CHAPTER - 3.

3.1 Introduction to theoratical aspects of Goal Programming

3.1 Introduction To Theoratical Aspects Of GOAL PROGRAMMING :-

For any managerial science technique to be a truely valuable tool for decision analysis, it must accommodate itself to a computer-based solution. The complexity of real-world problems usually compels the use of computers. Many simple hypothetical problems being discussed and taught in classrooms exist only in textbooks. This by no means suggests that simple examples are of no value. Actually, they provide the foundation for understanding complex concepts of various management science techniques. Nevertheless, in order to apply a technique to practical problems which is indeed the very purpose of management science training, computer-based analysis is usually required. We have developed many powerful and mathematically sophisticated techniques- nonlinear programming, dynamic programming, game theory, etc. that have found a disappointingly limited scope for practical applications to real world problems. Modeling with such techniques is extremely hard for complex problems, and consequently a computer analysis is of little value.

In order for goal progrmming to be a useful management science technique for decision analysis, a computer-based solution is an essential requirement. Thus far, however, there has not been an efficient computer program for goal programming that has been widely circulated. As a matter of fact, this may well be one of the reasons for the limited application of goal programming in spite of its advantages. In an eagerness to apply goal programming, some reasearchers have used the conventional linear programming package by converting the preemptive priority factors to numerical values. Their studies may have produced some interesting results and their desire to apply goal programming to practical decision problems may have been satisfies. However, the use of linear programming by estimating the numerical values of multiple conflicting goals brings us right back to where we started. Conceptually, goal programming is based on ordinal solution on the basis of the priority of goals. (i.e. $P_i > > > P_i + 1$ Consequently, employing the linear programming package for a goal programming problem is not only defeating the very purpose of the goal programming approach but also violates the basic rule of goal programming. Furthermore, since the estimation of numerical values for multidmenstional goals can only be achieved through a fabication and distortion of the existing information, the linear programming solution will have only very limited value, if even that, to the decision maker. Quite possibly and justifiably the solution may not be accepted for implementation since it does not reflect the true decision environment.

Suppose we assign cerrtant than the second goal. Then, how can we interpret the model solution. Some may insist that as long as the computer program allows the solution according to the importance rank of goals it should be sufficient. That may be so. But quiet frequently, complex decision problems have all types of technological oceffi- cients. Consequently, it is often possible that low-order goals become more important than a higher order goal or the higher order goals become less important than the lower order goals in the solution process. The extreme cases are related to the sensitivity analysis and dual solution of goal programming problems based on the numerical objective function.

Goal programming is a special type of technique developed by Charnes and Cooper. This technique uses the simplex method for finding optimum solution of a single dimensional or multi-dimensional objective function with a given set of constraints which are expressed in linear form. In goal programming technique, all management goals, whether one or many, are incorporated into the objective function and only the environment al conditions i.e., those outside the management's control are treated as constraints. Moreover, each goal is set at a satisfying level which may not necessarily be the best obtainable, but one that management would be satisfied to achieve given multiple and sometimes conflicting tgoals. The computational procedure in goal programmign is to select a set of solutions which satisfies the environmental constraints and providing a satisfactory goal, ranked in priority order.

In goal programming, management is made to set some estimated targets for each of their goals and to rank them in the order of their priorities or importance. When this information is supplied, the goal programming tries to minimize the deviations from the targets that were set. It starts with the most important goal and continues until the achievement of a less important goal would cause the management to fail to achieve a more important one.

Goal programming problem, which employs the simplex method ofr finding optimum solution with a single goal with multiple subgoals, or multiple goals with multiple subgoals, given a system of constraints which are expressed as lineas equations. Goal here refers to the objective function. For example, in a production scheduling and employment smoothing problem, the management might consider the cost of shortages to be higher than costs of changing the employment level, and the latter costs to be higher than inventory costs, thus identifying three separate goals- the levels of production, employment and inventories. The hierarchy among these multiple goals may be set in such a way that those with lower priorities are considered only after higher priority goals are satisfied or have reached points beyond which they cannot be improved under the given conditions.

Let, Xj represents j sector based on sectoral route length

& m denotes the total number of sectors.

1) OPERATIONAL COST

Let, Cj denotes the Average Operational Cost/Bus/Km in sector j .

C denotes the total number of sectors.

Then,

$$\sum_{j=1}^{n} Cj\ Xj + d_1^- - d_1^+ = C.$$

As the objective is to minimize the operational cost, we have to reduce the positive deviational variable, d_1^+ ,as this is the first to be done we mention first priority, P1.

2) TRAVEL TIME

Let, Tj denotes the Average Travel Time/Bus/Km for sector j .

T denotes the total Travel Time.

Then,

$$\sum_{j=1}^{n} Tj\ Xj + d_2^- - d_2^+ = T.$$

As the objective is to minimize the travel time , we have to reduce the positive deviational variable , d_2^+ , with the next priority, P2.

3) REVENUE

Let, Rj denotes the Average Revenue/Bus/Km for sector j .

R denotes the total Revenue.

Then,

$$\sum_{j=1}^{n} Rj\ Xj + d_3^- - d_3^+ = R.$$

As the objective is to maximize the revenue, thuswe have to reduce the negative deviational variable , d_3^- , with the next priority, P3.

4) OCCUPANCY

Let, Oj denotes the Average Occupancy/Bus/Km.

O denotes the total Occupancy of the bus system.

Then,

$$\sum_{j=1}^{n} Oj\ Xj + d_4^- - d_4^+ = O.$$

As the objective is to maximize the occupancy, thus we have to reduce the negative deviational variable , d_4^- , with the next priority, P4.

5) AVAILABILITY

Let,

N denotes the fixed number of total Km. available for the bus system under consideration.

Then,

$$\sum_{j=1}^{n} Xj + d_5^- - d_5^+ = N.$$

Here, d_5^+ and d_5^- are positive and negative deviational variables respectively. The negative deviational variable reflect the under achievement while the positive deviational variable, over achievement from the targeted goals. Since we are trying to achieve all the targeted goals simultaneously as closely as possible we have to minimize both these deviational variables, d and with the next priority P5.

Thus, the objective function (Z) becomes --

Minimize, Z =

$$P1(d_1^+ + d_5^+) + P2\ d_2^+ + P3\ d_3^- + P4\ d_4^- + P5\ d_5^-$$

Subject to goal constraints,

$$\sum_{j=1}^{n} Cj\ Xj + d_1^- - d_1^+ = C.$$

$$\sum_{j=1}^{n} Tj\ Xj + d_2^- - d_2^+ = T.$$

$$\sum_{j=1}^{n} Rj\ Xj + d_3^- - d_3^+ = R.$$

$$\sum_{j=1}^{n} Oj\ Xj + d_4^- - d_4^- = O.$$

$$\sum_{j=1}^{n} Xj + d_5^- - d_5^+ = N.$$

After the formulation of the theoretical model, a computer software program in BASIC (Beginners All-purpose Symbolic Instruction Code) Language is developed to solve the problem.

The software consists of two programs --

i) for creating the data file i.e. the formulated problem and

ii) for solving the problem and to print the output on computer stationary.