Branch&Cut

1 Readings:

[1]: Mitchell J.E. Integer programming: Branch-and-cut algorithms. Encyclopedia of Optimization, Volume II, Kluwer Academic Press, August 2001.

[1]: For a review of Gomory's cutting plane method, consult your msci 332 notes or Winston W.L. and M. Venkataramanan. *Introduction to mathematical programming*, Duxbury Press, 2003. page 545 to 549 for cutting plane methods; pages 329-334 for the dual simplex.

2 Branch-and-Cut:Combining Branch-and-Bound and Cutting Plane Methods

Consider the following example based on [2].

$$\begin{array}{rl} \min & -5x_1 - 6x_2 \\ \text{s.t.} & x_1 + 2x_2 \leq 7 \\ & 2x_1 - x_2 \leq 3 \\ & x_1, x_2 \geq 0, \text{integer.} \end{array}$$

that we want to solve through branch-and-cut. We start by solving the LP relaxation at Node 0 using the Simplex method. We start by putting the LP in standard format:

$$\begin{array}{rlll} \min & -5x_1 - 6x_2 & & \\ \text{s.t.} & x_1 + 2x_2 + s_1 & & = 7 \\ & 2x_1 - x_2 & +s_2 & = 3 \\ & x_1, x_2 \geq 0, & & \end{array}$$

and then in tableau format:

		↓	↑			
z	x_1	x_2	s_1	s_2	rhs	
1	-5	-6	0	0	0	
0	1	2	1	0	7	ratio test: 7/2=3.5 ← pivot row
0	2	-1	0	1	3	

The second tableau/iteration is:

	↓			↑		
z	x_1	x_2	s_1	s_2	rhs	
1	-2	0	3	0	21	row 0 ← row 0 + 3 row 1
0	.5	1	.5	0	3.5	row 1 ← (row 1)/2
0	2.5	0	.5	1	6.5	row 2 ← (row 2 + 1/2 row 1); ratio test=2.6

The third and optimal tableau is:

z	x_1	x_2	s_1	s_2	rhs
1	0	0	3.4	0.8	26.2
0	0	1	.4	-.2	2.2
0	1	0	.2	.4	2.6

yielding the optimal solution: $x_1^* = 2.2; x_2^* = 2.6; z^* = 26.2$

As the solution is not all integer, we should branch if we want to proceed with branch and bound. We could also add Gomory cuts if we want to proceed with a cutting plane algorithm. Let us add a Gomory cut based on row 1:

$$x_2 + \frac{2}{5}s_1 - \frac{1}{5}s_2 = 2.2 \Rightarrow x_2 - s_2 - 2 = .2 - \frac{2}{5}s_1 - \frac{4}{5}s_2$$

so the cut is $2s_1 + 4s_2 \geq 1$. Plugging s_1 and s_2 as a function of x_1, x_2, we get

$$2(7 - x_1 - 2x_2) + 4(3 - 2x_1 + x_2) \geq 1 \Rightarrow x_1 \leq 2.5$$

which can be strengthened to

$$x_1 \leq 2.$$

Transforming the cut to $x_1 + s_3 = 2$, and adding it to the optimal tableau yields:

z	x_1	x_2	s_1	s_2	s_3	rhs
1	0	0	3.4	0.8	0	26.2
0	0	1	.4	-.2	0	2.2
0	1	0	.2	.4	0	2.6
0	1	0	0	0	1	2

The x_1 entry in row3 should be transformed to 0 to maintain a basic solution. So, it has to be replaced by row3-row2.

z	x_1	x_2	s_1	s_2	s_3	rhs
1	0	0	3.4	0.8	0	26.2
0	0	1	.4	-.2	0	2.2
0	1	0	.2	.4	0	2.6
0	0	0	-.2	-.4	1	-0.6

This basic solution is optimal (as reduced costs are ≥ 0), but is infeasible since one of the rhs's is negative. To fix feasibility we apply the **Dual Simplex method** (See [1] pages 329-334). Applying the dual simplex, we get the optimal solution in just one iteration:

z	x_1	x_2	s_1	s_2	s_3	rhs
1	0	0	3	0	2	25
0	1	0	0	0	1	2
0	0	1	0.5	0	-0.5	2.5
0	0	0	0.5	1	-2.5	1.5

with solution $x_1^* = 2; x_2^* = 2.5; z^* = 25$. As x_2 is non-integer, we should proceed. At this point we may choose to add an other cut or branch. Let us choose to branch on x_2. Two nodes are created:

$$\text{Node 1:} \left\{ \begin{array}{rl} \min & -5x_1 - 6x_2 \\ \text{s.t.} & x_1 + 2x_2 \leq 7 \\ & 2x_1 - x_2 \leq 3 \\ & x_1, x_2 \geq 0, \\ & \mathbf{x_1 \leq 2} \leftarrow \text{Valid cut} \\ & \mathbf{x_2 \leq 2} \leftarrow \text{Branching constraint} \end{array} \right\}$$

$$\text{Node 2:} \left\{ \begin{array}{rl} \min & -5x_1 - 6x_2 \\ \text{s.t.} & x_1 + 2x_2 \leq 7 \\ & 2x_1 - x_2 \leq 3 \\ & x_1, x_2 \geq 0, \\ & \mathbf{x_1} \leq \mathbf{2} \leftarrow \text{Valid cut} \\ & \mathbf{x_2} \geq \mathbf{3} \leftarrow \text{Branching constraint} \end{array} \right\}$$

Each node is resolved from the optimal tableau at the root using the **dual** simplex method. The optimal tableau for Node 1 after adding $x_2 + s_4 = 2$ is:

z	x_1	x_2	s_1	s_2	s_3	s_4	rhs
1	0	0	0	0	5	6	22
0	1	0	0	0	1	0	2
0	0	1	0	0	0	1	2
0	0	0	1	0	-1	-2	1
0	0	0	0	1	-2	1	1

and for Node 2 after adding $x_3 - s_5 = 3$ is: is

z	x_1	x_2	s_1	s_2	s_3	s_5	rhs
1	0	0	5	0	0	4	23
0	1	0	1	0	0	2	1
0	0	1	0	0	0	-1	3
0	0	0	-2	1	0	-5	4
0	0	0	-1	0	1	-2	1

The optimal solution for Node 1 is $x_1^* = 2; x_2^* = 2; z^* = 22$ and for Node 2 is $x_1^* = 1; x_2^* = 3; z^* = 23$. Both are an all integer, so we fathom both and stop the search. The optimal solution is that found at Node 2.

Note:

- In branch-and-cut, at every node, we either add cuts or decide to branch.
- If the addition of cuts is only done at the root node, then it is branch-and-bound with a preprocessing step where cuts are added to enhance the formulation.
- At child nodes, there are usually two types of cuts, *local* and *global*. Local cuts are only valid for that node and its children. Global cuts are valid for all nodes in the tree, including the root node.
- The use of the dual Simplex method is very efficient is finding the optimal solution after a cut or a branching constraint is added.