

Lecture 33: The Branch-and-Bound Method

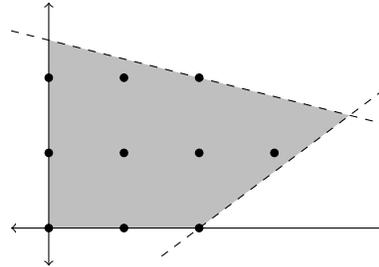
April 27, 2020

University of Illinois at Urbana-Champaign

1 Solving an example integer program

Consider the following integer program.

$$\begin{array}{ll} \text{maximize} & 4x + 5y \\ & x, y \in \mathbb{Z} \\ \text{subject to} & x + 4y \leq 10 \\ & 3x - 4y \leq 6 \\ & x, y \geq 0 \end{array}$$



The only feasible points of the integer program are the ten points marked as black dots in the diagram above, but we don't know how to find those yet. Right now, the only thing we know how to do is forget about the $x, y \in \mathbb{Z}$ constraint, and solve the linear program we get by taking $x, y \in \mathbb{R}$. This is called solving the *linear programming relaxation* of the integer program.

This is not a complicated linear program, and we can find the optimal solution in just two pivot steps:

$$\begin{array}{cc|cc|c} x & y & s_1 & s_2 & \\ \hline s_1 & 1 & 4 & 1 & 0 & 10 \\ s_2 & 3 & -4 & 0 & 1 & 6 \\ -z & 4 & 5 & 0 & 0 & 0 \end{array} \rightsquigarrow \begin{array}{cc|cc|c} x & y & s_1 & s_2 & \\ \hline y & 1/4 & 1 & 1/4 & 0 & 5/2 \\ s_2 & 4 & 0 & 1 & 1 & 16 \\ -z & 11/4 & 0 & -5/4 & 0 & -25/2 \end{array} \rightsquigarrow \begin{array}{cc|cc|c} x & y & s_1 & s_2 & \\ \hline y & 0 & 1 & 3/16 & -1/16 & 3/2 \\ x & 1 & 0 & 1/4 & 1/4 & 4 \\ -z & 0 & 0 & -31/16 & -11/16 & -47/2 \end{array}$$

We obtain the point $(x, y) = (4, 1.5)$ as our optimal solution, with objective value 23.5. But it does not solve the integer program, because y is not an integer. At the very least we can say the following, however:

Fact. The optimal objective value of the linear programming relaxation is an upper bound for the objective value of the integer program.

Whatever the optimal integer solution is, its objective value is at most 23.5. (It could be equal, or it could be smaller. The integer program might even be infeasible, which we'll often think of as an objective value of $-\infty$ for a maximization problem.)

The "branch" part of the branch-and-bound algorithm is named after the idea that whenever we get such a fractional optimal solution, we should consider two "branches" that both exclude

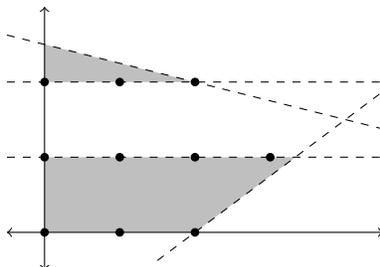
¹This document comes from the Math 482 course webpage: <https://faculty.math.illinois.edu/~mlavrov/courses/482-spring-2020.html>

it. If the optimal solution gives a fractional value $f \notin \mathbb{Z}$ to some variable x_i , we consider two possibilities:

$$x_i \leq \lfloor f \rfloor \quad \text{or} \quad x_i \geq \lceil f \rceil.$$

(Remember that in practical applications, our integer variables are often $\{0, 1\}$ -valued variables. In that case, the inequalities we get when x_i is fractional will correspond to $x_i = 0$ and $x_i = 1$.)

In the example above, we have $y = 1.5 \notin \mathbb{Z}$, so we should try two possibilities: either $y \geq 2$, or $y \leq 1$. These correspond to the top and bottom shaded regions of the diagram below, respectively. (Notice that we've eliminated the optimal solution $(4, 1.5)$ from the shaded region, but we haven't lost any integer points.)



We can add either $y \leq 1$ or $y \geq 2$ to our constraints, and we'll probably have to try both. Each one of those gives us a new linear program to solve. Let's try $y \geq 2$ first.

When we add a constraint, we don't have to solve the linear program from scratch. We can insert a new row for that constraint, getting a tableau that's still dual feasible. Here, we add a row corresponding to the inequality $y \geq 2$, or $-y + s_3 = -2$:

$$\begin{array}{cccc|c}
 x & y & s_1 & s_2 & \\
 \hline
 y & 0 & 1 & 3/16 & -1/16 & 3/2 \\
 x & 1 & 0 & 1/4 & 1/4 & 4 \\
 \hline
 -z & 0 & 0 & -31/16 & -11/16 & -47/2
 \end{array}
 \rightsquigarrow
 \begin{array}{cccc|c}
 x & y & s_1 & s_2 & s_3 & \\
 \hline
 y & 0 & 1 & 3/16 & -1/16 & 0 & 3/2 \\
 x & 1 & 0 & 1/4 & 1/4 & 0 & 4 \\
 s_3 & 0 & -1 & 0 & 0 & 1 & -2 \\
 \hline
 -z & 0 & 0 & -31/16 & -11/16 & 0 & -47/2
 \end{array}$$

Before we do anything, we should row-reduce the resulting tableau to eliminate the improper -1 in the y column. This gives us the first tableau below. Then, we use the dual simplex method to find a new optimal solution. (Here, we're done after s_3 replaces s_2 in the basis, giving us the second tableau below.)

$$\begin{array}{cccc|c}
 x & y & s_1 & s_2 & s_3 & \\
 \hline
 y & 0 & 1 & 3/16 & -1/16 & 0 & 3/2 \\
 x & 1 & 0 & 1/4 & 1/4 & 0 & 4 \\
 s_3 & 0 & 0 & 3/16 & -1/16 & 1 & -1/2 \\
 \hline
 -z & 0 & 0 & -31/16 & -11/16 & 0 & -47/2
 \end{array}
 \rightsquigarrow
 \begin{array}{cccc|c}
 x & y & s_1 & s_2 & s_3 & \\
 \hline
 y & 0 & 1 & 0 & 0 & -1 & 2 \\
 x & 1 & 0 & 1 & 0 & 4 & 2 \\
 s_2 & 0 & 0 & -3 & 1 & -16 & 8 \\
 \hline
 -z & 0 & 0 & -4 & 0 & -11 & -18
 \end{array}$$

We've obtained the optimal solution $(x, y) = (2, 2)$, with objective value 18. This doesn't mean we're done, though! There was still another branch to consider, the $y \leq 1$ branch, and that one might lead to a larger objective value.

- A list \mathcal{L} of “nodes”: linear programs describing all or part of our feasible region. (Initially, \mathcal{L} consists only of one linear program: the linear programming relaxation of our integer program.)
- A candidate optimal solution \mathbf{x}^* to the integer program, and its objective value z^* . (Initially, there is no candidate \mathbf{x}^* , and we set $z^* = -\infty$.)

A step in the algorithm examines a single node in \mathcal{L} : we solve the linear program associated with that node, and remove that node from \mathcal{L} . Let \mathbf{x} be the optimal solution we get, and let z be its objective value. There are several possibilities:

- If $z \leq z^*$, then this node can’t lead to any better solutions than what we’ve already got. We don’t explore it further. We call the node *pruned by bound*.

As a kind of special case, if the linear program is infeasible, then of course we can’t continue exploring that node, and we call it *pruned by infeasibility*. We can think of this as getting the objective value $z = -\infty$.

- If $z > z^*$, and all components of \mathbf{x} are integers, we save this as our best solution so far: set $\mathbf{x}^* = \mathbf{x}$ and $z^* = z$. We don’t explore the node further; there’s no point. We call it *pruned by integrality*.
- If $z > z^*$, but some component x_i has a fractional value f in \mathbf{x} , we branch. We add two new nodes to \mathcal{L} : one which adds the constraint $x_i \leq \lfloor f \rfloor$ to the current node, and one which adds the constraint $x_i \geq \lceil f \rceil$.

We repeat this procedure for as long as we can. Once \mathcal{L} is empty (there are no more unexplored nodes), the current value of \mathbf{x}^* is our optimal solution. It’s possible that we still have $z^* = -\infty$, and no \mathbf{x}^* ; if so, there are no feasible solutions to the integer program.

It’s convenient to represent this procedure by a tree of the branches we took. For example, here is a tree representation of our example:

