\[ x_1 + x_2 = 4 \]

\[ (x_1, x_2) = \left( \frac{8}{3}, \frac{2}{3} \right); \quad z = \frac{99}{3} \]

\[ x_1 + 4x_2 = 8 \]
Application of Branch-and-Bound Technique to Solve BIP Problems

Ex: Problem 12.5-1; pg. 551
(Reference 1 by Hillier & Lieberman)

Maximize
\[ z = 2x_1 - x_2 + 5x_3 - 3x_4 + 4x_5 \]

s.t.: 
\[ 3x_1 - 2x_2 + 7x_3 - 5x_4 + 4x_5 \leq 6 \]
\[ x_1 - x_2 + 2x_3 - 4x_4 + 2x_5 \leq 0 \]
\[ x_j \text{ is binary, for } j = 1, 2, \ldots, 5 \]

The branch-and-bound technique is based on the philosophy of \underline{branching} (dividing) and \underline{fathoming} (conquering).

1. Branching

A large problem is divided into smaller subproblems until these subproblems can be fathomed. In doing so (i.e., branching), the entire set of feasible solutions is divided into smaller subsets.
2. **Fathoming**

To fathom a subset (subproblem), the B&B technique uses the bound. How good the best solution in a subproblem can be, is given by the bound. A subproblem is fathomed if its bound indicates that it cannot contain an optimal solution for the original problem.

**Steps Associated with the B&B Algorithm**

Evaluate the optimal solution for the LP-relaxation to the original problem, i.e., the problem as is, except the binary restrictions on variables are now converted to upper bounds.

$$0 \leq x_j \leq 1 \quad ; \quad j = 1, 2, 3, 4, 5$$

(See the attached output from LINDO).

The optimal solution is:

$$(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (0.67, 1, 1, 1, 1), \quad \text{and}$$

$$z^* = 6.33$$
Max \( 2x_1 - x_2 + 5x_3 - 3x_4 + 4x_5 \)

\[
\begin{align*}
st \\
3x_1 - 2x_2 + 7x_3 - 5x_4 + 4x_5 &< 6 \\
x_1 - x_2 + 2x_3 - 4x_4 + 2x_5 &< 0 \\
\end{align*}
\]

end

sub \( x_1 \) 1
sub \( x_2 \) 1
sub \( x_3 \) 1
sub \( x_4 \) 1
sub \( x_5 \) 1

\[\text{LP OPTIMUM FOUND AT STEP } \ 6\]

**OBJECTIVE FUNCTION VALUE**

1) \ 6.333333

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>VALUE</th>
<th>REDUCED COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_1 )</td>
<td>0.666667</td>
<td>0.000000</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>1.000000</td>
<td>-0.333333</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>1.000000</td>
<td>-0.333333</td>
</tr>
<tr>
<td>( x_4 )</td>
<td>1.000000</td>
<td>-0.333333</td>
</tr>
<tr>
<td>( x_5 )</td>
<td>1.000000</td>
<td>-1.333333</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROW</th>
<th>SLACK OR SURPLUS</th>
<th>DUAL PRICES</th>
</tr>
</thead>
<tbody>
<tr>
<td>2)</td>
<td>0.000000</td>
<td>0.666667</td>
</tr>
<tr>
<td>3)</td>
<td>0.333333</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

**NO. ITERATIONS= 6**

**OBJECTIVE FUNCTION VALUE**

1) \ 6.333333

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>VALUE</th>
<th>REDUCED COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_1 )</td>
<td>0.666667</td>
<td>0.000000</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>1.000000</td>
<td>-0.333333</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>1.000000</td>
<td>-0.333333</td>
</tr>
<tr>
<td>( x_4 )</td>
<td>1.000000</td>
<td>-0.333333</td>
</tr>
<tr>
<td>( x_5 )</td>
<td>1.000000</td>
<td>-1.333333</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROW</th>
<th>SLACK OR SURPLUS</th>
<th>DUAL PRICES</th>
</tr>
</thead>
<tbody>
<tr>
<td>2)</td>
<td>0.000000</td>
<td>0.666667</td>
</tr>
<tr>
<td>3)</td>
<td>0.333333</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

**NO. ITERATIONS= 6**
We are now ready to branch. Assume that the variables are ordered in the increasing order of their subscripts.

i.e., \( x_1, x_2, x_3, x_4, \text{ and } x_5 \).

Thus, the first branching variable is \( x_1 \), and the last branching variable is \( x_5 \).

Branching on \( x_1 \) will create two subproblems.

**Subproblem 1 (SP1) created by setting** \( x_1 = 0 \)

\[
\text{Max}^w \ z = -x_2 + 5x_3 - 3x_4 + 4x_5 \\
\text{s.t.:} \quad -2x_2 + 7x_3 - 5x_4 + 4x_5 \leq 6 \\
\quad -x_2 + 2x_3 - 4x_4 + 2x_5 \leq 0 \\
\quad x_j \text{ is binary for } j = 2, 3, 4, 5
\]

**SP2 (\( x_1 = 1 \))**

\[
\text{Max}^w \ z = 2 - x_2 + 5x_3 - 3x_4 + 4x_5 \\
\text{s.t.:} \quad -2x_2 + 7x_3 - 5x_4 + 4x_5 \leq 3 \\
\quad -x_2 + 2x_3 - 4x_4 + 2x_5 \leq -1 \\
\quad x_j \text{ is binary for } j = 2, 3, 4, 5
\]
The solution tree created by branching on $x_i$ is:

Variable: $x_i$

For each of the subproblems, we need to determine a bound on how good its best feasible solution can be.

An approach that can be used to obtain a bound is to solve the LP relaxation of the subproblem. Lagrangean relaxation is another approach which can be used to obtain a bound for the subproblem. Typically, Lagrangean relaxation would give a tighter bound than that obtained from LP relaxation.
⇒ **SP1**
Same as before but the binary restriction is now relaxed.

**SP2**
Same as before but the binary restriction is now relaxed.

(See attached outputs from LINDO)
The solution tree would appear as:

Notice that the bound for SP2 is 6 because the coefficients of the objective function and constraints are all integers.
Max  - x2 + 5x3 -3x4 +4x5
st
  - 2x2 + 7x3 - 5x4 + 4x5 < 6
  - x2 + 2x3 - 4x4 + 2x5 < 0
end
sub x2 1
sub x3 1
sub x4 1
sub x5 1

LP OPTIMUM FOUND AT STEP  5

OBJECTIVE FUNCTION VALUE
 1)  6.000000

VARIABLE     VALUE     REDUCED COST
X2     0.000000     0.250000
X3     1.000000     -3.500000
X4     1.000000     0.000000
X5     1.000000     -2.500000

ROW     SLACK OR SURPLUS     DUAL PRICES
2)     0.000000     0.000000
3)     0.000000     0.750000

NO. ITERATIONS=  5
Max - x2 + 5x3 -3x4 +4x5
st
- 2x2 + 7x3 - 5x4 + 4x5 < 3
- x2 + 2x3 - 4x4 + 2x5 < -1
end
sub x2 1
sub x3 1
sub x4 1
sub x5 1

LP OPTIMUM FOUND AT STEP 4
OBJECTIVE FUNCTION VALUE
1) 4.285714

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>VALUE</th>
<th>REDUCED COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>X2</td>
<td>1.000000</td>
<td>-0.428571</td>
</tr>
<tr>
<td>X3</td>
<td>0.857143</td>
<td>0.000000</td>
</tr>
<tr>
<td>X4</td>
<td>1.000000</td>
<td>-0.571429</td>
</tr>
<tr>
<td>X5</td>
<td>1.000000</td>
<td>-1.142857</td>
</tr>
</tbody>
</table>

ROW SLACK OR SURPLUS DUAL PRICES
2) 0.000000 0.714286
3) 0.285714 0.000000

NO. ITERATIONS= 4

Actual z = 2 + 4.285714 ≈ 6.29
As the LP-relaxation of SP1 has a unique optimum solution which satisfies the integrality requirements, it can be considered as the best feasible solution found so far.

\[ z^* = 6.0 \]

A subproblem can be fathomed if one of the following three tests hold true:

**Test 1**

Its bound \( \leq z^* \)

**Test 2**

Its LP-relaxation has no feasible solutions.

**Test 3**

The optimal solution for its LP-relaxation is integer.

\( \Rightarrow \) SP1 can be fathomed due to test 3.

SP2 can also be fathomed due to test 1.
It means that any subproblem created by branching from SP2 cannot have an objective function value greater than 6, or may have a value equal to 6.

⇒ We are done!

\[(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (0, 0, 1, 1, 1)\]

Let's proceed to see if we can fathom the subproblems generated from SP2.

**Rule for Branching**

From the list of remaining subproblems, select the one that was created most recently. If there is a tie, break ties in favor of the subproblem which has the larger bound.

(Note: An alternative rule is to select the subproblem that has the larger bound.)
In this case, there is only SP2, so we branch from SP2, using \( x_3 \) as the branching variable.

\[ x_3 = 0 \]

(Notice that \( x_1 = 1 \) previously)

Maximize \( \sum \) subject to:

\[
\begin{align*}
2 - x_2 - 3x_4 + 4x_5 & - 2x_2 - 5x_4 + 4x_5 & \leq 3 \\
-x_2 - 4x_4 + 2x_5 & \leq -1 \\
x_2 & \text{ is binary} \quad j = 2, 4, 5
\end{align*}
\]

For SP3 and SP4, we need to determine a bound by relaxing the integer requirement on variables (i.e., LP-relaxation)
This part of the solution tree would appear as:

\[
\begin{array}{c}
X_1 \\
X_2 \quad \text{(F/T1)} \\
\quad \text{SP3} \\
\quad \quad \text{z = 3.75; BOUND = 3} \\
\quad \quad (1, 0, 0, 0.75, 1) \\
\quad \text{SP4} \\
\quad \quad \text{z = 6.0; BOUND = 6} \\
\quad \quad (1, 1, 1, 0.75) \\
\end{array}
\]

SP3 is fathomed due to test 1 (i.e., its bound = 3 < z* = 6).

Branding from SP4, we have:

\[
\bar{SP5} \quad (x_5 = 0)
\]

(x_1 = 1 and x_3 = 1 previously)

Max \[ z = 7 - x_2 - 3x_4 \]

s.t.: \[-2x_2 - 5x_4 \leq -4 \]
\[-x_2 - 4x_4 \leq -3 \]

x_j is binary; j = 2, 4
\[ \text{max} \quad -x_2 - 3x_4 + 4x_5 \]
\[ \text{st} \]
\[-2x_2 - 5x_4 + 4x_5 < 3 \]
\[-x_2 - 4x_4 + 2x_5 < -1 \]
end

sub x2 1
sub x4 1
sub x5 1

LP OPTIMUM FOUND AT STEP 2

OBJECTIVE FUNCTION VALUE

1) 1.750000

VARIABLE     VALUE     REDUCED COST
X2     0.000000     0.250000
X4     0.750000     0.000000
X5     1.000000     -2.500000

ROW SLACK OR SURPLUS DUAL PRICES
2) 2.750000     0.000000
3) 0.000000     0.750000

NO. ITERATIONS = 2

OBJECTIVE FUNCTION VALUE

1) 1.750000

VARIABLE     VALUE     REDUCED COST
X2     0.000000     0.250000
X4     0.750000     0.000000
X5     1.000000     -2.500000

ROW SLACK OR SURPLUS DUAL PRICES
2) 2.750000     0.000000
3) 0.000000     0.750000

NO. ITERATIONS = 2

Actual \( z = 2 + 1.75 = 3.75 \)
max -x2 - 3x4 + 4x5
st
-2x2 - 5x4 + 4x5 <= -4
-x2 - 4x4 + 2x5 <= -3
end
sub x2 1
sub x4 1
sub x5 1

LP OPTIMUM FOUND AT STEP    4

OBJECTIVE FUNCTION VALUE
1)    -1.000000

VARIABLE VALUE REDUCED COST
   X2   1.000000   -1.000000
   X4   1.000000   -2.000000
   X5   0.750000    0.000000

ROW SLACK OR SURPLUS   DUAL PRICES
2)   0.000000    1.000000
3)   0.500000    0.000000

NO. ITERATIONS=    4

OBJECTIVE FUNCTION VALUE
1)    -1.000000

VARIABLE VALUE REDUCED COST
   X2   1.000000   -1.000000
   X4   1.000000   -2.000000
   X5   0.750000    0.000000

ROW SLACK OR SURPLUS   DUAL PRICES
2)   0.000000    1.000000
3)   0.500000    0.000000

NO. ITERATIONS=    4

Actual z = 7 - 1 = 6
SP6 \( (x_5 = 1) \)
\( (x_1 = 1 \text{ and } x_3 = 1 \text{ previously}) \)
\[
\text{Max} \quad z = 11 - x_2 - 3x_4
\]
\[
\text{s.t.:} \quad -2x_2 - 5x_4 \leq -8
\]
\[-x_2 - 4x_4 \leq -5
\]

\( x_j \) is binary \( ; \quad j = 2, 4 \)

(See attached outputs from LINDO for the LP-relaxations of SP5 and SP6)

This part of the solution tree would appear as:

SP5
\[
z = 4.67 ; \quad \text{BOUND} = 4
\]
\[
(1, 0.33, 1, 0.67, 0)
\]

SP4
\[
z = 6.0 ; \quad \text{BOUND} = 6
\]
\[
(1, 1, 1, 1, 0.75)
\]

SP6
\[
F/T_2
\]
\[
(1, ?, 1, ?, 1)
\]

SP5 is fathomed due to test 1
\( (\text{i.e., its bound} = 4 < z^* = 6) \)
\[ \text{max } -x_2 - 3x_4 \]
\[ \text{st } \]
\[-2x_2 - 5x_4 < -4 \]
\[-x_2 - 4x_4 < -3 \]
\[\text{end} \]
\[\text{sub } x_2 1 \]
\[\text{sub } x_4 1 \]

LP OPTIMUM FOUND AT STEP 3

OBJECTIVE FUNCTION VALUE

1) -2.333333

VARIABLE VALUE REDUCED COST
X2 0.333333 0.000000
X4 0.666667 0.000000

ROW SLACK OR SURPLUS DUAL PRICES
2) 0.000000 0.333333
3) 0.000000 0.333333

NO. ITERATIONS = 3

OBJECTIVE FUNCTION VALUE

1) -2.333333

VARIABLE VALUE REDUCED COST
X2 0.333333 0.000000
X4 0.666667 0.000000

ROW SLACK OR SURPLUS DUAL PRICES
2) 0.000000 0.333333
3) 0.000000 0.333333

NO. ITERATIONS = 3

Actual \[ z = 7 - 2 \cdot 3.3 = 4.67 \]
54. NO FEASIBLE SOLUTION AT STEP <N>.
There is no solution which simultaneously satisfies all the constraints. Run the DEBUG command in command-line versions and the Solve|Debug command on Windows versions to try to narrow down the problem.
SP6 is fathomed due to test 2 (i.e., no feasible solution exists).

Thus, the optimal solution for the original problem is:

\[(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (0, 0, 1, 1, 1),\]

and

\[z^* = 6\]

given by SP1.
A Branch-and-Bound Algorithm for Mixed-Integer Programming

We have previously considered a class of integer programming problems called the BIPs, where all the variables are restricted to either 0 or 1. Another class of problems is called the mixed-integer programming (MIP) problem, where some variables are restricted to integer values (instead of 0 or 1), while the rest are real variables.

A MIP problem can be formulated as:

$$\text{Max} \quad z = \sum_{j=1}^{n} a_{ij} x_j$$

subject to:

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \quad i = 1, 2, \ldots, m$$

and

$$x_j \geq 0 \quad j = 1, 2, \ldots, n$$

$$x_j \text{ is integer} \quad j = 1, 2, \ldots, I \quad (I \leq n)$$

The problem becomes a pure integer programming problem when $I = n$. 
Land and Doig (1960) were the first to develop a branch-and-bound algorithm for pure integer programming. Based on this development, Dakin (1965) subsequently developed a structure for solving mixed-integer programming problems. The steps associated with the algorithm are similar to that of the algorithm for BIP problem, with the exception of a few changes.
Ex: Problem 12.6-2; pg. 553
(Reference 1 by Hillier & Lieberman)

Minimize \( z = 2x_1 + 3x_2 \)

\[ \text{S.t.:} \quad x_1 + x_2 \geq 3 \\
\quad x_1 + 3x_2 \geq 6 \\
\quad x_1, x_2 \geq 0 \\
\quad x_1, x_2 \text{ are integers} \]

This is a pure IP problem. It is easy to see that the B&B algorithm for MIP problems can be slightly modified and applied to find the optimal solution for pure IP problems.

First we solve the LP-relaxation of the original problem. As this is a 2-variable problem, the LP-relaxation can be solved conveniently using a graphical approach.

Also, \( z^* = \infty \) for a minimization problem.
\[ Z = 2x_1 + 3x_2 = 12 \]
The optimal solution is obtained by solving simultaneous equations:

\[
\begin{align*}
    x_1 + x_2 &= 3 \\
    x_1 + 3x_2 &= 6
\end{align*}
\]

\[\Rightarrow x_1 = 1.5, \quad x_2 = 1.5\]

\[z = 2(1.5) + 3(1.5) = 7.5\]

Notice that because it is a pure IP problem, the bound for this ("all") node is equal to 8. Had it been a MIP problem, the bound would be 7.5.

The solution tree at this stage is:

\[\text{All} \]

Bound = 8

\( (1.5, 1.5) \)

**Branching**

For branching, select the subproblem that was created most recently. Ties are broken in favor of the problem that has the larger bound (for maximization). Use natural ordering of variables to select the branching variable. The variable is, therefore, selected as the one that is
required to be integer but currently holds a noninteger value.

**SP1**

\[
\text{Min} \quad z = 2x_1 + 3x_2 \\
\text{s.t.} \quad x_1 + x_2 \geq 3 \\
\quad x_1 + 3x_2 \geq 6 \\
\quad x_1 \leq 1 \\
\quad x_1, x_2 \geq 0, \text{ and } x_1, x_2 \text{ are integers}
\]

We could have stated this problem conveniently as:

\[\text{Original problem} \quad + \quad \begin{cases} x_1 \leq \lfloor x_1^* \rfloor \end{cases} \quad \Rightarrow \quad \text{Original problem} \quad + \quad x_1 \leq 1\]

\[\text{Original problem} \quad + \quad \begin{cases} x_1 \geq \lceil x_1^* \rceil + 1 \end{cases} \quad \Rightarrow \quad \text{Original problem} \quad + \quad x_1 \geq 2\]

**SP2**
\[ z = 2x_1 + 3x_2 = 12 \]

Optimal solution: \( (1, 2) \)
\[(x, \frac{1}{3})\]

Optimal solution

\[Z = 2x_1 + 3x_2 \leq P_2\]
Solutions

SP1

The optimal solution is obtained by solving the following equations:

\[ x_1 = 1 \]
\[ x_1 + x_2 = 3 \]

\[ \Rightarrow (x_1, x_2) = (1, 2) \]

\[ z = 2x_1 + 3x_2 = 2 + 3(2) = 8 \]

\[ \text{Bound} = 8 \]

\[ z^* = 8 \]

SP2

The optimal solution is obtained by solving the equations:

\[ x_1 = 2 \]
\[ x_1 + 3x_2 = 6 \]

\[ \Rightarrow (x_1, x_2) = (2, \frac{4}{3}) \]

\[ z = 2x_1 + 3x_2 = 2(2) + 3\left(\frac{4}{3}\right) = 8 \]

\[ \text{Bound} = 8 \]
Fathoming

There are 3 tests that can be performed to fathom a test problem (same as before!)

Test 1
For $\max^m$, Bound $\leq z^*$
For $\min^m$, Bound $\geq z^*$

Test 2
The LP-relaxation of the subproblem has no feasible solutions.

Test 3
The optimal solution for the LP-relaxation of the subproblem has integer values for integer-restricted variables.

If the $z$ value is better than $z^*$, $z^*$ must be revised, and the unfathomed subproblems must be compared with the new $z^*$ to determine if any of the previously created subproblems can be fathomed.
The solution tree at this stage is:

[Diagram]

SP1

$\chi_1 \leq 1$

Bound = 8

$\varepsilon^* = 8$

$(1, 2)$

SP2

$\chi_1 > 2$

Bound = 8

$(2, 4/3)$

SP1 is fathomed due to test 3

SP2 can also be fathomed due to test 1

i.e., Bound = 8 = $\varepsilon^* = 8$

Yet, let's proceed with generating the subproblems of SP2 to determine their solutions and the potential for fathoming them.

$\Rightarrow$

SP3

SP2

$+$

$\chi_2 \leq [4/3]$

$\Rightarrow$

$+$

$\chi_2 \leq 1$
SP4

SP2 + \{ \}
\Rightarrow
SP2 +
\begin{align*}
x_2 \geq \left\lceil \frac{4}{3} \right \rceil + 1
\end{align*}
\Rightarrow
x_2 \geq 2

\textbf{Solutions}

\textbf{SP3}

The optimal solution is obtained by solving the following equations:
\begin{align*}
x_2 &= 1 \\
x_1 + 3x_2 &= 6
\end{align*}
\Rightarrow (x_1, x_2) = (3, 1)
\begin{align*}
z &= 2x_1 + 3x_2 = 2(3) + 3(1) = 9
\end{align*}
Bound = 9

\textbf{SP4}

The optimal solution is obtained by solving the following equations:
\begin{align*}
x_1 &= 2, \text{ and } \\
x_2 &= 2
\end{align*}
\Rightarrow (x_1, x_2) = (2, 2)
\begin{align*}
z &= 2x_1 + 3x_2 = 2(2) + 3(2) = 10
\end{align*}
Bound = 10
\[ Z = 2x_1 + 3x_2 = 12 \]

**Optimal Solution:**
The solution tree at this stage is:

SP3

\[ x_2 \leq 1 \quad \text{E/T1} \]

\[ \text{Bound} = 9 \]

\( (3, 1) \)

SP2

\[ x_1 \geq 2 \]

\[ \text{Bound} = 8 \]

\( (2, \frac{4}{3}) \)

SP4

\[ x_2 \geq 2 \quad \text{E/T1} \]

\[ \text{Bound} = 10 \]

\( (2, 2) \)

SP3 is fathomed due to Test 1
i.e., \( \text{Bound} = 9 > z^* = 8 \)

SP4 is also fathomed due to Test 1
i.e., \( \text{Bound} = 10 > z^* = 8 \)

\[ \Rightarrow \text{The optimal solution for the original problem is given by SP1} \]

i.e., \( (x^*_1, x^*_2) = (2) \)

\[ z^* = 8 \]