UNITED ARAB REPUBLIC

THE INSTITUTE OF NATIONAL PLANNING

التخطيط القومي

Memo No. 536

Notes on Operations Research
II. Integer Programming.

By

Dr. Hamdy A. Taha

Operations Research Center

February 1965.

CAIRO

1. Introduction

An integer linear problem is an ordinary linear programming problem with some additional constraints which force the optimal solution to take non-negative integer values only. In other words given an L.P. problem, it is required to select, from among all possible integers in the solution space, the values of the variables which are feasible ($\geqq 0$) and which optimize the value of the objective function. Recent liturature on integer linear programming includes some applications to different practical problems. In this brief summary, main attention is given to discussing the method of solution which is due to Gomory[1]. Two numerical example will be given which show the steps for finding the optimal integer solutions.

It is necessary however, before proceding with this memo, that the reader be familiar with the mathematics of the simplex method for solving L.P. problems.

2. raphical Representation of the Problem:

Consider the ordinary L.P. problem shown in Figure (1). This problem is given by:

maximize R

Subject to constraints (1), (2), (3), and (4)

---

1. R.E. Gomory, "Quttine of An Algorithm for Integer Solutions to Linear Programs," Bulletin of the American Mathematical Society, Vol. 64 (September, 1958), pp. 275-278.

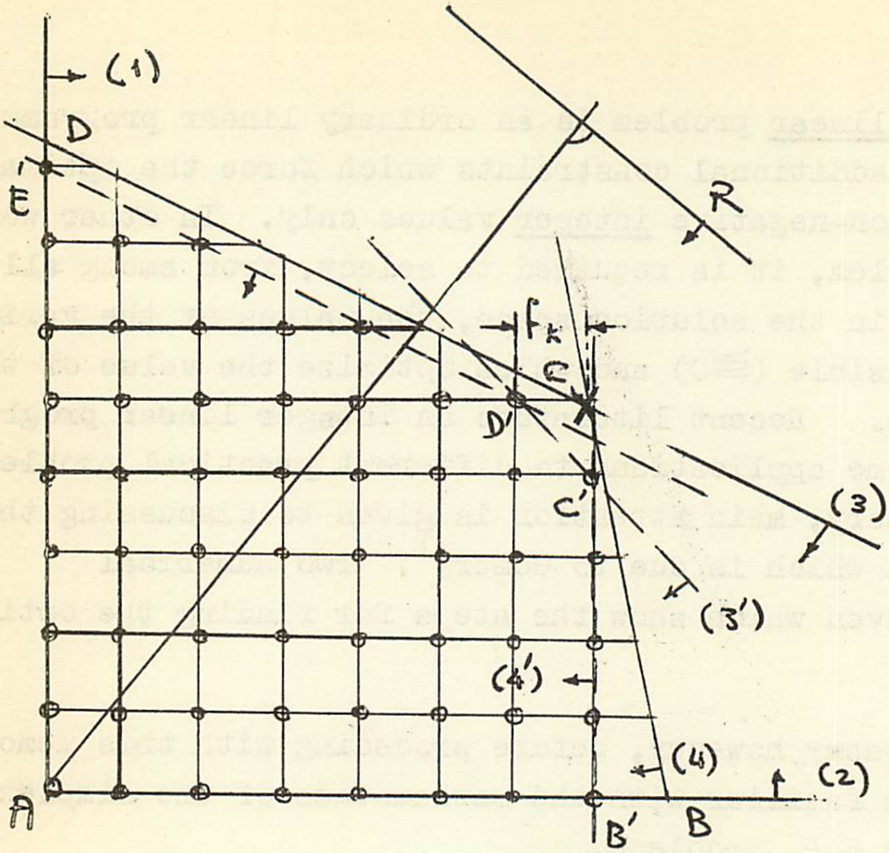

Figure (1)

Clearly, the optimal solution for this problem represented by point C, which is the point of intersection of constraints (3) and (4).

In order to obtain the optimal integer solution to this problem, we need first to define a new solution space consisting of all the integer values contained in the original convex hull ABCD. From Figure (1), the new integer solution space is given by the circled points which is defined by the new convex hull A B'C'D'E'.

Now the new problem becomes

maximize R

Subject to constraints(1), (2), (3'), (4'), and (5')

It is noted that the nex convex hull has integer values at its vertices and this assures that the new optimal, solution is all integers. This comes because, at every iteration in the solution of the L.P. problem, the basis solution is necessarily represented by a vertax on the convex hull.

It is clear from the above discussion that when the original constraints are substituted by new ones under the conditions given above, the problem can be solved as an ordinary L.P. problem using the simplex procedure. The construction of these new constraints however, eases to be feasible for the cases where there are three or more variables since in the case of two variables the graphical representation of the problem makes it possible to determine the new solution space. We thus need a method which automatically generates these new constraints (or some other constraints that would cause the same effect). This method will now be presented based on Gomory's model.[2]

3. Gomory's Integer Programming Model:

Basically the Gomory model can be summarized in the following steps:

1. Solve the ordinary L.P. under ordinary constraints
2. If the optimal solution contains all integer values, then there is nothing more to be done; otherwise,
3. Add new constraints which will force the final optimal solution into a new optimum with all integer values.
4. Repeat step (3) until integer optimal solution is obtained.

Clearly the addition of new constraints will enlarge the L.P. problem and in the same time will cut down the value of the objective function.

---

2. Ibid.

We now proceed to present the mathematics of the model:

Suppose that the set $\left\{X_i\right\}_{i=1}^{i=m}$ represents the basic variables at the last iteration of the simplex solution. Let the set $\left\{t_i\right\}_{i=1}^{i=m}$ represents the non-basic variables at the same iteration. We may thus write the long form representation of the basic variables as they appear in the constraints at the iteration as follows:

$$\left.\begin{aligned} X_k &= a_k + a_{k1}(-t_1) + a_{k2}(-t_2) + \dots + a_{kn}(-t_n) \\ &= a_k + \sum_{j=1}^{n} a_{kj}(-t_j) \end{aligned}\right\} k=1,2,\dots,m \quad (1)$$

where a_{kj} and a_k represent the constant parameters of the constraints as defined above.

Obviously, $X_k = a_k$, for all k
$t_j = 0$, for all j

which follows from the properties of the simplex solution. Namely at any iteration, the non-basic variables assume zero-values.

At the final iteration of the ordinary simplex procedure the final solution will be all integer if a_k, for all k, are integers. Otherwise, some or all a_k are non-integers. The Gomory model is thus used to force these non-integer values into new integer values while assuring optimal conditions.

Define $N_k = [a_k] = a_k - f_k$

$N_{kj} = [a_{kj}] = a_{kj} - f_{kj}$

where n_k and n_{kj} are the largest integers satisfying $n_k \leq a_k$ and $n_{kj} \leq a_{kj}$ respectively. From this definition it is clear that the factional values f_k and f_{kj} are always non-negative.[3]

---

3. See Figure(1) for graphical interpretation of f_k

Example:

a	n	f
5½	5	½
5	5	0
-1	-1	0
-½	-1	½

The idea now is to introduce some mathematical constrints (called Gomory constraints) which will cert off the optimal solution from point C to the new point D (Figure 1). In so doing the constraints must not cause any violation of the original constraints. Gomory defined constraints as

$$\sum_{j=1}^{n} f_{kj}\, t_j \geqq f_k \qquad (2)$$

We shall now show that:

1. Gomory constraints do not violate the original constrints of the problem.
2. They cut off the optimal solution from point C to point D thus giving an integer optimal solution.

Step 1. To show that Gomory constraints do not violate the original constraints of the problem:

Substituting for a_k and a_{kj} in equation (1) above, hence,

$$X_k = (n_k + f_k) + \sum_{j=1}^{n} (N_{kj} + F_{kj})(-t_j) \qquad (3)$$

Since X_k is ≥ 0, hence (3) can be written as:

$$\sum_{j=1}^{n} (N_{kj} + f_{kj}) \quad t_j \leq N_k + f_k$$

A necessary and sufficient condition that constraint (3) is not violated when f_k is "shaved" off from the solution is that:

$$\sum_{j=1}^{n} f_{kj} \quad t_j \geq f_k$$

(Note that f_k and f_{kj} are ≥ 0 by definition).

Instead of proving the above mathematical condition, we rather give an economic arguments (due to Baumol[4]) which will make it clear. Suppose that the original constraint is a warehouse space constraint so that a_k is the total number of cubic feet of the available ware house space. The constraint then says that the sum of the volumes of the different items stored in the ware house cannot exceed its total capacity. Now if we want to shave off a volume f_k from the ware house capacity, then the different items stored must be reduced by at least this volume. If follows then that if we want f_{kj} t_j to be shied off item j, then constraint (2) above must be satisfied.

Step 2. To show that Gomory constraints yield an Optimal integer solution.

Gomory constraint can now be written as

$$S_1 = -f_k - \sum_{j=1}^{n} f_{kj} \, (-t_j) \tag{4}$$

---

4. W.J. Baumal, Economic Theory and Operations Analysis, (Engle wood Chiffs: Prentice-Hall, 1961), P. 124.

where S_1 is a slack variable which is an integer ≥ 0 (later we prove the assertion that S_1 is necessarily an integer). The Gomary constraint is usually constructed from the final tableau of the simplex solution. This constraint is then added to this final tableau in which case the optimal solution just obtained by the simplex method, although optimal, it is not feasible. The idea then is carry out the simplex computations again until a feasible optimal solution is obtained.

(a) To show that S_1 should be integer when the solution of the problem is integer.

Let x_i^* and t_i^* be any integer solution hence (2) becomes:

$$S_1 = -f_k - \sum_{j=1}^{n} f_{kj}\,(-t_i^*)$$

$$= N_k - a_k - \sum_{j=1}^{n} (N_{kj} - a_{kj})\,(-t_j^*)$$

$$= N_k + \sum_{j=1}^{n} n_{kj}\,(-t_j^*) - a_k - \sum_{j=1}^{n} a_{kj}(-t_j)$$

$$= N_k + \sum_{j=1}^{n} n_{kj}\,(-t_j^*) - X_k^*$$

Since $X_k^* = a_k + \sum_{j=1}^{n} a_{kj}\,(-t_j^*)$, it is clear that all the components of the R.H.S. of S_1 are integers. If then follows that S_1 must be integer for an all-integer solution.

b) <u>To show that S_1 cuts out the old solution:</u>

Since in the old solution all $t_j = 0$, and since $f_k \geq 0$, hence the new slack variable S_1 (from (3) above) must be ≤ 0. Now $S_1 \leq 0$ violates the simplex procedure and $f_k \leq 0$ cannot hold by definition. This means that the old optimal solution must be cut down until the feasibility constraint is satisfied. It is noted that the feasibility condition is satisfied when $f_k = 0$, i.e., when the old solution becomes all integer. It is somuwhat difficult to show in mathematical terms how this "rounding" of factinal parts takes place. Rather we will use two numerical examples to make this point clear.

4. <u>Examples</u>:

Two numerical examples are solved in this section. The first example has two variables only so that we can represent it graphically. The second example consists of three variables so that we can show more iterations.

<u>Example 1</u>:

$$\begin{aligned} \text{maximize} \quad & Z = 2x + 6y \\ \text{Subject to} \quad & 3x + y \leq 5 \\ & 4x + 4y \leq 9 \\ & x, y \geq 0 \text{ and integers.} \end{aligned}$$

This problem can be put in table form as follows (V,W are slacks)[5]

---

5. We use the method of solution introduced by H.M. Wagner, "The Simplex method for Beyinners," <u>Operations Research</u>, VI (1958), pp. 190-199. A summary of this method is found in, H. Taha, Notes on Operations Research - part I, Institute of national Planning, 1965.

z	x	y	v	w	current solution
1	-2	-6	0	0	0
0	3	1	1	0	5
0	4	4	0	1	9

First iterations:

y in, w out

z	x	y	v	w	current solution
1	4	0	0	6/4	27/2
0	2	0	1	-1/4	11/4
0	1	1	0	1/4	9/4

which optimal and feasible but not integral.

The second step is to add Gomary constraint which will cut down the factional parts optimally. The above solution gives.

$$y = 2\tfrac{1}{4}$$
$$v = 2\tfrac{3}{4}$$
$$x = 0$$
$$w = 0$$

As a rule we select f_k = maximum fraction in all the values obtained. Namely

$$f_k = \max \left[\tfrac{1}{4}, \tfrac{3}{4}, 0, 0 \right] = \tfrac{3}{4}$$

Hence f_k corresponds to the variable V. The equation corresponding to (1) above can be written as:

$$v = 2\tfrac{3}{4} + \left[2(-x) + (-\tfrac{1}{4})(-w) \right]$$

Or in terms of equation (3),

$$v = (2 + 3/4) + \left[2(-x) + (-1 + 3/4)(-w) \right]$$

Hence $f_{k1} = 0$, $f_{k2} = 3/4$, and the resulting Gomory constraint becomes:

$$S_1 = -3/4 - 3/4(-w)$$
$$= -3/4 + 3/4 w$$

Adding this constraint to the previous tableau gives:

z	x	y	v	w	s_1	current solution
1	4	0	0	6/4	0	27/2
0	2	0	1	-1/4	0	11/4
0	1	1	0	1/4	0	9/4
0	0	0	0	-3/4	1	-3/4

at this iteration we use the Dual Simplex algorithm since some of the basic variables (namely S_1) is not feasible (=-3/4). Using this condition we obtain the following tableau.

<u>**Second iteration:**</u>

w in, S_1 out

z	x	y	v	w	s_1	Current solution
1	4	0	0	0	2	12
0	2	0	1	0	1/3	3
0	1	1	0	0	1/3	2
0	0	0	0	1	-4/3	1

which is now optimal feasible and all integers.

$$x = 0$$
$$y = 2$$
$$z = 12$$

(See the graphical solution of the problem).

Solution Space for integer solution =
{(0,0), (0,1), (1,0) (1,1), (0,2)}

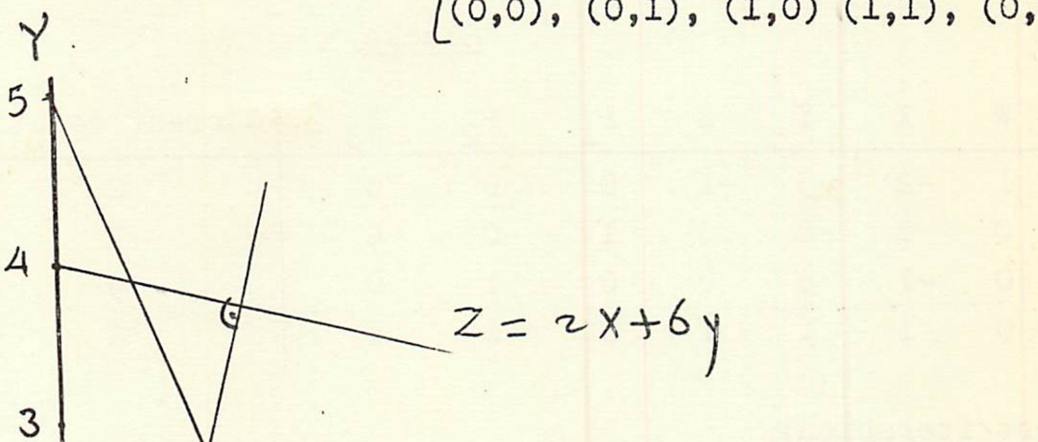

Figure 2 Graphical solution of Example I

Example 2:

maximize $W = 2x + 3y + Z$

Subject to

$$8x - 8y \leq 7$$
$$-x + 6y \leq 9$$
$$x + y + Z \leq 6$$
$$x, y, Z \geq 0 \text{ and integers.}$$

The above problem can be put in the following tableau form:

Slacks

W	X	Y	Z	P_1	P_2	P_3	Current solution
1	-2	-3	-1	0	0	0	0
0	8	-8	0	1	0	0	7
0	-1	6	0	0	1	0	9
0	1	1	1	0	0	1	6

First iteration:

y in , P_2 out

W	X	Y	Z	P_1	P_2	P_3	Current Solution
1	$-\frac{5}{2}$	0	-1	0	$\frac{1}{2}$	0	9/2
0	$\frac{20}{3}$	0	0	1	$\frac{8}{6}$	0	19
0	$-\frac{1}{6}$	1	0	0	$\frac{1}{6}$	0	$\frac{3}{2}$
0	$\frac{7}{6}$	0	1	0	$-\frac{1}{6}$	1	9/2

Second iteration

X in , P_1 out

W	X	Y	Z	P_1	P_2	P_3	Current Solution
1	0	0	-1	$\frac{3}{8}$	1	0	$\frac{93}{8}$
0	1	0	0	$\frac{3}{20}$	$\frac{1}{5}$	0	$\frac{57}{20}$
0	0	1	0	$\frac{1}{40}$	$\frac{1}{5}$	0	$\frac{79}{40}$
0	0	0	1	$-\frac{7}{40}$	$-\frac{2}{5}$	1	$\frac{47}{40}$

Third iteration

Z in, P_3 out

W	X	Y	Z	P_1	P_2	P_3	Current solution
1	0	0	0	$\frac{1}{5}$	$\frac{3}{5}$	1	$\frac{64}{5}$
0	1	0	0	$\frac{3}{20}$	$\frac{1}{5}$	0	$\frac{57}{20}$
0	0	1	0	$\frac{1}{40}$	$\frac{1}{5}$	0	$\frac{79}{40}$
0	0	0	1	$\frac{-7}{40}$	$\frac{-2}{5}$	1	$\frac{47}{40}$

which is the optimal non-integer solution

$$x = 2\ \frac{17}{20}$$

$$y = 1\ \frac{39}{40}$$

$$z = 1\ \frac{7}{40}$$

To obtain an optimal all-integer solution, we follow the same procedure as above.

$$f_k = \max\left[\frac{17}{20},\ \frac{39}{40},\ \frac{7}{40}\right] = \frac{39}{40}$$

This corresponds to y, hence expressing its equation as (3) above gives,

$$y = (1 + \frac{39}{40}) + \left[0(-x) + 0(-Z) + (\frac{1}{40})(-P_1) + \frac{1}{5}(-P_2) + 0(-P_3)\right]$$

$$= 1 + \frac{39}{40} + \left[(\frac{1}{40})(-P_1) + (\frac{1}{5})(-P_2)\right]$$

Hence $S_1 = -\frac{39}{40} + \frac{1}{40}P_1 + \frac{1}{5}P_2$

and the last tableau becomes:

W	X	Y	Z	P_1	P_2	P_3	S_1	Current solution
1	0	0	0	$\frac{1}{5}$	$\frac{3}{5}$	1	0	$\frac{64}{5}$
0	1	0	0	$\frac{3}{20}$	$\frac{1}{5}$	0	0	$\frac{57}{20}$
0	0	1	0	$\frac{1}{40}$	$\frac{1}{5}$	0	0	$\frac{79}{40}$
0	0	0	1	$-\frac{7}{40}$	$-\frac{2}{5}$	1	0	$\frac{47}{40}$
0	0	0	0	$-\frac{1}{40}$	$-\frac{1}{5}$	0	1	$-\frac{39}{40}$

Again using dual algorithm gives.

Fourth iteration

P_2 in, S_1 out

W	X	Y	Z	P_1	P_2	P_3	S_1	Current solution
1	0	0	0	$\frac{1}{8}$	0	1	3	$\frac{79}{8}$
0	1	0	0	$\frac{1}{8}$	0	0	1	$\frac{15}{8} = 1\frac{7}{8}$
0	0	1	0	0	0	0	1	1
0	0	0	1	$-\frac{1}{8}$	0	1	-2	$\frac{25}{8} = 3\frac{1}{8}$
0	0	0	0	$\frac{1}{8}$	1	0	-5	$\frac{39}{8} = 4\frac{7}{8}$

which is optimal but still non-integer.

$$f_k = \max\left[\frac{7}{8}, \frac{1}{8}, \frac{7}{8}\right] = \frac{7}{8}$$

In this case there is a two between X and P_2 but we select X since it involves less computations

$$X = (1 + \tfrac{7}{8}) + \left[(\tfrac{1}{8})(-P_1)\right]$$

Hence

$$S_2 = -\frac{7}{8} + \frac{1}{8} P_1$$

and the last tableau becomes:

W	X	Y	Z	P_1	P_2	P_3	S_1	S_2	Current solution
1	0	0	0	$\frac{1}{8}$	0	1	3	0	$\frac{79}{8}$
0	1	0	0	$\frac{1}{8}$	0	0	1	0	$\frac{15}{8}$
0	0	1	0	0	0	0	1	0	1
0	0	0	1	$-\frac{1}{8}$	0	1	-2	0	$\frac{25}{8}$
0	0	0	0	$\frac{1}{8}$	1	0	-5	0	$\frac{39}{8}$
0	0	0	0	$-\frac{1}{8}$	0	0	0	1	$-\frac{7}{8}$

__Fifth iteration:__

P_1 in, S_2 out

W	X	Y	Z	P_1	P_2	P_3	S_1	S_2	Current solution
1	0	0	0	0	0	1	3	1	9
0	1	0	0	0	0	0	1	1	1
0	0	1	0	0	0	0	1	0	1
0	0	0	1	0	0	1	-2	-1	4
0	0	0	0	0	1	0	-5	1	4
0	0	0	0	1	0	0	0	-8	7

which is now optimal, feasible and all-integer solution.

$x = 1$
$y = 1$
$z = 4$
$w = 9$