General Cutting Planes

Consider the following integer program:

$$\begin{array}{lrcrcl} \text{Maximize} & 7x_1 & + & 9x_2 & & \\ \text{subject to} & -x_1 & + & 3x_2 & \leq & 6 \\ & 7x_1 & + & x_2 & \leq & 35 \\ & x_1, & & x_2 & \geq & 0 \text{ integer.} \end{array}$$

If we ignore integrality, we get the following optimal tableau (with the updated columns and reduced costs shown for nonbasic variables):

Variable	x_1	x_2	s_1	s_2	$-z$	RHS
x_2	0	1	7/22	1/22	0	7/2
x_1	1	0	-1/22	3/22	0	9/2
$-z$	0	0	28/11	15/11	1	63

Let's look at the first constraint:

$$x_2 + 7/22 s_1 + 1/22 s_2 = 7/2$$

We can manipulate this to put all of the integer parts on the left side, and all the fractional parts on the right to get:

$$x_2 - 3 = 1/2 - 7/22 s_1 - 1/22 s_2$$

Now, note that the left hand side consists only of integers, so the right hand side must add up to an integer. Which integer can it be? Well, it consists of some positive fraction minus a series of positive values. Therefore, the right hand side can only be $0, -1, -2, \ldots$; it cannot be a positive value.

Therefore, we have derived the following constraint:

$$1/2 - 7/22 s_1 - 1/22 s_2 \leq 0.$$

This constraint is satisfied by every feasible integer solution to our original problem. But, in our current solution, s_1 and s_2 both equal 0, which is infeasible to the above constraint. This means the above constraint is a cut, called the *Gomory cut* after its discoverer. We can now add this constraint to the linear program and be guaranteed to find a different solution, one that might be integer.

We can also generate a cut from the other constraint. Here we have to be careful to get the signs right:

$$x_1 - 1/22 s_1 + 3/22 s_2 = 9/2$$
$$x_1 + (-1 + 21/22) s_1 + 3/22 s_2 = 4 + 1/2$$
$$x_1 - s_1 - 4 = 1/2 - 21/22 s_1 - 3/22 s_2$$

gives the constraint

$$1/2 - 21/22 s_1 - 3/22 s_2 \leq 0.$$

In general, let $\lfloor a \rfloor$ be defined as the largest integer less than or equal to **a**. For example, $\lfloor 3.9 \rfloor = 3$, $\lfloor 5 \rfloor = 5$, and $\lfloor -1.3 \rfloor = -2$.

If we have a constraint

$$x_k + \sum a_i x_i = b$$

with **b** not an integer, we can write each $a_i = \lfloor a_i \rfloor + a_i'$, for some $0 \leq a_i' < 1$, and $b = \lfloor b \rfloor + b'$ for some $0 < b' < 1$. Using the same steps we get:

$$x_k + \sum \lfloor a_i \rfloor x_i - \lfloor b \rfloor = b' - \sum a_i' x_i$$

to get the cut

$$b' - \sum a_i' x_i \leq 0.$$

This cut can then be added to the linear program and the problem resolved. The problem is guaranteed not to get the same solution.

This method can be shown to guarantee finding the optimal integer solution. There are a couple of disadvantages:

1. Round-off error can cause great difficulties: Is that 3.000000001 really a 3, or should I generate a cut? If I make the wrong decision I could either cut off a feasible solution (if it is really a 3 but I generate a cut) or I could end up with an infeasible solution (if it is not a 3 but I treat it as one).
2. The number of constraints that are generated can be enormous. Just like branch and bound can generate a huge number of subproblems, this technique can generate a huge number of constraints.

The combination of these makes this cutting plane technique impractical by itself. Recently however, more powerful techniques have been discovered for special problem structure. This is the subject of the next section.

Michael A. Trick
Thu Oct 26 12:05:11 EDT 1995