The University of Jordan
Department of Mathematics

Cutting Plane Method

Prepared and presented by:

Nadjiba MANSOURI
Ilhem KADRI
Samia MAZOUZ
Ahmed BOUCHENAK

Soraya ABDEL AZIZ
Mohamed BEN RABIA
Said ATLAOUI
Ibtissem ATTAFI

Supervised and edited by: Dr. Baha ALZALG

Integer and Combinatorial Optimization (0301973)
Spring 2018
Sommaire

1. Introduction

2. The Cutting Plane Method
   - General Scheme
   - An Example (with the graphical method)

3. The Gomory Cutting Plane Method
   - The Algorithm Steps
   - Examples (with the simplex method)
   - How to avoid cycling?

4. Conclusion
An integer programming problem is a mathematical optimization or feasibility program in which some or all of the variables are restricted to be integers which is very difficult to solve. However, there are three different types of algorithms:

1. **Exact algorithms.**
2. **Approximation algorithms.**
3. **Heuristic algorithms.**
Exact Algorithms

Examples of exact algorithms designed and used to solve combinatorial optimization problems:

2. Branch and Bound Method.

In this work, we present a method called the Cutting Plane Method.
Let $P$ be a mixed integer linear program:

\[
\begin{aligned}
\min & \quad c^T x + d^T y + e^T z \\
\text{s.t.} & \quad Ax + By + Cz \leq b \\
& \quad x, y, z \geq 0 \\
& \quad y \in \mathbb{Z}.
\end{aligned}
\]  

\( (P) \)

The linear programming relaxation is:

\[
\begin{aligned}
\min & \quad c^T x + d^T y + e^T z \\
\text{s.t.} & \quad Ax + By + Cz \leq b \\
& \quad x, y \geq 0.
\end{aligned}
\]  

\( (P') \)

Problem \((P')\) is called linear relaxation of \((P)\).
General Scheme of the Cutting Plane Method

The integer programming problem

\[
\begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad Ax \leq b \\
& \quad x \geq 0 \text{ integer.}
\end{align*}
\]

The linear programming relaxation

\[
\begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad Ax \leq b \\
& \quad x \geq 0.
\end{align*}
\]
General Scheme of the Cutting Plane Method

The main idea of the **Cutting Plane Method** is to solve the integer programming problem by solving a sequence of linear programming problems, as follows:

1. Solve the linear programming relaxation.
2. If the resulting optimal solution $x^\star$ is integral, stop (*the optimal solution is found*).
3. If not, generate a cut by using the Gomory cutting plane algorithm, i.e., a constraint which is satisfied by all feasible integer solution but not by $x^\star$.
4. Add this new constraint, resolve problem, and go back to (2). This terminates after a finite number of iterations in (2) and the resulting $x^\star$ is integral and optimal.
Example 1

Consider the following problem:

\[
\begin{align*}
\text{max} & \quad x_0 = 5x_1 + 8x_2 \\
\text{s.t.} & \quad x_1 + x_2 \leq 6 \\
& \quad 5x_1 + 9x_2 \leq 45 \\
& \quad x_1, x_2 \geq 0 \\
& \quad x_1, x_2 \in \mathbb{Z}.
\end{align*}
\]
Step 1. Solve the IP problem with continuous variables instead of discrete ones.
Step 2. If the resulting optimal solution $x^*$ is integral, stop
\Rightarrow \text{ optimal solution found.}

Resulting solution is $x^* = (2.25, 3.75)$ and hence \textit{not} integral.
**Step 3.** Generate a *cut*, i.e., a constraint which is satisfied by all feasible integer solutions but not by \( x^* \).

We generate cut \( 2x_1 + 3x_2 \leq 15 \).

*(We will discuss later how to generate such cuts.)*
Step 4. Add this new constraint, resolve problem, and go back to (2).

New optimal solution is $x^* = (3, 3)$.
Note: Previous $x^*$ is not feasible anymore.
Remark. The cut we introduced removed *non-integral* solutions only. Cuts *never* cut off feasible solutions of the original IP problem!
The Gomory Cutting Plane Algorithm

The first finitely terminating algorithm for integer programming was a cutting plane algorithm proposed by Ralph Gomory in 1958 at IBM.
The Gomory Cutting Plane Algorithm

Step 1

We solve the standard form linear programming problem with the simplex method. Let $x^*$ be the optimal basic feasible solution.

We partition $x$ into two subvectors: $x = \begin{pmatrix} x_B \\ x_N \end{pmatrix}$, where $x_B$ is the subvector corresponding the basic variables, and $x_N$ is the subvector corresponding the nonbasic variables. Assume for simplicity that $x_B(i) = x_i, \forall i = 1, \ldots, m$ (i.e., the first $m$ variables are basic).
From the optimal tableau

\[ B^{-1}A\vec{x} = B^{-1}b, \ldots \ldots \ldots \ldots \ldots (\star) \]

where
- A is the coefficient matrix of the problem in standard form.
- B is the submatrix of A whose columns are \( a_i \) for \( i \) in the set of indices of basic variables.

From (\( \star \)) we obtain

\[ x_B + B^{-1}A_N\vec{x}_N = B^{-1}b \]

where N is the set of indices of nonbasic variables and \( A_N \) is the submatrix of A whose columns are \( a_i \) for \( i \in N \).

(Therefore, \( A = [B \ A_N] \)).
Step 2

We consider one equality from the optimal tableau

\[ x_i + \sum_{j \in N} a_{ij} x_j = a_{i0}, \]

where

\[ a_{ij} = (B^{-1}A_j)_i \quad \text{and} \quad a_{i0} = (B^{-1}b)_i \quad \text{and} \quad a_{i0} \text{ is fractional}, \]

where \([t]\) is floor of \(t\). Since \(x_j \geq 0\ \forall j\), we have

\[ x_i + \sum_{j \in N} [a_{ij}] x_j \leq a_{i0}, \quad \text{as} \quad [a_{ij}] \leq a_{ij}. \]

Since \(x_j\) must be integer, we obtain

\[ x_i + \sum_{j \in N} [a_{ij}] x_j \leq [a_{i0}]. \]

This inequality is valid for all integer sols, but is not satisfied by \(x^*\).
Example 2 (a minimization integer program)

Consider the integer programming problem

\[
\begin{align*}
\text{min} & \quad x_1 - 2x_2 \\
\text{s.t.} & \quad -4x_1 + 6x_2 \leq 9 \\
& \quad x_1 + x_2 \leq 4 \\
& \quad x_1, x_2 \geq 0 \text{ and both integers.}
\end{align*}
\]

The problem in the standard form becomes

\[
\begin{align*}
\text{min} & \quad x_1 - 2x_2 \\
\text{s.t.} & \quad -4x_1 + 6x_2 + x_3 = 9 \\
& \quad x_1 + x_2 + x_4 = 4 \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{align*}
\]
Solve the linear programming relaxation by using the simplex method: Note that

\[ A = \begin{pmatrix} -4 & 6 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}. \]

We partition \( x \) into two subvectors \( x = (x_B, x_N) \), where

\[ x_B = (x_1, x_2) \quad \text{(basic variables)}, \]

\[ x_N = (x_3, x_4) \quad \text{(nonbasic variables)}. \]

It follows that

\[ B = \begin{pmatrix} -4 & 6 \\ 1 & 1 \end{pmatrix} \quad \text{and} \quad A_N = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \]
The system $B^{-1}Ax = B^{-1}b$ can be written as

$$x_1 - \frac{1}{10}x_3 + \frac{3}{5}x_4 = \frac{15}{10}, \quad ........... \quad (1)$$

$$x_2 + \frac{1}{10}x_3 + \frac{2}{5}x_4 = \frac{25}{10}. \quad ........... \quad (2)$$

So the optimal solution is $x^1 = \left( \frac{15}{10}, \frac{25}{10} \right)$.

While the two equations have the same fractional part in the right-hand side values (which is 5/10), equation (2) has the biggest right-hand side value in this system.

As a result, choosing equation (2) and applying the Gomory cutting plane, we get

$$x_2 + \begin{bmatrix} 1/10 \\ 2/5 \end{bmatrix} x_3 + \begin{bmatrix} 25 \\ 10 \end{bmatrix} x_4 \leq \begin{bmatrix} 25 \\ 10 \end{bmatrix}. \quad \text{This leads to } x_2 \leq 2.$$
We augment the linear programming relaxation by adding the constraint $x_2 + x_5 = 2$ to obtain

$$\begin{align*}
\text{min} & \quad x_1 - 2x_2 \\
\text{s.t.} & \quad -4x_1 + 6x_2 + x_3 = 9 \\
& \quad x_1 + x_2 + x_4 = 4 \\
& \quad x_2 + x_5 = 2 \\
& \quad x_1, x_2, x_3, x_4, x_5 \geq 0.
\end{align*}$$

Applying the simplex method, we get
The Gomory Cutting Plane Method

### Examples (with the simplex method)

#### Iteration 1

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$c$</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_3$</td>
<td>-4</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>$\frac{3}{2}$</td>
</tr>
<tr>
<td>$x_4$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>$x_5$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$z$</td>
<td>1</td>
<td>-2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

#### Iteration 2

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$c$</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_2$</td>
<td>$\frac{-2}{3}$</td>
<td>1</td>
<td>$\frac{1}{6}$</td>
<td>0</td>
<td>0</td>
<td>$\frac{3}{2}$</td>
<td>-1</td>
</tr>
<tr>
<td>$x_4$</td>
<td>$\frac{5}{3}$</td>
<td>0</td>
<td>$\frac{-1}{6}$</td>
<td>1</td>
<td>0</td>
<td>$\frac{5}{2}$</td>
<td>$\frac{3}{2}$</td>
</tr>
<tr>
<td>$x_5$</td>
<td>$\frac{2}{3}$</td>
<td>0</td>
<td>$\frac{-1}{6}$</td>
<td>0</td>
<td>1</td>
<td>$\frac{1}{2}$</td>
<td>$\frac{3}{4}$</td>
</tr>
<tr>
<td>$z$</td>
<td>$\frac{-1}{3}$</td>
<td>0</td>
<td>$\frac{1}{3}$</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

#### Iteration 3

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$c$</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_2$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
<td>$\frac{3}{12}$</td>
<td>1</td>
<td>$\frac{-5}{2}$</td>
<td>$\frac{-5}{2}$</td>
<td></td>
</tr>
<tr>
<td>$x_1$</td>
<td>0</td>
<td>0</td>
<td>$\frac{-1}{4}$</td>
<td>0</td>
<td>$\frac{3}{2}$</td>
<td>$\frac{3}{2}$</td>
<td></td>
</tr>
<tr>
<td>$z$</td>
<td>0</td>
<td>0</td>
<td>$\frac{1}{2}$</td>
<td>0</td>
<td>1</td>
<td>$\frac{13}{2}$</td>
<td></td>
</tr>
</tbody>
</table>
So, the new optimal solution is $x^2 = \left( \frac{3}{4}, 2 \right)$, which is not integer. The equations in the optimal tableau are

$$
\begin{align*}
    x_1 - \frac{1}{4} x_3 + \frac{3}{2} x_5 &= \frac{3}{4}, \quad \cdots \cdots (3) \\
    x_2 + x_5 &= 2. \quad \cdots \cdots (4)
\end{align*}
$$

Choosing equation (3) (because it has the biggest fractional part among all the right-hand side values in the system ($3/4 > 0$)) and applying the Gomory cutting plane method, we get

$$
\begin{align*}
    x_1 + \left[ -\frac{1}{4} \right] x_3 + \left[ \frac{3}{2} \right] x_5 &\leq \left[ \frac{3}{4} \right]. \quad \text{This leads to } x_1 - x_3 + x_5 \leq 0.
\end{align*}
$$

Since $x_2 + x_5 = 2$, we have $x_5 = 2 - x_2$. In term of the original variables $x_1, x_2$, we get

$$
-3x_1 + 5x_2 \leq 7.
$$
We add this constraint, together with the previously added constraint.

Applying the simplex method, we get the new optimal solution $x^3 = (1, 2)$, which is integer! So, it is the optimal solution of the original problem.
Example 3 (a maximization integer program)

Consider the following integer programming problem

\[
\begin{align*}
\text{max} & \quad 3x_1 + 4x_2 \\
\text{s.t.} & \quad 2x_1 + 5x_2 \leq 15 \\
& \quad 2x_1 - 2x_2 \leq 5 \\
& \quad x_1, x_2 \geq 0 \text{ and both integers.}
\end{align*}
\]

The standard form is

\[
\begin{align*}
\text{max} & \quad 3x_1 + 4x_2 \\
\text{s.t.} & \quad 2x_1 + 5x_2 + x_3 = 15 \\
& \quad 2x_1 - 2x_2 + x_4 = 5 \\
& \quad x_1, x_2, x_3, x_4 \geq 0.
\end{align*}
\]
We solve the LP relaxation by using the simplex method. We have

\[ A = \begin{pmatrix} 2 & 5 & 1 & 0 \\ 2 & -2 & 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 5 \\ 2 & -2 \end{pmatrix}, \quad \text{and} \quad b = \begin{pmatrix} 15 \\ 5 \end{pmatrix}. \]

\[ B^{-1}Ax = B^{-1}b \iff x_1 + \frac{1}{7}x_3 + \frac{5}{14}x_4 = \frac{55}{14}, \]
\[ x_2 + \frac{1}{7}x_3 - \frac{1}{7}x_4 = \frac{10}{7}. \]

The optimal solution is

\[ x^1 = \left( \frac{55}{14}, \frac{10}{7} \right). \]
Choosing the first equation and applying the Gomory cutting plane method, we obtain

\[ x_1 + \left[ \frac{1}{7} \right] x_3 + \left[ \frac{5}{14} \right] x_4 \leq \left[ \frac{55}{14} \right], \text{ which leads to } x_1 \leq 3. \]

Introducing a new slack variable to get \( x_1 + x_5 = 3, x_5 \geq 0 \). Adding this cut to the linear problem and solving by the simplex method, we find the new optimal solution \( x^2 = (3, \frac{9}{5}) \), which is not integer.
We choose an equation from the optimal tableau, which is

\[ x_2 + \frac{1}{5}x_3 - \frac{2}{5}x_5 = \frac{9}{5}, \]

and apply the Gomory cutting plane method and find the cut

\[ x_1 + x_2 \leq 4 \text{ or } x_2 + x_1 + x_6 = 4, \ x_6 \geq 0. \]

Add this cut to the problem and go back to step 1. The new optimal solution is \( x^3 = \left( \frac{5}{3}, \frac{7}{3} \right) \), which is still not integer.
From the optimal tableau, we choose the equation

\[ x_2 + \frac{1}{3}x_3 - \frac{2}{3}x_6 = \frac{7}{3}, \]

and after applying the Gomory cutting plane method, we find the cut

\[ x_1 + 2x_2 \leq 6 \quad \text{or} \quad x_1 + 2x_2 + x_7 = 6, \quad x_7 \geq 0. \]

We add this cut to the problem and go back step 1. Finally, find the new optimal solution \( x^4 = (2, 2) \), which is integer! It is an optimal solution to the original problem.
A graph illustrates the three cuts generated in Example 3.
Another cut to avoid the cycling

We apply the following cut to avoid the cycling:

$$\sum_{j \in N} f_j x_j \geq f, \quad \cdots \cdots \cdots \ (**)$$

where

$$f_j = a_{ij} - \lfloor a_{ij} \rfloor \quad \text{and} \quad f = a_{i0} - \lfloor a_{i0} \rfloor.$$ 

This inequality is satisfied by all feasible $x$ but it is not satisfied by $x^*$. 
Example 4 (on avoiding cycling)

\[
\text{max} \quad z = 2x_1 + 3x_2 \\
\text{s.t.} \quad 2x_1 + 2x_2 \leq 7 \\
\quad x_1 \leq 2 \\
\quad x_2 \leq 2 \\
\quad x_1, x_2 \geq 0 \text{ and both integers.}
\]

The standard form is

\[
\text{max} \quad z = 2x_1 + 3x_2 \\
\text{s.t.} \quad 2x_1 + 2x_2 + x_3 = 7 \\
\quad x_1 + x_4 = 2 \\
\quad x_2 + x_5 = 2 \\
\quad x_1, x_2, x_3, x_4, x_5 \geq 0.
\]
After solving the relaxation by using the simplex method, the optimal tableau is

<table>
<thead>
<tr>
<th>$CB_i$</th>
<th>$c_j$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>Sol</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$x_1$</td>
<td>1</td>
<td>0</td>
<td>$\frac{1}{2}$</td>
<td>0</td>
<td>-1</td>
<td>$\frac{3}{2}$</td>
</tr>
<tr>
<td>0</td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
<td>$-\frac{1}{2}$</td>
<td>1</td>
<td>1</td>
<td>$\frac{1}{2}$</td>
</tr>
<tr>
<td>3</td>
<td>$x_2$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$z_j$</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>$c_j - z_j$</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The optimal solution is

$$x^1 = \left( \frac{3}{2}, 2 \right),$$

but it is not integer.
We choose the following equation from the optimal tableau

\[ x_1 + \frac{1}{2}x_3 - x_5 = \frac{3}{2}, \]

and apply the Gomory cutting plane method to obtain the cut

\[ x_1 + x_2 \leq 3. \]

If we add this constraint to the LPP and use the simplex method, we get the same cut in each sequence, which will lead to cycling!

<table>
<thead>
<tr>
<th>( CB_i )</th>
<th>( c_j )</th>
<th>2</th>
<th>3</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>Sol</th>
</tr>
</thead>
<tbody>
<tr>
<td>basic variable</td>
<td>( x_1 )</td>
<td>1</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>( \frac{3}{2} )</td>
</tr>
<tr>
<td>( x_4 )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>( \frac{-1}{2} )</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>( x_6 )</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>( z_j )</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>( c_j - z_j )</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
So, we use the cut (**) in our example for solve the ILP. The new cut is
\[
\frac{1}{2} x_3 \geq \frac{1}{2} \text{ or } -\frac{1}{2} x_3 \leq -\frac{1}{2} ,
\]
which is equivalent to
\[
-\frac{1}{2} x_3 + x_6 = -\frac{1}{2} , \quad x_6 \geq 0.
\]

Now, we use the dual simplex method to solve the resulting problem.
## Iteration 1

<table>
<thead>
<tr>
<th>$CB_i$</th>
<th>basic variable</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>Sol</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$x_1$</td>
<td>1</td>
<td>0</td>
<td>$\frac{1}{2}$</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>$\frac{3}{2}$</td>
</tr>
<tr>
<td>0</td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
<td>$\frac{-1}{2}$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>$\frac{1}{2}$</td>
</tr>
<tr>
<td>3</td>
<td>$x_2$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>$x_6$</td>
<td>0</td>
<td>0</td>
<td>$\frac{-1}{2}$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>$\frac{-1}{2}$</td>
</tr>
<tr>
<td>$z_j$</td>
<td></td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>$c_j - z_j$</td>
<td></td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
## Iteration 2

<table>
<thead>
<tr>
<th>$c_j$</th>
<th>2</th>
<th>3</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$CB_i$</td>
<td>basic variable</td>
<td>$x_1$</td>
<td>$x_2$</td>
<td>$x_3$</td>
<td>$x_4$</td>
<td>$x_5$</td>
</tr>
<tr>
<td>2</td>
<td>$x_1$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>0</td>
<td>$x_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>$x_2$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>$x_6$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$z_j$</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$c_j - z_j$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>-2</td>
</tr>
</tbody>
</table>

The tableau is now optimal and the optimal solution is $x = (1, 2)$, which is integer!
The Cutting Plane Method is very useful for solving integer programming problems, but there is a difficulty lies in the choice of inequalities which represent the cut of only a very small piece of the feasible set of the linear programming relaxation.