ESTIMATES AND BOUNDS ON COMPUTATIONAL EFFORT IN THE ACCELERATED BOUND-AND-SCAN ALGORITHM

by

Bruce H. Faaland

TECHNICAL REPORT NO. 35

May 15, 1972

PREPARED UNDER CONTRACT

N00014-67-A-0112-0058 (NR-047-061)

FOR THE OFFICE OF NAVAL RESEARCH

Frederick S. Hillier, Project Director

This research was supported in part by

National Science Foundation Grant GJ-31521.

DEPARTMENT OF OPERATIONS RESEARCH

STANFORD UNIVERSITY

STANFORD, CALIFORNIA

Estimates and Bounds on Computational Effort in the Accelerated Bound-and-Scan Algorithm

by

Bruce H. Faaland

A. Introduction

The pure integer linear programming (IP) problem is

$$\text{maximize} \quad c^T x \tag{1}$$

$$\text{subject to} \quad Ax \leq b \tag{2}$$

$$x \geq 0 \tag{3}$$

$$x \quad \text{integer}, \tag{4}$$

where A is an $m \times n$ matrix, b is an m-vector, and c is an n-vector. A difficulty shared by the many algorithms which have been proposed for solving the IP problem is that their computation time varies greatly from problem to problem of similar size and structure. The purpose of this paper is to give upper bounds and estimates on the number of arithmetic operations needed by the Accelerated Bound-and-Scan Algorithm to solve any given problem.

The Accelerated Bound-and-Scan Algorithm solves the integer programming problem (1,2,3,4) by implicit enumeration of all lattice points within a certain n-simplex in Euclidean n-space. The number of arithmetic operations required by the algorithm will be given as a function of the number of eligible partial solutions through each variable. Methods for obtaining upper bounds and estimates on the number of eligible partial solutions, for any given problem, will be discussed in detail. The basic tools of analysis used are various results from the theory of partitions of numbers, the Central-limit theorem, and geometrical interpretations of the algorithm. Many of the results described here also apply to the Bradley-Wahi algorithm [3]. See [9] for details on the relationship of the Bradley-Wahi algorithm to the Accelerated Bound-and-Scan Algorithm.

B. Analysis of Arithmetic Operations

A somewhat simplified version of the skeleton algorithm will be analyzed here. The "simplified algorithm" includes the basic enumeration scheme and bounds $\bar{\rho}_i$ on variables ρ_i used in the skeleton algorithm, but it does not include the scanning procedure or the method for eliminating redundant nonbinding constraints. Instead, whenever a completion within the n-simplex is generated, the m nonbinding constraints are checked in sequence for feasibility. If all m nonbinding constraints are satisfied, an improved solution has been found. Otherwise, the basic enumeration scheme continues where it left off as soon as it has been determined that some nonbinding constraint is violated by the completion.

The simplified algorithm is considered instead of a more sophisticated (and probably more efficient) version of the algorithm in order to make the analysis tractable. The simplified algorithm includes the cone algorithm as a special case, and the results of the analysis for the simplified algorithm provide partial answers to fundamental questions which arise regardless of the level of sophistication in the design of the algorithm.

These questions include (a) What is the greatest amount of time the algorithm could take to solve a given problem? (b) What is the expected computation time? (c) How should the variables and nonbinding constraints be ordered to minimize the expected computation time? (d) How does the computation time increase as the number of variables is increased? and (e) How is the computation time affected by the quality of the starting solution?

In the following it will be necessary to refer to equation numbers in [9]. Equation number q in [9] will be designated by (9.q), whereas equation number q in this chapter will be labeled (q). It should also be noted at this point that only the "iterative portion" of the algorithm is being considered; the relatively predictable and minor "setup time" is being ignored.

The basic idea of the Accelerated Bound-and-Scan Algorithm is to examine all completions which may yield a larger objective function value than the current best solution. To arrive at a completion, a sequence of partial solutions is constructed. The computational effort required by the algorithm depends on the number of partial solutions and on the amount and type of computation required to generate each partial solution.

Partial solutions are generated on either a "forward step" or a "backtrack step" in the algorithm. Each will be examined in terms of the computational effort it requires.

Suppose $(\rho_1^*,\cdots,\rho_{k-1}^*)$ is an eligible partial solution, but not a completion. The forward step consists of setting ρ_k at t in (9.26) to form the next partial solution. Consider the work required to find t. If $k \leq n_1$, $f = 0$ by (9.30) and $t = 0$ by (9.26); thus, if $k \leq n_1$, $(\rho_1^*,\cdots,\rho_{k-1}^*,0)$ is automatically an eligible partial solution. No multiplications or additions are required. If $k > n_1$, the calculation of t first requires $(k-1)$ multiplications and k additions to obtain f, and then one additional multiplication and addition in (9.26) to obtain t. (Assume that prior to the iterative part of the algorithm the $1/B_{kk}$, $k = 1,\cdots,n$, have been calculated and stored.) To test whether $(\rho_1^*,\cdots,\rho_{k-1}^*,t)$ is an eligible partial solution, t must be added to the running sum $\sum_{i=1}^{k-1} \rho_i^*$ in (9.24), resulting in one more addition. Therefore, when $k > n_1$, $(k+2)$ additions and k multiplications are needed on the forward step, but if $k \leq n_1$, none are required. If $(\rho_1^*,\ldots,\rho_{k-1}^*)$ is a completion $(k-1 = n)$, the nonbinding constraints must be checked for feasibility, requiring at most mn multiplications and additions, where m is the number of rows of A in (2).

It is now possible to give the following lemma.

Lemma 1 Let A_F be the number of additions and M_F the number of multiplications done during the forward steps of the simplified Accelerated Bound-and-Scan Algorithm. If E_j is the number of eligible partial solutions through variable j,

$$A_F \leq \sum_{j=n_1}^{n-1} (j+3)E_j + mnE_n \text{ ,}$$

and

$$M_F \leq \sum_{j=n_1}^{n-1} (j+1)E_j + mnE_n \text{ .}$$

Proof The proof has largely been given above. Every time one of the E_j eligible partial solutions $(\rho_1^*,\cdots,\rho_j^*)$, $n_1 \leq j < n$, is generated, $(j+3)$ additions and $(j+1)$ multiplications are required. If $j = n$, at most mn of each are needed.

The "backtrack steps" of the algorithm will now be analyzed. Suppose again that $(\rho_1^*,\cdots,\rho_{k-1}^*)$ is an eligible partial solution and that ρ_k is set at t during the forward step of the algorithm. If $(\rho_1^*,\cdots,\rho_{k-1}^*,t)$ is an eligible partial solution, the algorithm proceeds with an additional forward step. If it is not an eligible partial solution, the algorithm attempts to construct a new eligible partial solution up to variable k-2 by resetting ρ_{k-2} at $\rho_{k-2}^*+1/B_{k-2,k-2}$. This operation initiates the "backtrack step" and requires a single addition. (Recall that the $1/B_{ii}$ are stored, $i = 1,\cdots,n$.) To check whether $(\rho_1^*,\cdots,\rho_{k-3}^*,\rho_{k-2})$ is an eligible partial solution, $1/B_{k-2,k-2}+(-\rho^*_{k-1})$ must be added to $\sum_{i=1}^{k-1}\rho^*_i$. That requires two more additions, or three in all during a backtrack step. If the partial solution is not eligible, another backtrack step to ρ_{k-3} is made, requiring another three additions. If it is eligible, a forward step is then made. There is one exception to this description. Once a completion $(\rho^*_1,\cdots,\rho^*_n)$ has been found and tested against the non-binding constraints, the algorithm "backtracks" to ρ_n by resetting ρ_n at ρ^*_n+1/B_{nn}. Since adjusting $\sum_{i=1}^{n}\rho^*_i$ requires only one addition, backtracking to ρ_n requires only two additions.

<u>Lemma 2</u> Let A_B be the number of additions and M_B the number of multiplications done on the backtrack steps of the Accelerated Bound-and-Scan Algorithm. Then

$$A_B \leq 3 \sum_{j=1}^{n-1} E_j + 2E_n$$

and

$$M_B = 0.$$

<u>Proof</u> The algorithm can backtrack to variable j, i.e., reset ρ_j at ρ^*_j+1/B_{jj}, if and only if an eligible partial solution $(\rho^*_1,\ldots,\rho^*_j)$ has previously been constructed. Therefore the number of times the algorithm backtracks to variable j is at most E_j, and three additions are needed for each such occasion

The following fundamental theorem follows immediately from Lemmas 1 and 2.

Theorem 1 If A_T is the number of additions and M_T the number of multiplications required during the iterative part of the simplified Accelerated Bound-and-Scan Algorithm, then

$$A_T \leq \sum_{j=n_1}^{n-1} (j+3)E_j + mnE_n + 3\sum_{j=1}^{n-1} E_j + 2E_n \ , \tag{5}$$

and

$$M_T \leq \sum_{j=n_1}^{n-1} (j+1)E_j + mnE_n . \tag{6}$$

The only arithmetic operations done during the iterative part of the algorithm are additions and multiplications. No divisions are necessary, and subtractions have been assumed to be equivalent to additions in computation time. Comparison and assignment operations have been ignored, but they could easily have been counted. The importance of Theorem 1 is that if the time to do an addition and the time for a multiplication are known, and if upper bounds (or estimates) on the E_j are known, it is possible to give an upper bound (or an estimate) on the time required by the Accelerated Bound-and-Scan Algorithm to solve any particular case of problem (1,2,3,4). Upper bounds on the E_j will now be developed.

C. Some Upper Bounds

Lemma 3 The number of eligible partial solutions through variable ρ_j, defined as the number of solutions to (9.21), is bounded above by the number of distinct solutions to the system

$$\begin{pmatrix} B_{11} & & 0 \\ & \ddots & \\ 0 & & B_{jj} \end{pmatrix} \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_j \end{pmatrix} \equiv 0 \pmod 1 \tag{7}$$

$$\sum_{i=1}^{j} \rho_i \leq 1$$

$$0 \leq \rho_i \leq \bar{\rho}_i \qquad i = 1,\cdots,j \ .$$

Proof The lemma will be proved by defining - one-to-one mapping from the set of solutions to (9.21) into the set of solutions to (7). Consider any solution $(\rho_1^*,\cdots,\rho_j^*)$ to (9.21). By (9.27) each ρ_i^* is of the form

$$\rho_i^* = t_i^* + y_i^*/B_{ii} \qquad i = 1,\cdots,j,$$

where y_i^* is a non-negative integer, and $0\le t_i^*<1/B_{ii}$. The solution $(\rho_1^*,\cdots,\rho_j^*)$ also satsifies

$$0 \le \rho_i^* \le \bar{\rho}_i \qquad i = 1,\cdots,j$$

and

$$\sum_{i=1}^{j} \rho_i^* \le 1.$$

Corresponding to every solution $(\rho_1^*,\cdots,\rho_j^*)$ to (9.21) is a solution $(\rho_1,\cdots,\rho_j)$ to (7) obtained by letting

$$\rho_i = \rho_i^* - t_i^* = y_i^*/B_{ii} \qquad i = 1,\cdots,j.$$

To see that $(\rho_1,\cdots,\rho_j)$ is feasible for (7), notice that $B_{ii}\rho_i = y_i^*$, an integer; $\rho_i\ge 0$ since $y_i^*\ge 0$ and $B_{ii}>0$; since $\rho_i\le\rho_i^*$, $\rho_i\le\bar{\rho}_i$ and $\sum_{i=1}^{j}\rho_i\le 1$.

It still remains to be proved that the mapping is one-to-one. Suppose it is not. Then two different solutions $\rho^* = (\rho_1^*,\ldots,\rho_j^*)$ and $\rho^{**} = (\rho_1^{**},\cdots,\rho_j^{**})$ to (9.21) must correspond to a single solution to (7). Let k denote the first index for which $\rho_i^* \ne \rho_i^{**}$. Recall that $\rho_k^* = t_k^*+y_k^*/B_{kk}$ and $\rho_k^{**} = t_k^{**}+y_k^{**}/B_{kk}$. If $\rho_k^* \ne \rho_k^{**}$, then either $t_k^* \ne t_k^{**}$ or $y_k^* \ne y_k^{**}$. But y_k^* must equal y_k^{**}, for otherwise ρ^* and ρ^{**} map into different solutions to (7). And t_k^* must equal t_k^{**} because t_k^* and t_k^{**} are functions only of the first $(k-1)$ components of ρ^* and ρ^{**} respectively. Therefore, there are at least as many solutions to (7) as there are to (9.21), as was to be shown.

<u>Lemma 4</u> The number of solutions to (7) is equal to the number of integer solutions to the inequality

$$\sum_{i=1}^{j} y_i/B_{ii} \le 1 \tag{8}$$

$$0 \le y_i \le \bar{y}_i \qquad i = 1,\cdots,j,$$

where $\bar{y}_i = \lfloor \bar{\rho}_i B_{ii} \rfloor$ for $i = 1,\cdots,j$.

<u>Proof</u> The condition $B_{ii}\rho_i \equiv 0 \pmod 1$ simply says that $B_{ii}\rho_i$ must be an integer, say y_i. Then $\rho_i = y_i/B_{ii}$ substituted into the requirements $0 \le \rho_i \le \bar{\rho}_i$ $(i = 1,\cdots,j)$ and $\sum_{i=1}^{j} \rho_i \le 1$ results in (8)

If the B_{ii} are rational, it is possible to calculate precisely the number of solutions to (8) by solving a related problem in the theory of partitions of numbers. The theory of partitions [11] deals with the problem of counting the number of nonnegative integer solutions to equations of the form

$$\sum_{i=1}^{r} a_i y_i = a,$$

where $a_1,\cdots,a_r$ and a are known positive integers. The problem may be posed as that of counting the number of ways the integer a may be divided into parts of sizes $a_1,\cdots,a_r$. If only $\bar{y}_i$ parts of size a_j are available, the problem becomes one of finding the number of integer solutions to the bounded variable Diophantine equation

$$\sum_{i=1}^{r} a_i y_i = a \tag{9}$$

$$0 \le y_i \le \bar{y}_i \qquad i = 1,\ldots,r.$$

Assume the B_{ij} in (8) are rational. A sufficient condition for this to be true is that the coefficients of the original IP problem be rational. Recall that the B_{ij} are also positive. It is possible therefore to multiply both sides of (8) by a positive integer and obtain a new inequality whose coefficients are positive integers. The addition of an integer valued slack variable to this inequality results in an equation of type (9).

One way of counting the number of integer solutions has been developed previously by the author [7], and is restated here as Theorem 2. The notation "$k|s$" means "the integer k divides the integer s".

<u>Theorem 2</u> Let $F(a)$ denote the number of integer solutions to (7) and $P_i \equiv a_i(\bar{y}_i+1)$, $i = 1,\cdots,r$. Define

$$Q_s \equiv \sum_{\substack{i=1 \\ i \ni a_i | s}}^{r} a_i - \sum_{\substack{i=1 \\ i \ni P_i | s}}^{r} P_i \qquad s = 1,\cdots,a.$$

Then $F(0) = 1$ and

$$F(s) = (1/s) \sum_{\ell=1}^{s} Q_\ell F(s-\ell) \qquad s = 1,\cdots,a. \tag{10}$$

<u>Proof</u> See [7].

Although Theorem 2 can be used to calculate upper bounds on computational effort in the Accelerated Bound-and-Scan Algorithm, it is likely to be very expensive because the calculation of $F(a)$ by (10) requires $1+2+\cdots+a = \frac{a(a+1)}{2}$ multiplications and additions. Theorem 3 is another procedure (first presented in [8]) for calculating $F(a)$ which requires elementary arithmetic operations proportional to a.

<u>Theorem 3</u> Let $G_k(\omega)$ be defined for all $\omega = 0,1,\cdots,a$ and $k = 1,\cdots,r$ to be the number of integer solutions to the bounded variable linear Diophantine equation

$$\begin{aligned} &\textstyle\sum_{i=1}^{k} a_i y_i = \omega \\ &0 \le y_i \le \bar{y}_i && i = 1,\cdots,k \qquad (18) \\ &y_i \text{ is an integer} && i = 1,\cdots,k. \end{aligned}$$

Let $y_1' \equiv \min\{\bar{y}_1, [a/h_1]\}$. Then

$$G_1(\omega) = \begin{cases} 1 & \omega = 0,a_1,2a_1,\cdots,y_1'a_1 \\ 0 & \text{otherwise,} \end{cases} \qquad (19)$$

and for $k = 2,\cdots,r$,

$$G_k(\omega) = G_{k=1}(\omega) \qquad \omega = 0,1,\cdots,a_k-1, \qquad (20)$$

and

$$G_k(\omega+a_k) = \begin{cases} G_{k-1}(\omega+a_k)+G_k(\omega) & 0 \le \omega < a_k\bar{y}_k \\ G_{k-1}(\omega+a_k)+G_k(\omega)-G_{k-1}(\omega-a_k\bar{y}_k) & a_k\bar{y}_k \le \omega \le a - a_k. \end{cases} \qquad (21)$$

<u>Proof</u> See [8].

Algorithms based on Theorem 2 or Theorem 3 may be used to calculate the number of integer solutions to equation (9). It would be very useful, however, to approximate this number by an explicit function of $a_1,\cdots,a_r,a$, and r. Such an explicit function would provide information on the rate of growth of the number of eligible partial solutions up to variable r as r and the coefficients vary.

E. Cesaro [5] stated, without proof, that the number of positive integer solutions to the equation

$$\sum_{i=1}^{r} a_i y_i = a \tag{22}$$

is approximately equal to

$$a^{r-1}/((r-1)!\ \Pi_{i=1}^{r} a_i). \tag{23}$$

Lemmas 5 and 6 will put Cesaro's conjecture into a precise form. Theorem 4 will apply the results of the lemmas to the inequality

$$\sum_{i=1}^{k} y_i/B_{ii} \leq 1 \tag{24}$$

$$y_i \geq 0 \qquad i = 1,\cdots,k,$$

which is the same as (8) without the upper bound restrictions $y_i \leq \bar{y}_i$ $(i = 1,\cdots,k)$. The number of nonnegative integer solutions to (24) is an upper bound on the number of eligible partial solutions through variable k for both the IP problem and the cone problem.

The form of (22) which is most useful for the analysis here is the case in which one of the variables y_i, say y_1, represents an integer slack variable. Assume then that $a_1 = 1$ and the a_j $(j = 2,\cdots,r)$ are otherwise arbitrary strictly positive integers.

<u>Lemma 5</u> Let $a_1,\cdots,a_r$ and a be strictly positive integers. If $a_1 = 1$, the number of integer solutions $(y_1,\cdots,y_r)$ to equation (22), all of whose components y_j are strictly positive, is bounded above by (23).

<u>Proof</u> For $s = 1,\cdots,r$ and all integer i, let $F_s^*(i)$ denote the number of strictly positive integer solutions to the equation

$$\sum_{j=1}^{s} a_j y_j = i . \tag{25}$$

The proof of the lemma will be by induction on s. Consider first the case $s = 1$. Since $a_1 = 1$, $F_1^*(i) = 1/a_1 = 1$ for all strictly positive i. Now assume the lemma is true for all $s<k$. Notice that for $k\geq 2$,

$$F_k^*(i) = \sum_{j=1}^{[i/a_k]} F_{k-1}^* (i-ja_k). \tag{26}$$

Since $F_{k-1}^*(h)$ is nondecreasing in h,

$$F_{k-1}^*(i-ja_k) \leq (1/a_k) \sum_{p=0}^{a_k-1} F_{k-1}^* (i-ja_k+p). \tag{27}$$

Therefore,

$$\begin{aligned}
F_k^*(i) &\leq \sum_{j=1}^{[i/a_k]} (1/a_k) \sum_{p=0}^{a_k-1} F_{k-1}^* (i-ja_k+p) \\
&= (1/a_k) \sum_{\ell=i-[i/a_k]a_k}^{i-1} F_{k-1}^*(\ell) \\
&\leq (1/a_k) \sum_{\ell=0}^{i-1} F_{k-1}^*(\ell) \quad \text{since } F_{k-1}(\cdot) \geq 0 \\
&\leq (1/a_k) \sum_{\ell=0}^{i-1} \ell^{k-2}/((k-2)! \textstyle\prod_{j=1}^{k-1} a_j) \quad \text{by the induction hypothesis} \\
&\leq (1/a_k) \sum_{\ell=0}^{i-1} \int_{\ell}^{\ell+1} \{t^{k-2}/((k-2)! \textstyle\prod_{j=1}^{k-1} a_j)\} \, dt \\
&= (1/a_k) \int_0^i \{t^{k-2}/((k-2)! \textstyle\prod_{j=1}^{k-1} a_j)\} \, dt \\
&= i^{k-1}/((k-1)! \textstyle\prod_{j=1}^{k} a_j) ,
\end{aligned}$$

as was to be shown.

Lemma 5 stated that the number (23) is an upper bound on the number of strictly positive integer solutions to (22). Using very similar arguments, Lemma 6 below states that (23) is a lower bound on the number of nonnegative integer solutions to (22).

<u>Lemma 6</u> Let $a_1,\cdots,a_r$ be strictly positive integers. If $a_1 = 1$, the number of nonnegative integer solutions to equation (22) is bounded below by (23).

<u>Proof</u> For $s = 1,\cdots,r$ and for all integer i, let $F_s(i)$ denote the number of nonnegative integer solutions to (25). Of course, if $i<0$, $F_s(i) = 0$. Since $a_1 = 1$, $F_s(i)$ is a nondecreasing function of i, and $F_1(i) = (1/a_1) = 1$ for all nonnegative integer i. To complete the induction proof, assume the lemma is true for all $s<k$. For $k\geq 2$,

$$
\begin{aligned}
F_k(i) &= \sum_{j=0}^{[i/a_k]} F_{k-1}(i-ja_k) \\
&\geq \sum_{j=0}^{[i/a_k]} (1/a_k) \sum_{p=0}^{a_k-1} F_{k-1}(i-ja_k-p) \\
&= (1/a_k) \sum_{\ell=i-[i/a_k]a_k-a_k+1}^{i} F_{k-1}(\ell) \\
&= (1/a_k) \sum_{\ell=0}^{i} F_{k-1}(\ell) \quad \text{since } F_{k-1}(\ell) = 0 \text{ if } \ell < 0. \\
&\geq (1/a_k) \sum_{\ell=0}^{i} \ell^{k-2}/((k-2)!\,\Pi_{j=1}^{k-1} a_j) \quad \text{by the induction hypothesis} \\
&= (1/a_k) \sum_{\ell=1}^{i} \ell^{k-2}/((k-2)!\,\Pi_{j=1}^{k-1} a_j) \\
&\geq (1/a_k) \sum_{\ell=1}^{i} \int_{\ell-1}^{\ell} \{t^{k-2}/((k-2)!\,\Pi_{j=1}^{k-1} a_j)\}dt \\
&= (1/a_k) \int_0^i \{t^{k-2}/((k-2)!\,\Pi_{j=1}^{k-1} a_j)\}dt \\
&= i^{k-1}/((k-1)!\,\Pi_{j=1}^{k} a_j),
\end{aligned}
$$

as to be shown.

<u>Theorem 4</u> Let $\bar{E}_k$ be the number of nonnegative integer solutions to (24). Then

$$E_k^* \leq \bar{E}_k \leq E_k^{**}, \tag{28}$$

where

$$E_k^* = [\Pi_{i=1}^k B_{ii}]^k/(k!\ \Pi_{j=1}^k \langle \Pi_{\substack{i=1\\ i\neq j}}^k B_{ii} \rangle), \tag{29}$$

and

$$E_k^{**} = (1+[\Pi_{i=1}^k B_{ii} + \textstyle\sum_{j=1}^k \Pi_{\substack{i=1\\ i\neq j}}^k B_{ii}])^k/(k!\ \Pi_{j=1}^k[\Pi_{\substack{i=1\\ i\neq j}}^k B_{ii}]). \tag{30}$$

<u>Proof</u> Since $B_{ii}>0$ for all $i = 1,\cdots,k$, (24) is equivalent to

$$\textstyle\sum_{j=1}^k (\Pi_{\substack{i=1\\ i\neq j}}^k B_{ii})y_j \leq \Pi_{i=1}^k B_{ii} \tag{31}$$

$$y_j \geq 0 \qquad j = 1,\cdots,k.$$

The number of solutions to (31) is greater than or equal to the number of solutions to

$$\textstyle\sum_{j=1}^k \left\langle \Pi_{\substack{i=1\\ i\neq j}}^k B_{ii} \right\rangle y_j + y_{k+1} = [\Pi_{i=1}^k B_{ii}] \tag{32}$$

$$y_j \geq 0 \qquad j = 1,\cdots,k+1.$$

Applying Lemma 6 to (32) results in (29). To prove (30), consider the transformation of variables $\ell_j = y_j+1$; ℓ_j is a positive integer if and only if y_j is a nonnegative integer, and (31) is equivalent to

$$\textstyle\sum_{j=1}^k (\Pi_{\substack{i=1\\ i\neq j}}^k B_{ii})\ell_j \leq \Pi_{i=1}^k B_{ii} + \sum_{j=1}^k (\Pi_{\substack{i=1\\ i\neq j}}^k B_{ii}) \tag{33}$$

$$\ell_j > 0 \qquad j = 1,\cdots,k.$$

The number of solutions to (33) is bounded above by the number of solutions to

$$\textstyle\sum_{j=1}^k [\Pi_{\substack{i=1\\ i\neq j}}^k B_{ii}]\ell_j + \ell_{k+1} = 1 + [\Pi_{i=1}^k B_{ii} + \sum_{j=1}^k \Pi_{\substack{j=1\\ i\neq j}}^k B_{ii}] \tag{34}$$

$$\ell_j > 0 \qquad j = 1,\cdots,k+1.$$

Applying Lemma 5 to (34) results in (30), as was to be shown. Results similar to those of Theorem 4 may also be found in Padberg [16].

Notice that E_k^* is approximately equal to

$$\Pi_{i=1}^{k} B_{ii}/k!, \tag{35}$$

and E_k^{**} is approximately equal to

$$(1/k!)\ \{1+\Pi_{i=1}^{k}B_{ii}+\textstyle\sum_{j=1}^{k}\ \Pi_{\substack{i=1\\ i\neq j}}^{k}B_{ii}\}\{1+(1/\Pi_{i=1}^{k}B_{ii}) + \textstyle\sum_{j=1}^{k}(1/B_{jj})\}^{k-1}\ . \tag{36}$$

The expression (35) has an interesting geometrical intrepretation. If the $y_i (i = 1,\cdots,k)$ are considered as continuous variables, the region defined by (24) is a k-simplex with vertices $(0,\cdots,0)$, $(B_{11},0,\cdots,0)$, $(0,B_{22},0,\cdots,0),\cdots,(0,\cdots,0,B_{nn})$. By [10] its volume is precisely (35). That is, the estimate (35) of the number of lattice points within the region (24) is simply the volume of the region.

The Accelerated Bound-and-Scan Algorithm examines lattice points within an n-simplex. In the process of constructing the completions within the n-simplex, lattice points within lower dimensional projected k-simplices (partial solutions) are also examined. According to Theorem 1, the computational effort in the algorithm depends on the number of eligible partial solutions through each variable, that is, the number of lattice points within each projected k-simplex. One very crude way of estimating the computation time would be to use (35) or (36) as an estimate of E_k, the number of eligible partial solutions through variable k.

This raises a very natural question: Is there an optimal ordering of the variables which minimizes the estimated computation time? A change in the ordering of the variables corresponds to a change in the ordering of the binding constraints (9.8). Each of the $n!$ orderings of these constraints corresponds to a different inequality system (9.14). However, for a given system (9.14), the first n_1 variables can be reordered without disturbing the lower triangular property of (9.14) and (9.18). If either (35) or (36) is taken as the estimate of the number of eligible partial solutions through variable k, the first n_1 variables should be ordered in increasing order of the B_{ii}. This is equivalent to ordering the first n_1 variables in increasing order of their ranges within the n-simplex. Such an ordering is in agreement with the computational results reported by Cabot [4] on the knapsack problem.

Another question which can be answered to some extent by the use of volume as an approximation to the number of eligible partial solutions is: How does the quality of the starting solution affect the computational effort of the algorithm?

Denote by V_k the volume of the region defined by (24); V_k is given by (35). If an improved starting solution is known, the region corresponding to (24) is given by

$$\sum_{i=1}^{k} y_i/B_{ii} \leq \alpha$$

$$y_i \geq 0 \qquad i = 1,\cdots,k$$

for some $\alpha<1$. The volume of this region is equal to $V_{k,\alpha}$, where

$$V_{k,\alpha} = (\Pi_{i=1}^{k} \alpha B_{ii})/k! = \alpha^k V_k .$$

Therefore, the estimate of the number of eligible partial solutions through variable k is reduced by a factor of α^k. This can result in a substantial reduction in computational effort, particularly for large k. That the reduction in computational effort can be so great is in general agreement with the common observation that the efficiency of implicit enumeration algorithms is highly dependent on the quality of the starting solution.

The volume of a k-simplex may not be a good estimate of the number of lattice points within it. The extent to which this estimate can differ from the actual number is discussed next.

D. Volume and the Number of Lattice Points Within an n-Simplex

The relationship between the volume of a region and the number of lattice points within the region has for a long time been of interest in number theory. The Minkowski geometry of numbers [12] for example, specifies the minimum volume that an arbitrary convex body symmetric about the origin must have to guarantee the existence of a lattice point other than the origin within the body. Another example of interest is the following theorem proved by G. Pick in 1899.

Theorem 5 Let S be a simplex in E^2 (a triangle) with integer vertices and area $V(S)$. If b is the number of lattice points on the boundary of S and c is the number inside, then

$$V(S) = b/2 + c - 1. \tag{37}$$

See Coxeter [6] for a discussion of this theorem.

The following example shows that Theorem 5 cannot be generalized directly to simplices in E^k. For $k \geq 3$ the k-simplex in E^k whose $k+1$ integer vertices are $(0,0,\cdots,0)$, $(1,0,\cdots,0)$, $(0,1,0,\cdots,0),\cdots,$ $(0,0,\cdots,1,0)$ and $(1,1,\cdots,1,j)$ has volume $j/k!$, but the only lattice points within the simplex are the $k+1$ vertices. So for fixed k the volume may be any positive integer multiple of $1/k!$, but $c = 0$ and $b = k+1$, values independent of j.

Under the assumptions of Theorem 5, $(b+c)$, the total number of lattice points in triangle S, must satisfy

$$(b+c) \leq 2V(S) + 2 \tag{38}$$

$$(b+c) \geq V(S) + 1 . \tag{39}$$

The above example shows that (39) cannot be generalized directly. However, relationship (38) has an analog in Euclidean k-space E^k:

<u>Theorem 6</u> If S is a k-simplex in E^k with integer vertices and volume $V(S)$, the number of lattice points within S is at most $k!V(S)+k$.

<u>Proof</u> Let $x^{(0)},x^{(1)},\cdots,x^{(k)}$ be the vertices of S. Every point x in S can be expressed as convex combination of the vertices of S:

$$x = x^{(0)} + \sum_{i=1}^{k} \rho_i(x^{(i)}-x^{(0)})$$

$$\sum_{i=1}^{k} \rho_i \leq 1$$

$$\rho_i \geq 0 \quad i = 1,\cdots,k.$$

The only points x in S of interest are those which satisfy the linear congruences $x \equiv 0 \pmod 1$. Since $x^{(0)}$ is an integer vertex, $x \equiv 0 \pmod 1$ if and only if $x-x^{(0)} \equiv 0 \pmod 1$. Therefore the number of lattice points x in S is the number of distinct, but not necessarily integer, $(\rho_1,\cdots,\rho_k)$ which satisfy

$$\begin{aligned} &\sum_{i=1}^{k} \rho_i(x^{(i)}-x^{(0)}) \equiv 0 \pmod 1. \\ &\sum_{i=1}^{k} \rho_i \leq 1 \\ &\quad \rho_i \geq 0 \qquad i = 1,\cdots,k. \end{aligned} \tag{40}$$

Notice that the above congruence system in variables $\rho_1,\cdots,\rho_k$ has a non-singular coefficient matrix of integers. The solution set of this system is preserved by the following elementary row operations:

(a) multiplication of a row by -1,

(b) interchange of two rows,

(c) addition of an integer multiple of one row to another row.

Using only the row operations specified above it is possible to transform the congruence system into one in which the coefficient matrix is in Hermite normal form [1]:

$$\begin{pmatrix} B_{11} & & & \\ B_{21} & B_{22} & & \\ \vdots & & \ddots & \\ B_{k1} & B_{k2} & \cdots & B_{kk} \end{pmatrix} \begin{pmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_k \end{pmatrix} \equiv 0 \pmod 1, \tag{41}$$

$$\sum_{i=1}^{k} \rho_i \leq 1$$

$$\rho_i \geq 0 \qquad i = 1,\cdots,k,$$

where

$$\begin{array}{lll} B_{ij} & \text{is an integer} & j = 1,\cdots,k;\ i = j,\cdots,k; \\ B_{jj} > 0 & & j = 1,\cdots,k; \\ 0 \leq B_{ij} < B_{jj} & & j = 1,\cdots,k;\ i > j. \end{array}$$

We now show that the system (41) can be satisfied by at most

$$(B_{11}B_{22}\cdots B_{kk}) + k$$

distinct values of $(\rho_1,\cdots,\rho_k)$. If any $\rho_j = 1$, $j = 1,\cdots,k$, then all other ρ_i, $i \neq j$, must take the value zero. The k ways in which this can happen (some $\rho_j = 1$, $j = 1,\cdots,k$) determine the k lattice points $x^{(1)},\cdots,x^{(k)}$. Now consider the number of ways system (41) can be satisfied with the additional requirement that $\rho_j<1$ for all $j = 1,\cdots,k$. To satisfy the congruence system ρ_1 must be chosen so that

$$B_{11}\rho_1 \equiv 0 \pmod 1.$$

Since we have counted $\rho_1 = 1$ already, ρ_1 can take on at most the B_{11} values $0,\frac{1}{B_{11}},\cdots,\frac{B_{11}-1}{B_{11}}$. Now suppose that $\rho_1,\cdots,\rho_{s-1}$ have been set at feasible values $\rho_1^*,\cdots,\rho_{s-1}^*$. Then ρ_s must be chosen so that if f is defined to be the fractional part of $\sum_{j=1}^{s-1} B_{sj}\rho_j^*$,

$$B_{ss}\rho_s \equiv -f \pmod 1.$$

Let
$$t = \begin{cases} 0 & f = 0 \\ \frac{1-f}{B_{ss}} & 0 < f < 1. \end{cases}$$

Then, excluding the case $\rho_s = 1$ which has already been counted, ρ_s can take on at most the B_{ss} values $t, t+\frac{1}{B_{ss}},\cdots,t+\frac{B_{ss-1}}{B_{ss}}$. Therefore, there are at most $(B_{11}\cdots B_{kk})$ ways in which the system (41) can be satisfied by distinct $(\rho_1,\cdots,\rho_k)$ for which no $\rho_j = 1$, $j = 1,\cdots,k$. Since there are k ways in which one can choose some $\rho_j = 1$, $j = 1,\cdots,k$, there are $(B_{11}\cdots B_{kk})+k$ ways to solve (41) and hence (40).

The relationship of the upper bound to the volume of the k-simplex is very interesting. The volume $V(S)$ of an k-simplex S with vertices $x^{(0)},\cdots,x^{(k)}$ is given by [10]:

$$V(S) = \frac{1}{k!}\left|\det\begin{pmatrix}(x_1^{(1)}-x_1^{(0)}) & \cdots & (x_1^{(k)}-x_1^{(0)})\\ \vdots & & \vdots\\ (x_k^{(1)}-x_k^{(0)}) & \cdots & (x_k^{(k)}-x_k^{(0)})\end{pmatrix}\right| ,$$

where $|Z|$ denotes the absolute value of Z. The matrix whose determinant is taken to compute $V(S)$ is just the coefficient matrix of the congruence system (40). The absolute value of the determinant is not changed by the row operations in (a), (b), (c). Since $B_{jj}>0$, $j = 1,\cdots,k$,

$$B_{11} \cdots B_{kk} = k!\, V(S).$$

We may conclude that if S is an k-simplex with integer vertices and volume $V(S)$, the number of lattice points within S is at most

$$k!V(S) + k.$$

This upper bound is sharp, as illustrated by the following example. The simplex with vertices $0,e_1,\cdots,e_k$, where e_j is the j^{th} unit coordinate vector, is called the standard k-simplex. The only lattice points in this k-simplex are the $k+1$ vertices. Let us verify that this agrees with the upper bound $(B_{11}\cdots B_{kk})+k$. Letting $x^{(0)} = 0$, $x^{(i)} = e_i$, $i = 1,\cdots,k$, the congruence system is already in Hermite normal form with $B_{jj} = 1$, $j = 1,\cdots,k$, so $(B_{11}\cdots B_{kk})+k = k+1$.

E. Estimating the Number of Eligible Partial Solutions

In the previous section it was shown that the volume of a k-simplex with integer vertices may differ from he number of lattice points within it by as much as a factor of $k!$ In spite of this possible shortcoming, the volume remains a useful, though rather crude, tool for specifying the order in which the variables should be placed and for demonstrating the importance of finding a good starting solution.

In this section a somewhat more sophisticated technique is used to approximate the number of eligible partial solutions through each variable j. It is based on the observation that the number of solutions to (7) is equal to the number of eligible partial solutions through variable j if $j \leq n_1$, but if $j > n_1$, the number of solutions to (7) is an upper bound which may be significantly higher. Theorem 3 is appropriate for $j \leq n_1$, but for $j > n_1$ another approach would be useful.

The approach developed in this section is to compare the basic enumeration scheme of the algorithm to a stochastic process in which the extreme point weights are random variables. The design of the stochastic model will require certain subjective judgments about the behavior of the algorithm.

To simplify the analysis, assume that $\bar{\rho}_j = 1$ for all $j = 1,\cdots,n$. This assumption is required only for the approximation of E_j for $j > n_1$. For $j \leq n_1$, Theorem 3 (which allows $\bar{\rho}_j \neq 1$) may be used to calculate E_j.

If a total enumeration scheme involving the ρ_k variables were used to solve the IP problem, all lattice points within the parallelepiped,

$$\begin{aligned} y &= y^{(0)} + \textstyle\sum_{i=1}^{n} \rho_i (y^{(i)} - y^{(0)}) \\ 0 &\leq \rho_i \leq 1 \qquad\qquad i = 1,\cdots,n, \end{aligned} \tag{44}$$

corresponding to (9.16) would be enumerated. Of course, the algorithm does not enumerate all lattice points within the parallepiped, but it does consider partial solutions $(\rho_1,\cdots,\rho_k)$ for which $\sum_{j=1}^{k} \rho_j \leq 1$. In the design of the stochastic model, the first subjective judgment is that the behavior of the extreme point weights $\rho_1,\cdots,\rho_n$ in a total enumeration scheme may be approximated by n independent random variables uniformly distributed between zero and one. If the extreme point weights were so distributed, any realization $(\rho_1,\cdots,\rho_n)$ would correspond to a random "continuous" point in the parallepiped, rather than to a random lattice point.

Recall that the objective of this analysis is to approximate the number of eligible partial solutions through variable k for every $k = n_1+1,\cdots,n$. Let k be any integer, $n_1<k\leq n$, and suppose $(\rho_1,\cdots,\rho_{k-1})$ is an eligible partial solution through variable $k-1$. Then there exists an eligible partial solution through variable k of the form $(\rho_1,\cdots,\rho_{k-1},\rho_k)$ if and only if conditions (9.23) and (9.24) hold:

$$f + B_{kk}\rho_k \equiv 0 \pmod 1 \tag{9.23}$$

$$0 \leq \rho_k \leq 1 - \textstyle\sum_{j=1}^{k-1} \rho_j , \tag{9.24}$$

where f is the fractional part of

$$d_k + \textstyle\sum_{j=1}^{k-1} B_{kj}\rho_j .$$

If the ρ_j are assumed to be random variables in the model then f is also a random variable. In the design of the stochastic model, the second subjective judgment is that for $k>n_1$ the fractional part f may be approximated by a random variable uniformly distributed between zero and one, and that f is independent of the partial sum $\sum_{j=1}^{k-1} \rho_j$. Under these assumptions, $f = 0$ occurs with probability zero for $k>n_1$, so that the number of eligible partial solutions through variable k, for any partial solution $(\rho_1,\cdots,\rho_{k-1})$, is the largest integer ω_k such that

$$(1-f)/B_{kk} + (\omega_k-1)/B_{kk} \leq 1 - \textstyle\sum_{j=1}^{k-1} \rho_j ,$$

or

$$\omega_k = [B_{kk}(1-\textstyle\sum_{j=1}^{k-1} \rho_j) + f].$$

Let φ_k denote the random variable

$$\varphi_k = B_{kk}\,(1-\textstyle\sum_{j=1}^{k-1} \rho_j).$$

Lemma 7 Let f be a random variable uniformly distributed between zero and one, and suppose f is independent of the random variable φ_k. If $\omega_k = [\varphi_k+f]$, then

$$\mathcal{E}(\omega_k) = \mathcal{E}(\varphi_k),$$

where $\mathcal{E}$ denotes the expectation operator.

Proof Let $\mathcal{E}(\omega_k|\varphi_k)$ denote the conditional expectation of ω_k given φ_k. Then

$$\begin{aligned}\mathcal{E}(\omega_k|\varphi_k) &= [\varphi_k]P(\varphi_k-[\varphi_k]+f<1) + ([\varphi_k]+1)P(\varphi_k-[\varphi_k]+f\geq 1)\\ &= [\varphi_k]P(f<1-\varphi_k+[\varphi_k]) + ([\varphi_k]+1)P(f\geq 1-\varphi_k+[\varphi_k])\\ &= [\varphi_k]\ \ (1-\varphi_k+[\varphi_k]) + ([\varphi_k]+1)(\varphi_k-[\varphi_k])\\ &= \varphi_k.\end{aligned}$$

Therefore,

$$\mathcal{E}(\omega_k) = \mathcal{E}(\mathcal{E}(\omega_k|\varphi_k)) = \mathcal{E}(\varphi_k),$$

as was to be shown.

At this point it is appropriate to review the motivation for considering the stochastic model. The purpose of the model is to provide a tractable means of estimating E_k, the number of eligible partial solutions through ρ_k. For each eligible partial solution through variable $(k-1)$, there will be a certain number, ω_k, of values of ρ_k which yield an eligible partial solution through variable k. In the stochastic model the E_k and ω_k (for $k>n_1$) are random variables with the property that

$$\mathcal{E}(E_k | E_{k-1} = \ell) = \ell\mathcal{E}(\omega_k) = \ell\mathcal{E}(\varphi_k).$$

Therefore, in the stochastic model,

$$\mathcal{E}(E_k) = \mathcal{E}(E_{k-1})\,\mathcal{E}(\varphi_k).$$

By induction, for $k>n_1$

$$\mathcal{E}(E_k) = (\Pi_{j=n_1+1}^{k}\,\mathcal{E}(\varphi_j))E_{n_1}, \tag{45}$$

where E_{n_1} is a constant which can be found from Theorem 3.

It is still necessary to interpret and calculate the expression $\mathcal{E}(\varphi_k)$ in (43). If a total enumeration scheme were used to solve the IP problem, all values of ρ_k given in (9.27) would be enumerated. However, an implicit enumeration scheme is used which considers only partial solutions $(\rho_1,\cdots,\rho_k)$ for which

$$\textstyle\sum_{j=1}^{k}\rho_j \leq 1 \qquad\qquad k = 1,\cdots,n.$$

The following model is used to approximate the behavior of the partial sums $H_k \equiv \sum_{j=1}^{k}\rho_j$. Assume initially that the ρ_j are independent random variables, uniformly distributed between zero and one. The uniform distribution is used to approximate the distribution of values that each ρ_j would assume in a total enumeration scheme. In a total enumeration scheme H_k may be approximated by the sum of k independent, uniformly distributed random variables. Each ρ_j would have mean $\mathcal{E}(\rho_j) = 1/2$ and variance $v(\rho_j) = 1/12$. By the Central-limit theorem [15, p. 149], H_k would be approximately normally distributed with mean $k/2$ and variance $k/12$. However, the algorithm allows only the values $\rho_1,\cdots,\rho_k$ for which $0 \leq H_k \leq 1$. Therefore the distribution of H_k in this implicit enumeration scheme may be approximated by a truncated normal distribution.

The following lemma is easy to verify.

<u>Lemma 8</u> Let H_k^* be a normal random variable with mean $k/2$ and variance $k/12$, and let H_k be distributed according to the conditional distribution of H_k^*, given $0 \leq H_k^* \leq 1$. Then

$$\mathcal{E}(H_k) = k/2 - \sqrt{k/12}\,\{n(q_k) - n(-\sqrt{3k})\}/\{N(q_k) - N(-\sqrt{3k})\}, \quad (46)$$

where

$$q_k = \sqrt{12}\,(1-k/2)/\sqrt{k}\ ,$$

and $n(x)$ and $N(x)$ are respectively the density function and cumulative distribution function of a normal random variable with mean zero and variance one.

Note that

$$\mathcal{E}(\varphi_k) = B_{kk}(1-\mathcal{E}(H_{k-1})). \quad (47)$$

The suggested approximation to the number of eligible partial solutions through ρ_k is given by $\mathcal{E}(E_k)$ in (45). Combining (45), (46), and (47), the estimate $\mathcal{E}(E_k)$ for $k > n_1$ is

$$\mathcal{E}(E_k) = E_{n_1}\Pi_{j=n_1+1}^{k} B_{jj}\{1-(j-1)/2+\sqrt{(j-1)/12}\left(n(q_{j-1})-n(-\sqrt{3(j-1)})\right)\Big/\left(N(q_{j-1})-N(-\sqrt{3(j-1)})\right)\}.$$

Let

$$\alpha_j = 1-(j-1)/2+\sqrt{(j-1)/12}\left(n(q_{j-1})-n(-\sqrt{3(j-1)})\right)\Big/\left(N(q_{j-1})-N(-\sqrt{3(j-1)})\right). \quad (48)$$

Then the α_j may be calculated using tables of the normal distribution. The values of α_j are independent of the data of the problem and may be stored in a table in a computer program. Using this table, the estimate of E_k is

$$\mathcal{E}(E_k) = E_{n_1}\Pi_{j=n_1+1}^{k} B_{jj}\alpha_j. \quad (49)$$

F. Checking the Nonbinding Constraints

Once a completion within the n-simplex has been generated by the simplified version of the Accelerated Bound-and-Scan Algorithm, the m nonbinding constraints are checked for feasibility. This process requires, as Theorem 1 notes, at most mn additions and multiplications for each completion. However, if the completion violates one of these constraints the computation needed may be significantly less because control returns to the basic enumeration scheme as soon as the infeasibility has been discovered.

The m nonbinding constraints are checked in some particular sequence. If P_i is defined to be the probability that a completion satisfies the i^{th} nonbinding constraint, the expected number of multiplications and additions needed to check the nonbinding constraints in the course of the algorithm is given by

$$\mathcal{E}(E_n)\ \{\textstyle\sum_{i=1}^{m-1} (in)(1-P_i)(\Pi_{j=1}^{i-1}P_j) + mn\ \Pi_{j=1}^{m-1}P_j\}, \tag{50}$$

where $\mathcal{E}(E_n)$ represents the expected number of completions.

If the P_i $(i = 1,\cdots,m)$ are know, the nonbinding constraints should be ordered so that

$$P_1 \leq P_2 \leq \cdots \leq P_m \tag{51}$$

to minimize (50). The ordering (51) simply says that the constraints most likely to be violated by a completion should be checked first. However, the P_i $(i = 1,\cdots,m)$ are not known, so it becomes necessary to estimate them.

One estimate of P_i would be

$$P_i \cong V(R_i)/V(S) ,$$

where $V(S)$ denotes the volume of the n-simplex S and $V(R_i)$ is the volume of the region R_i, where $R_i = \{x\varepsilon S:\ x$ satisfies the i^{th} nonbinding constraint$\}$. The calculation of $V(S)$ by (42) is relatively simple, but the calculation of $V(R_i)$, though mathematically trivial, is somewhat complicated by the difficulty of determining the limits of integration.

Another estimate of the P_i could be obtained by a simple experiment. Suppose N points $(\rho_1,\cdots,\rho_n)$ are generated randomly within the n-simplex. Choose the estimate

$$P_i \cong N_i/N , \tag{52}$$

where N_i is the number of points satisfying the i^{th} nonbinding constraint. To generate a single point, select each ρ_i $(i = 1,\cdots,n)$ according to a uniform distribution on the interval $[0,1]$. If $H_n \doteq \sum_{i=1}^{n}\rho_i \leq 1$, then

$(\rho_1,\cdots,\rho_n)$ represents a point within the n-simplex. If $H_n>1$, continue by selecting additional $(\rho_1,\ldots,\rho_n)$ until the condition $H_n\leq 1$ is satisfied.

The major [illegible] back to this experiment is that for large n the probability that $\sum_{j=1}^{n}\rho_j\leq 1$ is very small. It is likely that many random points (on the average, n!) within the paralleplped (44) would have to be generated for every point within the n-simplex. Rather than perform the experiment every time a new integer programming problem is to be solved, it would be possible to construct a table of N random points $(\rho_1,\cdots,\rho_n)$ for which $\sum_{j=1}^{n}\rho_j\leq 1$. The values of $(\rho_1,\cdots,\rho_n)$ in this table are independent of the data for any particular integer programming problem. The table would have to be constructed only once, and it would be used for every problem solved by the algorithm.

Although the table would have to be constructed only once, the number of points (approximately Nn!) which would have to be generated for moderately large n might be prohibitively large. For this reason the following modification to the experiment is suggested for large n. Recall that every point x within an n-simplex S with vertices $x^{(0)},x^{(1)},\cdots,x^{(n)}$ may be represented as

$$x = x^{(0)} + \sum_{i=1}^{n}\rho_i\,(x^{(i)}-x^{(0)})$$

where

$$\sum_{i=1}^{n}\rho_i \leq 1$$

$$1 \geq \rho_i \geq 0 \qquad i = 1,\cdots,n.$$

In the experiment each ρ_i is selected according to a [0,1] uniform distribution, and if $H_n = \sum_{i=1}^{n}\rho_i\leq 1$, $(\rho_1,\cdots,\rho_n)$ represents a point within the n-simplex. If it were possible to generate the ρ_i directly so that $H_n\leq 1$, the major drawback to this approach would be eliminated.

Consider the following approximation to the experiment. It is based on the observation that, for points in the n-simplex, H_n is approximately distributed as a truncated normal distribution.

Let H_k^*, for $k = 1,\cdots,n$, be a normal random variable with mean 1/2 and variance $k/12$, and let $R_k(t)$ be distributed according to the conditional distribution of H_k^*, given $0 \le H_k^* \le t$. In the experiment, first generate a value for $R_n(1)$, and let $\rho_n = 1-R_n(1)$. Next generate a value for $R_{n-1}(1-\rho_n)$, and let $\rho_{n-1} = 1-R_{n-1}(1-\rho_n)$. In general, the k^{th} extreme points weight ρ_k is generated by letting

$$\rho_k = (1-\textstyle\sum_{i=k+1}^{n}\rho_i) - R_k(1-\sum_{i=k+1}^{n}\rho_i),$$

where values for $\rho_{k+1},\cdots,\rho_n$ have already been specified.

The design of the experiment is explained in the following way. There are (n+1) extreme points in the n-simplex. In order to determine all ρ_i $(i = 0,1,\cdots,n)$, it is necessary to specify only the weights ρ_i on n of the extreme points since $\sum_{i=0}^{n}\rho_i = 1$. Note that the partial sum $\sum_{i=0}^{n-1}\rho_i$ is approximately distributed as $R_n(1)$ within the n-simplex. Given $\rho_n,\rho_{n-1},\cdots,\rho_{k+1}$, the partial sum

$$\textstyle\sum_{i=0}^{k-1} \rho_i$$

is approximately distributed as $R_k(1-\sum_{i=k+1}^{n}\rho_i)$, so that given $\rho_n,\cdots,\rho_{k+1}$, the kth extreme point weight $\rho_k = (1-\sum_{i=k+1}^{n}\rho_i-\sum_{i=0}^{k-1}\rho_i)$ is approximately distributed as $1-\sum_{i=k+1}^{n}\rho_i-R_k(1-\sum_{i=k+1}^{n}\rho_i)$. Each $\rho_n,\rho_{n-1},\cdots,\rho_1$ is generated, in that order, thereby guaranteeing that $H_n \le 1$.

Although this approach of using the truncated normal distribution eliminates excessive computation which would otherwise be required, it has one outstanding weakness: the use of $R_k(1-\sum_{i=k+1}^{n}\rho_i)$ as an approximation for small k to the partial sum $\sum_{i=0}^{k-1}\rho_i$. This difficulty may be circumvented by performing a hybrid experiment. Let k_0 be some fixed positive integer such that for $k \geq k_0$, the truncated normal distribution is an "acceptable" approximation to the partial sum $\sum_{i=0}^{k-1}\rho_i$. Generate $\rho_n,\cdots,\rho_{k_0}$ using $R_k(\cdot)$ as described above for $k = n,n-1,\cdots,k_0$. To generate $\rho_1,\cdots,\rho_{k_0-1}$, select each ρ_i $(i = 1,\cdots,k_0-1)$ according to a uniform $[0,1]$ distribution. If $H_{k_0-1} \equiv \sum_{i=1}^{k_0-1}\rho_i \leq (1-\sum_{i=k_0}^{n}\rho_i)$, then $(\rho_1,\cdots,\rho_n)$ represents a "random" point within the n-simplex. If this condition on H_{k_0-1} is not satisfied, continue selecting additional $(\rho_1,\cdots,\rho_{k_0-1})$ until it is satisfied.

G. An Example

The data for the example below is given in the form of (7). Suppose that all eligible partial solutions through variable ρ_3 are given by the solutions to the system

$$\begin{aligned} 4\rho_1 &\equiv 0 \pmod 1 \\ 8\rho_2 &\equiv 0 \pmod 1 \\ 1/8 + \rho_1 + 4\rho_2 + \rho_3/4 &\equiv 0 \pmod 1, \end{aligned} \tag{53}$$

where

$$\rho_1 + \rho_2 + \rho_3 \leq 1$$

and

$$\rho_i \geq 0 \qquad (i = 1,2,3).$$

There are only two solutions to (53), given by $(\rho_1,\rho_2,\rho_3) = (1/4, 1/8, 1/2)$ and $(\rho_1,\rho_2,\rho_3) = (1/4, 3/8, 1/2)$. According to Lemma 3, the number of solutions to (53) is bounded above by the number of solutions to the system

$$
\begin{aligned}
4\rho_1 &\equiv 0 \pmod 1 \\
8\rho_2 &\equiv 0 \pmod 1 \qquad (54)\\
(\rho_3)/4 &\equiv 0 \pmod 1,
\end{aligned}
$$

where

$$\rho_1 + \rho_2 + \rho_3 \leq 1$$

and $\rho_i \geq 0 \qquad (i = 1,2,3).$

There are 24 solutions to (54), indicating that Lemma 3 does not necessarily provide good approximation. However, the volume of the 3-simplex is $8/3! = 4/3$, and the approximation (45) gives $\mathcal{E}(E_3) = 1.88$.

H. Summary of Estimates and Bounds on Computational Effort

This chapter has examined in detail the computational effort required by a simplified version of the Accelerated Bound-and-Scan Algorithm. It was shown in Theorem 1 that the computational effort (additions and multiplications) may be expressed as linear functions of the E_k, the number of eligible partial solutions through each variable k.

Corollary 1 If A_T is the number of additions and M_T the number of multiplications required by the iterative part of the simplified algorithm,

$$A_T \leq \sum_{j=n_1}^{n-1} (j+3)E_j + mn\, E_n + 3\sum_{j=1}^{n-1} E_j + 2E_n,$$

and

$$M_T \leq \sum_{j=n_1}^{n-1} (j+1)E_j + mn\, E_n.$$

Upper bounds for A_T and M_T are given in Corollary 1. These upper bounds are exact if no improved solution is found and if all m non-binding constraints must be checked for every completion generated.

Upper bounds on the E_k were developed next. Once upper bounds on the E_k are known, upper bounds on the A_T and M_T given in Corollary 1 may be calculated.

Corollary 2 E_k is bounded above by the number of solutions to the inequality

$$\sum_{i=1}^{k} y_i/B_{ii} \leq 1 \tag{55}$$

$$0 \leq y_i \leq \bar{y}_i, \quad y_i \text{ integer } \quad i = 1,\cdots,k.$$

Methods for calculating the number of solutions to (55) were given in Theorems 2 and 3. The bound of Corollary 2 is exact if $k \leq n_1$.

Let $\bar{E}_k$ be the number of solutions to (55) when the upper bound restriction $y_i \leq \bar{y}_i$ is dropped. Crude upper and lower bounds on $\bar{E}_k$ as explicit functions of the B_{ii} were derived in Theorem 4 in order to show that the nonbasic variables should be ordered according to increasing B_{ii}. The lower bound on $\bar{E}_k$ was shown to be the volume of the k-simplex which is the projection of the n-simplex onto the first k variables. The importance of having a good starting solution was discussed.

In Section D it was shown that the volume of a k-simplex with integer vertices may differ from the number of lattice points within it by as much as a factor of $k!$

In Section E an estimate of E_k for $k > n_1$ was developed, and in Section F an experiment was designed to find an estimate P_i of the probability that a completion satisfies the i^{th} non-binding constraint. Assuming that the non-binding constraints are ordered so that

$$P_1 \leq P_2 \leq \cdot \cdot \cdot \leq P_m,$$

it is possible to estimate the computational effort in the algorithm as follows.

Corollary 3 Let $\mathcal{E}(A_T)$ and $\mathcal{E}(M_T)$ denote respectively the estimated number of additions and multiplications in the algorithm. Then

$$\mathcal{E}(A_T) = (n_1+3)E_{n_1} + \sum_{k=n_1+1}^{n-1}(k+6)E_{n_1}(\Pi_{j=n_1+1}^{k}B_{jj}\alpha_j)$$

$$+ E_{n_1}(\Pi_{j=n_1+1}^{n}B_{jj}\alpha_j)\{2+\sum_{i=1}^{m-1}(in)(1-P_i)(\Pi_{j=1}^{i-1}P_j) + mn\Pi_{j=1}^{m-1}P_j\}$$

$$+ 3\sum_{j=1}^{n_1} E_j,$$

and

$$\mathcal{E}(M_T) = (n_1+1)E_{n_1} + \sum_{k=n_1+1}^{n-1}(k+1)E_{n_1}(\Pi_{j=n_1+1}^{k}B_{jj}\alpha_j)$$

$$+ E_{n_1}(\Pi_{j=n_1+1}^{n}B_{jj}\alpha_j)\ \{\sum_{i=1}^{m-1}(in)(1-P_i)(\Pi_{j=1}^{i-1}P_j) + mn\Pi_{j=1}^{m-1}P_j\},$$

where $E_1,\cdots,E_{n_1}$ are calculated using Theorem 3, the P_i are given by (52), and the α_j are given in (48).

Corollary 3 follows directly from Corollary 1 and equation (49) and (50).

The estimates of Corollary 3 are conservative in the sense that they assume no improved solution is found during the course of the algorithm. Such an assumption may be very realistic, since the Hillier heuristic procedures often find an optimal solution to the problem. Computational experience reported by Hillier [13] indicates that his better heuristic procedures obtained an optimal solution to the IP problem on approximately half of his test problems. When his more powerful multiple-solution approaches were used, an optimal solution was found on about three quarters of the problems If such a heuristic procedure is used to obtain a starting solution in the algorithm, and if that solution is actually optimal, the estimates of Corollary 3 may be quite accurate. If an improved solution is found, the estimates may be somewhat conservative .

This chapter has presented bounds and estimates of the computational effort required by a simplified version of the Accelerated Bound-and-Scan Algorithm to solve any given problem. The simplified version of the algorithm includes the basic enumeration scheme and bounds $\bar{\rho}_i$ on variables ρ_i used in the skeleton algorithm, but it does not include the scanning procedure or the elimination of redundant constraints. A very natural question to consider at this point is how the bounds and estimates would change if the more sophisticated skeleton algorithm were analyzed. In general one would expect the estimates given for the simplified algorithm to be somewhat conservative if the skeleton algorithm were used. However, it would be possible for the skeleton algorithm to require more computational effort on a given problem. Fortunately, it is possible to bound this additional computational effort, and as a result derive bounds for the skeleton algorithm.

Once the right-hand side of (9.35) has been calculated once for each nonbinding constraint at the outset of the algorithm, no additional multiplications or additions are required to check for redundancy in the iterative part of the skeleton algorithm.

The scanning procedure may require more additions and multiplications than the corresponding procedure for checking the nonbinding constraints in the simplified algorithm. The scanning procedure can be entered only once for each of the E_n possible completions generated by the skeleton algorithm. In the worst possible case, all m nonbinding constraints would be checked for feasibility for each completion before the scanning procedure is entered. The most computation which could occur in the scanning procedure (for a given completion) would result in $n(n-1)$ additions and $(n(n-1)/2)+1$ multiplications in Step 5 and three multiplications and additions in Step 7, or $n(n-1)+3$ additions and $(n(n-1)/2)+4$ multiplications in all.

<u>Corollary 4</u> If A_T is the number of additions and M_T the number of multiplications required by the iterative part of the skeleton version of the Accelerated Bound-and-Scan Algorithm,

$$A_T \le \sum_{j=n_1}^{n-1} (j+3)E_j + 3\sum_{j=1}^{n-1} E_j + (mn+n(n-1)+5)\, E_n,$$

and

$$M_T \le \sum_{j=n_1}^{n-1} (j+1)E_j + (mn+n(n-1)/2+4)E_n,$$

where the E_j (or upper bounds on the E_j) may be calculated from Theorem 2 or Theorem 3.

The proof of Corollary 4 follows from Corollary 1.

REFERENCES

1. BRADLEY, Gordon H., "Algorithms for Hermite and Smith Normal Matrices and Linear Diophantine Equations", Report No. 34, Department of Administrative Sciences, Yale University, June 1970

2. BRADLEY, Gordon H., "Equivalent Integer Programs and Canonical Problems", Management Science, Vol. 17, No. 5 (January 1971), pp. 354-366

3. BRADLEY, Gordon H. and WAHI, Pran N., "An Algorithm for Integer Linear Programming: A Combined Algebraic and Enumeration Approach", Report No. 29, Department of Administrative Sciences, Yale University, December 1969 (Revised February 1971)

4. CABOT, Victor A., "An Enumeration Algorithm for Knapsack Problems", Operations Research, Vol. 18, No. 2, (March-April 1970), pp. 306-31[illegible]

5. CESARO, E., Memoires de la Societe Royale des Sciences de Liege, Series 2, Vol. 10, p. 229 (May 1883).

6. COXETER, H. S. M., Introduction to Geometry, Wiley, New York 1969.

7. FAALAND, Bruce, "On the Number of Solutions to a Diophantine Equation", Technical Report No. 18, Contract N00014-67-A-0112-0058, Department of Operations Research, Stanford University, November 12, 1970 (to appear in Journal of Combinatorial Theory)

8. FAALAND, Bruce, "Solution of the Value-Independent Knapsack Problem by Partitioning", Technical Report No. 22, Contract N00014-67-A-0112-0058, Department of Operations Research, Stanford University, February 12, 1971 (to appear in Operations Research)

9. FAALAND, Bruce and HILLIER, Frederick S., "The Accelerated Bound-and-Scan Algorithm for Integer Programming", Technical Report No. 34, Contract N00014-67-A0112-0058, and Technical Report No. 9, Contract NSF GJ-31521, Department of Operations Research, Stanford University, May 15, 1972 (submitted to Operations Research)

10. FLEMING, Wendell H., Functions of Several Variables, Addison Wesley, Palo Alto, 1965

11. GUPTA, H., "Partitions - A Survey", JNBS-B, Vol. 74B, No. 1 (January-March 1970), pp. 1-29

12. HANCOCK, Harris, Foundations of the Theory of Algebraic Numbers, MacMillan Company, New York, 1931

13. HILLIER, Frederick S., "Efficient Heuristic Procedures for Integer Linear Programming with an Interior", Operations Research, Vol. 17, No. 4 (July-August 1969), pp. 600-637

14. HILLIER, Frederick S., "A Bound-and-Scan Algorithm for Pure Integer Linear Programming with General Variables", Operations Research, Vol. 17, No. 4 (July-August 1969), pp. 638-679.

15. MOOD, Alexander M. and GRAYBILL, Franklin A., Introduction to the Theory of Statistics, McGraw-Hill, New York, 1963

16. PADBERG, Manfred W., "A Remark on an Inequality for the Number of Lattice Points in a Simplex", Management Sciences Research Report No. 222, Carnegie-Mellon University, October 1970.

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Department of Operations Research
Stanford University
Stanford, California 94305

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE

Estimates and bounds on computational effort in the accelerated bound-and-scan algorithm

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Report

5. AUTHOR(S) (Last name, first name, initial)

Bruce H. Faaland

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May 15, 1972	38	16

8a. CONTRACT OR GRANT NO.
N00014-67-A-0112-0058

b. PROJECT NO.
(NR-047-061)

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
Technical Report No. 35

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
Technical Report No. 10 - NSF GJ-31521

10. AVAILABILITY/LIMITATION NOTICES

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

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Operations Research Program (Code 434)
Office of Naval Research
Washington, D.C. 20360

13. ABSTRACT

Methods for obtaining estimates and bounds on computational effort in the Accelerated bound-and-scan algorithm are presented. The basic tools of analysis used are various results from the theory of partitions of numbers, the central-limit theorem, and geometric interpretations of the algorithm. The results of the analysis are used to provide partial answers to questions such as (a) What is the greatest amount of time the algorithm could require to solve a given problem? (b) What is the expected computation time? (c) How should the variables and constraints be ordered? (d) How does the computation time depend on the number of variables? and (e) How is the computation time affected by the quality of the starting solution?

DD FORM 1 JAN 64 1473

UNCLASSIFIED
Security Classification

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
OPERATIONS RESEARCH MATHEMATICAL PROGRAMMING INTEGER PROGRAMMING INTEGER LINEAR PROGRAMMING ALGORITHMS INTEGER PROGRAMMING ALGORITHMS IMPLICIT ENUMERATION ALGORITHMS						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.