [8.] A. Kolen and A. Tamir (1990). "Covering Problems", in Discrete Location Theory, P. Mirchandani and R. Francis, Eds., John Wiley & Sons, New York, pp 263-304.
- [9.] F. Lin (1992). "Reliability Constrained Integer Programming Models With Applications to Facility Location", Ph.D. Thesis, University of Maryland.
- [10.] G. Nemhauser and L. Wolsey (1988). Integer and Combinatorial Optimization, John Wiley & Sons.
- [11.] J. Orlin (1982). "A Polynomial Algorithm For Integer Programming Covering Problems Satisfying The Integer Round-Up Property," Mathematical Programming, **22**, 231-235.
- [12.] M. Padberg (1979). "Covering, Packing And Knapsack Problems," Annals Of Discrete Mathematics, **4**, 265-287.
- [13.] A. Tamir (1983). "A Class of Balanced Matrices Arising from Location Problems", SIAM Journal on Algebraic and Discrete Methods, **4**, 363-370.