CHAPTER-1
INTRODUCTION

1.1. INTEGER OPTIMIZATION

An integer optimization problem is a decision problem (with an objective to be maximized or minimized) in which the decision variables must take discrete values. An integer problem may be constrained or un-constrained; the objective function may be linear or non-linear. An integer problem is classified as linear if, by relaxing the integer restriction on the variables, the resulting functions are strictly linear. Otherwise, it is classified as non-linear integer programming problem. Many practical optimisation problems in non-negative integers, e.g. number of automobiles to be produced, number of men required to do a job etc, though at first sight, do not seem to fall under this category, can conveniently be formulated as integer programming problems.

1.2. MATHEMATICAL DEFINITION OF INTEGER PROGRAMMING PROBLEM

Integer programming is a branch of mathematical programming. A general mathematical programming problem can be stated as

maximise or minimise, \( z = f(x) \) \( \ldots (1) \)

subject to, \( g_i(x) \left\{ \begin{array}{c} \leq \ 1 \end{array} \right\} b_i, i = 1, \ldots, m, \ldots (2) \)
where \( \mathbf{x} \) is a column vector with components \( x_1, \ldots, x_n \),

\( f \) is a real valued function, called the objective function,

defined over the constraint set given by (2). In (2) one

and only one of the signs \( \leq, =, \geq \) holds for each constraint,

but the sign may vary from one constraint to another. Usually

some or all of the variables are required to satisfy the

non-negativity restrictions:

\[
\mathbf{x} \geq \mathbf{0} \quad \ldots \quad (3)
\]

In an integer linear programming problem (ILP) the

objective function is

\[
f(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j \quad \ldots \quad (4)
\]

and the constraint set is given by

\[
S = \left\{ \mathbf{x} \mid \sum_{j=1}^{n} a_{ij} x_j = b_i, \; i=1, \ldots, m, x_j \in \text{integers} \; j=1, \ldots, n \right\} \quad \ldots \quad (5)
\]

If all the components \( x_j, \; j=1, \ldots, n, \) of \( \mathbf{x} \) are restricted
to integer values, the problem is called a pure integer linear

problem. If some, but not all, of the variables are required
to be integer, we have a mixed integer linear programs (MILP).

Integer (or mixed integer) programs in which the integer
variables are constrained to be 0 or 1 are called 
\textit{zero-one integer (or mixed integer) programs}. It is 
interesting to note that every integer program in which 
the integer variables are bounded above can be written as 
zero-one integer problem. For if \( x_j \leq d_j \) (an \(\mathbb{Z}\) integer), 
then replace \( x_j \) by \( \sum_{j=1}^{d_j} u_{ij} \), where the \( u_{ij} \)'s are 0-1 
variables, and omit the \( x_j \leq d_j \) constraint.

1.3. METHODS FOR SOLVING GENERAL INTEGER PROGRAMS

In Dantzig (1951), it is shown that an integer linear
program in which the coefficient matrix \((a_{ij})\) has a uni-
modular property may be solved as a linear program. However,
most integer programs will not have this property and the
simplex method will generally not solve the integer programs.

The principal solution techniques for solving the integer
problems are categorized into two broad types (1) the cutting
plane techniques (2) search methods.

The cutting plane methods are developed primarily for
the integer linear programs. The method proposed by Dantzig,
Fulkerson and Johnson (1954) is motivated by the fact that
the simplex solution to linear program must occur at an
extreme point. The idea then is to add specially developed
constraints (cuts) that are violated by the current non-integer solution but never by any feasible integer point. The successive introduction of such cuts produces a solution space with its optimum extreme point properly satisfying the integrality conditions. The first cutting plane method was developed in 1958 by Gomory applicable to pure integer problems. Thun, Beal (1958) and Gomory (1960) developed the procedure for mixed integer linear problems. In 1960 Gomory obtained a new method for the pure integer problem which requires only addition and subtraction in computation called as all integer integer method. The above cutting plane methods differ mainly in the way the cut is constructed. The other types of cuts for solving both pure and mixed integer programs are introduced by Glover (1965), named as Bound-Escalation Cuts and by Young (1971), Balas (1971) and Glover (1973), named as "intersection" or "convexity" cuts.

The above methods during the successive steps maintain linear programs which are dual feasible. Glover (1968) and Young (1968) developed cutting plane methods that preserve primal feasible all integer tableaux. The cutting plane idea is also used by Guttmann (1965), for solving the integer problems in which the constraints are parabolic. Hanso and Osittli (1965) use the cutting plane idea for solving the quadratic integer programs. The procedure for mixed integer quadratic programs is developed by introducing the Gomory
The search methods are motivated by the fact that the integer solution space can be regarded as consisting of a finite number of points. The search methods are designed to enumerate implicitly or explicitly all such points. An exhaustive enumeration is difficult because the number of points in the integer solution space is generally very large. The search methods develop the techniques to enumerate only a (small) portion of all the candidate solutions while discarding the remaining points as non-promising. Search methods primarily include branch and bound techniques and implicit enumeration techniques. The second type may actually be considered as a special case of the branch and bound methods and is generally used for zero-one problems.

Land and Doig (1960) are the first to propose an enumerative procedure for MILP. This method was specialized for the travelling salesman problem by Little, Murty, Sweeney and Karel (1963). They named their specialized procedure as branch and bound method.

In the branch and bound procedure first one solves a continuous problem defined by relaxing the integer restrictions on the variables. If the optimal continuous solution is all integer, then it is also optimum for the
integer problem. Otherwise the following two operations are implemented successively. We assume that we are dealing with a maximization problem.

(1) Branching: The continuous solution space is partitioned into subspaces, which are also continuous. The purpose of partitioning is to eliminate parts of the continuous spaces that are not feasible for the integer problem. The partitioning should, however, be such that the collection of the partitioned subspaces includes every feasible integer point of the original problem.

(2) Bounding: The optimal objective value for each subproblem created by branching, sets an upper bound on the objective value associated with any of its integer feasible values. This bound is needed for ranking the optimal solutions of the subsets and hence locating the optimum integer solution.

When the solution of a subproblem is integer, the corresponding node is not branched; otherwise, further branching is necessary. The optimum integer solution is found when the subproblem having the largest upper bound amongst all subproblems yields an integer solution.

The original Land and Doig algorithm adjoins the equality constraint for creating the subproblems. This has a drawback
that a large number of branches may originate from the same node and this number normally can not be predicted in advance. An improved branching rule which creates exactly two branches from each eligible node was given by Dakin (1965). If at a node the optimal continuous solution has \( x_k = t \), where \( t \) is not integral, then the first node is created by introducing the inequality \( x_k < \lfloor t \rfloor \) and the second is created by introducing \( x_k > \lfloor t \rfloor + 1 \), where \( \lfloor t \rfloor \) is the largest integer less than or equal to \( t \). This is in contrast to the rule of creating nodes in Land and Doig's method by setting \( x_k = \lfloor t \rfloor \) and \( x_k = \lfloor t \rfloor + 1 \) and then defining other nodes to the left or right of these.

We have used the Land and Doig's approach in chapter two for solving a convex programming problem which arises in multivariate stratified sampling. This approach is used as it is convenient to solve a subproblem with the equality constraints. The Dakin's approach is used in chapter six for solving a quadratic integer programming problem.

The first work in the area of implicit enumeration has been reported by Balas (1965) for solving the zero-one integer programs. The technique is referred to as the additive algorithm. This work was elaborated on by Glover (1965), by introducing the use of surrogate constraints, whose work was the basis for other developments by Geoffrion (1967,69) and Balas (1967).
The Balas additive algorithm is generalised to solve the zero-one quadratic integer program by Ginsburg and Van Peetersen (1969).

Among other approaches for integer programming is a method for zero-one integer programming problems given by Kirby, Love and Sawarup (1972). They convert the zero-one integer problem into an equivalent non linear programming problem whose objective function is convex quadratic and the constraints are linear.

In 1965 Gomory showed that any integer program can be represented, by releasing the non-negativity constraint on certain variables, as a minimisation problem defined on a group. If this group problem is solved and its solution yields non-negative values for the variables of the original problem then the integer program is solved. Later it was shown by Gomory (1969) that the group problem can be treated as an integer program with one constraint. Algorithm designed to solve the integer programming formulation are of the dynamic programming type. The group theoretic approach has been studied further by Baxter (1968), Glover (1969), Shapiro (1968,1968), Garzy and Shapiro (1971).

A detailed survey of search methods is given by Balinski (1965), Lawler and Wood (1966) and Salkin (1973).
1.4. METHODS FOR SPECIAL INTEGER PROGRAMMING PROBLEMS

A problem of finding, from among a finite set of alternatives, one that optimizes the value of an objective function is called a combinatorial optimization problem. A large number of real life problems such as sequencing, routing and scheduling problems are the combinatorial optimization problems. One can always devise a nontrivial integer programming representation of a combinatorial optimization problem. Frequently such integer programming problems are very large in size and the integer programming formulation is of little computational interest. An advantage is taken of the special structures of such problems for developing the special purpose algorithms for them.

An important class of problems which can be formulated as integer programming problem are the problems of optimally allocating the resources to fulfill the requirements. For example, in the classical transportation problem the unimodular property of the coefficient matrix is exploited to develop the solution procedure (Dantzig (1951)). B. Khumawala (1974) has developed the heuristic approaches for the warehouse location problem which is also an MILP with specialized structure. In chapter two we treat the problem of allocating the sample number to different strata in multivariate surveys (Khan and Khan (1967)). First we develop a procedure for obtaining a non-integral solution. The integer solution is then obtained by applying the branch and bound
method (Khan and Bari (1977)). The structure of the problem suggests to use the Land and Doig's rule for branching.

The above allocation problem, when cost is fixed, is formulated as a convex programming problem with multiple objectives in chapter three. The various approaches for the non-linear programming problems with multiple objectives are discussed in Roy (1971). We generalise the STEP method of Banyaeun et al (1971) given for linear objective functions to our problem. A heuristic approach is used for finding the integer solution.

A large variety of integer programming formulations have a single constraint, e.g., capital budgeting (Lorie and Savage (1955)), Weingartner (1965), cutting stock problem (Gilmore and Gomory (1961, 1965, 1966)) and the capital investment problem as explained by Hansmann (1961). Such problems are classified as a Knapsack problem. The solution procedures are the dynamic programming techniques (Bartisig (1957)) and enumerative techniques (Kolesar (1967), Greenberg and Hagerich (1970)). The computational results with Knapsack algorithms are in Gilmore and Gomory (1963) and Kolesar (1967). In chapter four we formulate the problem of cutting the sheets of varying sizes into the pieces of specified sizes as an integer linear programming problem and describe a procedure
which makes use of the solution of some Knapsack problems.

In the above problem of cutting the sheets, the demands for the pieces of various sizes may not be fixed; but a discrete probability distribution may be available. This has been discussed in chapter five. The problem has been given the formulation of a two stage programming problem under uncertainty Madansky (1962). A solution procedure is presented by exploiting the special structure of the problem in which there is no optimisation problem in the second stage as the first stage variables and the values of the random elements uniquely determine the second stage variables.

In chapter six we discuss the integer quadratic programming problem. We apply the branch and bound method by using the Dakin's approach for branching. The created subproblems are the linear programs whose tableau differ slightly from the tableau of the initial continuous problem.

The numerical examples are also supplemented with the solution procedures. The references are given in the end of the thesis.