Mixed Integer Programming

Introduction

In the previous lecture we have discussed the procedure for solving integer linear programming in which all the decision variables are restricted to take only integer values. In a Mixed Integer Programming (MIP) model, some of the variables are real valued and some are integer valued. When the objective function and constraints for a MIP are all linear, then it is a Mixed Integer Linear Program (MILP). Although Mixed Integer Nonlinear Programs (MINLP) also exists, in this chapter we will be dealing with MILP only.

Mixed Integer Programming

In mixed integer programming (MIP), only some of the decision and slack variables are restricted to take integer values. Gomory cutting plane method is used popularly to solve MIP and the solution procedure is similar in many aspects to that of all-integer programming. The first step is to solve the MIP problem as an ordinary LP problem neglecting the integer restrictions. The procedure ends if the values of the basic variables which are constrained to take only integer values happen to be integers in this optimal solution. If not, a Gomory constraint is developed for the basic variable with integer restriction and has largest fractional value which is explained below. The rest of the procedure is same as that of all-integer programming.

Generation of Gomory Constraints:

Let \( x_i \) be the basic variable which has integer restriction and also has largest fractional value in the optimal solution of ordinary LP problem. Then from the \( i^{th} \) equation of table,

\[
x_i = b_i - \sum_{j=1}^{m} c_{ij} y_j
\]

As explained in all-integer programming, express \( b_i \) as an integer value plus a fractional part.

\[
b_i = \bar{b}_i + \beta_i
\]

\( c_{ij} \) can be expressed as
\[ c_{ij} = \tilde{c}_{ij}^+ + \tilde{c}_{ij}^- \]

where

\[
\tilde{c}_{ij}^+ = \begin{cases} 
  c_{ij} & \text{if } c_{ij} \geq 0 \\
  0 & \text{if } c_{ij} < 0
\end{cases}
\]
\[
\tilde{c}_{ij}^- = \begin{cases} 
  0 & \text{if } c_{ij} \geq 0 \\
  c_{ij} & \text{if } c_{ij} < 0
\end{cases}
\]

Therefore,

\[
\sum_{j=1}^{m} \left( \tilde{c}_{ij}^+ + \tilde{c}_{ij}^- \right) y_j = \beta_i + (\bar{h}_i - x_i)
\]

Since \( x_i \) is restricted to be an integer, \( \bar{h}_i \) is also an integer and \( 0 < \beta_i < 1 \), the value of \( \beta_i + (\bar{h}_i - x_i) \) can be \( \geq 0 \) or \( < 0 \).

Case I: \( \beta_i + (\bar{h}_i - x_i) \geq 0 \)

For \( x_i \) to be an integer, \( \beta_i + (\bar{h}_i - x_i) = \beta_i \) or \( \beta_i + 1 \) or \( \beta_i + 2, \ldots \).

Therefore,

\[
\sum_{j=1}^{m} \left( \tilde{c}_{ij}^+ + \tilde{c}_{ij}^- \right) y_j \geq \beta_i
\]

Finally it takes the form

\[
\sum_{j=1}^{m} \tilde{c}_{ij}^+ y_j \geq \beta_i
\]

Case II: \( \beta_i + (\bar{h}_i - x_i) < 0 \)

For \( x_i \) to be an integer,

\[ \beta_i + (\bar{h}_i - x_i) = -1 + \beta_i \text{ or } -2 + \beta_i \text{ or } -3 + \beta_i, \ldots \]

Therefore,

\[
\sum_{j=1}^{m} \left( \tilde{c}_{ij}^+ + \tilde{c}_{ij}^- \right) y_j \leq \beta_i - 1
\]
Finally it takes the form

\[ \sum_{j=1}^{m} \bar{c}_{ij} y_j \leq \beta_i - 1 \]

Dividing this inequality by \((\beta_i - 1)\) and multiplying with \(\beta_i\), we have

\[ \frac{\beta_i}{\beta_i - 1} \sum_{j=1}^{m} \bar{c}_{ij} y_j \geq \beta_i \]

Considering both cases I and II, the final form of the inequality becomes (since one of the inequalities should be satisfied)

\[ \sum_{j=1}^{m} \bar{c}_{ij} y_j + \frac{\beta_i}{\beta_i - 1} \sum_{j=1}^{m} \bar{c}_{ij} y_j \geq \beta_i \]

Then, the Gomory constraint after introducing a slack variable \(s_i\) is

\[ s_i - \sum_{j=1}^{m} \bar{c}_{ij} y_j - \frac{\beta_i}{\beta_i - 1} \sum_{j=1}^{m} \bar{c}_{ij} y_j = -\beta_i \]

Generate the Gomory constraint for the variables having integer restrictions. Insert this constraint as the last row of the final tableau of LP problem and solve this using dual simplex method. MIP techniques are useful for solving pure-binary problems and any combination of real, integer and binary problems.