

# Column generation

Let a linear programming (LP) problem be

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \in \mathbb{R}_+^n. \end{aligned}$$

Column generation (also called *delayed* column generation), is a general approach for solving LP problems that have too many columns to consider them all (i.e.,  $n \gg m$ ). It proceeds as follows:

**Step 1.** Consider an LP having only a subset of the columns. This is a restriction of the original LP since we are implicitly fixing all the variables associated to the missing columns to zero (and nonbasic). Optimize over this restricted LP.

**Step 2.** Consider an optimal basis of the restricted LP. For that basis, in the original LP, find a column that is currently not in the restricted LP, but has a negative reduced cost: find  $j$  such that  $c_j - c_B B^{-1} A_j < 0$  where  $A_j$  is the  $j$ th column of the original problem. In general, this would require enumerating all the columns of the original LP. However, it is often possible to express “finding the minimum reduced cost” as an optimization problem (called the *pricing* problem). If there exists a column with a negative reduced cost, add it to the restricted LP, and repeat. Otherwise, the current basis is optimal for the original LP, and we are done.

In order to perform column generation without enumerating all columns, one needs to express a pricing problem that finds the relevant columns. This pricing problem will depend on the original LP problem. Here, we give an example where the original LP is a formulation for the *cutting-stock* problem.

## 1 The cutting-stock problem

**Problem statement.** We are given an infinite supply of *stocks* of a fixed size (stocks is a generic term that can represent any material, say, steel beams, or fabric sheets). We want to cut them into a given number of *pieces* of given sizes. Arrange the cuts so as to minimize the number of stocks required.

input:	<div style="border: 1px solid black; padding: 2px 10px; display: inline-block;">17</div>	$\infty \times$
output:	<div style="border: 1px solid black; padding: 2px 10px; display: inline-block;">3</div>	$25 \times$
	<div style="border: 1px solid black; padding: 2px 10px; display: inline-block;">5</div>	$20 \times$
	<div style="border: 1px solid black; padding: 2px 10px; display: inline-block;">9</div>	$15 \times$

**Notation for data.**

$a_i$  is the length for pieces of type  $i$ , for  $i = 1, 2, \dots, m$ ,

$d_i$  is the demand for pieces of type  $i$ , for  $i = 1, 2, \dots, m$ ,

$b$  is the length of a stock.

In our example,  $b = 17$ ,  $a = [3 \ 5 \ 9]^T$  and  $d = [25 \ 20 \ 15]^T$ .

**Variables.** A *pattern* is an arrangement of pieces that can be cut from a single stock. In our example, we can cut, from a single stock, 2 pieces of size 3, 2 pieces of size 5, and 0 piece of size 9. This gives us the pattern  $[2 \ 2 \ 0]^T$ . Now, consider all possible patterns; we number them  $p = 1, 2, \dots$  and denote them

$$Q_p = \begin{bmatrix} q_{1p} \\ \vdots \\ q_{mp} \end{bmatrix}$$

where pattern  $p$  contains  $q_{ip}$  pieces of type  $i$ , for  $i = 1, \dots, m$ . Remark that for all  $p$ , the pattern  $Q_p$  will satisfy

$$\sum_{i=1}^n q_{ip} a_i \leq b.$$

Our variables will be  $x_p$  the number of times we select pattern  $p$ , for all  $p$ .

**Formulation.**

$$\begin{aligned} \min \quad & \sum_p x_p \\ \text{s.t.} \quad & \sum_p Q_p x_p \geq d \quad \text{for all } p \\ & x_p \geq 0 \quad \text{for all } p \\ & x_p \in \mathbb{Z} \quad \text{for all } p \end{aligned}$$

In the example, some patterns are

$$\begin{aligned} Q_1 &= [5 \ 0 \ 0]^T \\ Q_2 &= [0 \ 3 \ 0]^T \\ Q_3 &= [0 \ 0 \ 1]^T \\ Q_4 &= [2 \ 2 \ 0]^T \\ &\vdots \end{aligned}$$

and the formulation becomes:

$$\begin{aligned} \min \quad & x_1 + x_2 + x_3 + x_4 + \dots \\ \text{s.t.} \quad & \begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 + \begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix} x_4 + \dots \geq \begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} \\ & x_p \geq 0 \quad \text{for all } p \\ & x_p \in \mathbb{Z} \quad \text{for all } p \end{aligned} \tag{LP}$$

which we can solve by branch-and-bound.

**Column generation for the cutting-stock problem.** In many cases, there are too many distinct patterns to enumerate them all, even just to solve the LP relaxation. Therefore, we apply column generation: we start with

a small subset of the possible patterns. Typically, we first use the simplest patterns, those that use only pieces of a single type. In our example with  $b = 17$  and  $a = [3 \ 5 \ 9]^T$ , those are

$$\begin{aligned} Q_1 &= [5 \ 0 \ 0]^T \\ Q_2 &= [0 \ 3 \ 0]^T \\ Q_3 &= [0 \ 0 \ 1]^T \end{aligned}$$

which give us the restricted LP relaxation

$$\begin{aligned} \min \quad & x_1 + x_2 + x_3 \\ \text{s.t.} \quad & \begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 \geq \begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} \\ & x \in \mathbb{R}_+^3. \end{aligned} \tag{R1}$$

We can add “slack” variables to obtain equality constraints:

$$\begin{aligned} \min \quad & x_1 + x_2 + x_3 \\ \text{s.t.} \quad & \begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 + \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} s_1 + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} s_2 + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} s_3 = \begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} \\ & x \in \mathbb{R}_+^3 \\ & s \in \mathbb{R}_+^3 \end{aligned}$$

and solve the above with the simplex method. We get an optimal primal solution  $x^* = [5 \ \frac{20}{3} \ 15]^T$ , and an optimal dual solution  $y^* = [\frac{1}{5} \ \frac{1}{3} \ 1]^T$ . Recall that  $y^* = c_B B^{-1}$  where  $\mathcal{B}$  is the set of basic columns in the optimal basis, and  $B$  is the corresponding basis matrix. Now,  $x^*$  is feasible for both (R1) and (LP), and it is optimal for (R1), but is it also optimal for (LP)? In order to answer this question, we need to compute the *reduced costs* associated to the basis  $\mathcal{B}$  in (LP). They are given by  $\bar{c} = c - c_B B^{-1} A = c - y^{*T} A$ , where  $A$  is the matrix of the constraint coefficients, i.e.,  $A = [Q_1 \ Q_2 \ \cdots \ -I]$ . We thus have  $\bar{c}_p = c_p - y^{*T} Q_p = 1 - y^* Q_p$  for all  $p$ . If  $\bar{c}_p \geq 0$  for every pattern  $p$  (including those not considered so far), then  $\mathcal{B}$  is optimal not only for (R1) but also for (LP), and  $x^*$  is an optimal solution to (LP). Otherwise, if  $\bar{c}_p < 0$  for some  $p$ , then the column  $Q_p$  must be added to (R1) in order for it to enter the basis.

Note that answering “ $\exists p : \bar{c}_p < 0$ ?” is equivalent to answering “ $\min_p \bar{c}_p < 0$ ?”. The latter asks whether there exist a pattern  $Q_p$  that minimizes  $\bar{c}_p = 1 - y^{*T} Q_p$ . This can be formulated as an optimization problem where the variables  $q_1, \dots, q_m$  will be the elements of  $Q_p$ , i.e., the number of pieces of each type that we take in the pattern:

$$\begin{aligned} \min \quad & 1 - \sum_{i=1}^m y_i^* q_i & = \quad & 1 - \max \sum_{i=1}^m y_i^* q_i \\ \text{s.t.} \quad & \sum_{i=1}^m a_i q_i \leq b & & \text{s.t.} \quad \sum_{i=1}^m a_i q_i \leq b \\ & q \in \mathbb{Z}_+^m & & q \in \mathbb{Z}_+^m. \end{aligned}$$

This is the *pricing* problem for our cutting-stock formulation. Observe that it is an integer knapsack. Going back

to the example, we get

$$\begin{aligned}
& 1 - \max_{\substack{\frac{1}{5}q_1 + \frac{1}{3}q_2 + 1q_3 \\ \text{s.t. } 3q_1 + 5q_2 + 9q_3 \leq 17 \\ q \in \mathbb{Z}_+^3}} \\
&= 1 - \frac{1}{15} \max_{\substack{3q_1 + 5q_2 + 15q_3 \\ \text{s.t. } 3q_1 + 5q_2 + 9q_3 \leq 17 \\ q \in \mathbb{Z}_+^3}} \\
&= 1 - \frac{1}{15} [3 \ 5 \ 15] [1 \ 1 \ 1]^T \\
&= 1 - \frac{23}{15} = -\frac{8}{15}
\end{aligned}$$

where the optimal solution  $[1 \ 1 \ 1]^T$  to the knapsack problem was found by dynamic programming. So the column  $[1 \ 1 \ 1]^T$  has a negative reduced cost  $-\frac{8}{15}$ , and we add it to our restricted formulation, yielding

$$\begin{aligned}
& \min \quad x_1 + x_2 + x_3 + x_4 \\
& \text{s.t.} \quad \begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} x_4 \geq \begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} \\
& \quad \quad \quad x \in \mathbb{R}_+^4.
\end{aligned} \tag{R2}$$

Applying the simplex method on (R2), we get an optimal primal solution  $x^* = [2 \ \frac{5}{3} \ 0 \ 15]^T$ , and an optimal dual solution  $y^* = [\frac{1}{5} \ \frac{1}{3} \ \frac{7}{15}]^T$ . The pricing problem is now

$$\begin{aligned}
& 1 - \max_{\substack{\frac{1}{5}q_1 + \frac{1}{3}q_2 + \frac{7}{15}q_3 \\ \text{s.t. } 3q_1 + 5q_2 + 9q_3 \leq 17 \\ q \in \mathbb{Z}_+^3}} \\
&= 1 - \frac{1}{15} \max_{\substack{3q_1 + 5q_2 + 7q_3 \\ \text{s.t. } 3q_1 + 5q_2 + 9q_3 \leq 17 \\ q \in \mathbb{Z}_+^3}} \\
&= 1 - \frac{1}{15} [3 \ 5 \ 7] [4 \ 1 \ 0]^T \\
&= 1 - \frac{17}{15} = -\frac{2}{15}
\end{aligned}$$

where the optimal solution  $[4 \ 1 \ 0]^T$  to the knapsack problem was found, again, by dynamic programming. So we still have a negative reduced cost  $-\frac{2}{15}$  for the column  $[4 \ 1 \ 0]^T$ . We add that column, too, to our restricted formulation:

$$\begin{aligned}
& \min \quad x_1 + x_2 + x_3 + x_4 + x_5 \\
& \text{s.t.} \quad \begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} x_4 + \begin{bmatrix} 4 \\ 1 \\ 0 \end{bmatrix} x_5 \geq \begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} \\
& \quad \quad \quad x \in \mathbb{R}_+^5.
\end{aligned} \tag{R3}$$

Applying the simplex method on (R3), we get an optimal primal solution  $x^* = [0 \ \frac{5}{6} \ 0 \ 15 \ \frac{5}{2}]^T$ , and an optimal dual solution  $y^* = [\frac{1}{6} \ \frac{1}{3} \ \frac{1}{2}]^T$ . The pricing problem is now

$$\begin{aligned}
& 1 - \max \quad \frac{1}{6}q_1 + \frac{1}{3}q_2 + \frac{1}{2}q_3 \\
& \quad \text{s.t.} \quad 3q_1 + 5q_2 + 9q_3 \leq 17 \\
& \quad \quad \quad q \in \mathbb{Z}_+^3 \\
& = 1 - \frac{1}{6} \max \quad 1q_1 + 2q_2 + 3q_3 \\
& \quad \text{s.t.} \quad 3q_1 + 5q_2 + 9q_3 \leq 17 \\
& \quad \quad \quad q \in \mathbb{Z}_+^3. \\
& = 1 - \frac{1}{6} [1 \ 2 \ 3] [4 \ 1 \ 0]^T \\
& = 1 - \frac{6}{6} = 0
\end{aligned}$$

where the optimal solution  $[4 \ 1 \ 0]^T$  to the knapsack problem was found, again, by dynamic programming. This time, since  $\min_p \bar{c}_p \geq 0$ , we now that  $\bar{c}_p \geq 0$  for all the reduced costs of (LP). Therefore,  $x^* = [0 \ \frac{5}{6} \ 0 \ 15 \ \frac{5}{2}]^T$  is an optimal solution not only to (R3), but also to (LP). It has an objective function value of  $\frac{55}{3} \approx 18.33$ . We now have a solution to the LP relaxation of our formulation, and can proceed to perform branch-and-bound. However, it is not necessary in our particular example, as we will demonstrate.

**Observation 1.** Given an IP formulation in minimization form, if  $z^{LP}$  is the optimal objective function value of the LP relaxation, then any integer solution will have cost  $z^{IP} \geq z^{LP}$ . In a cutting-stock problem, the objective function only has integer coefficients, so  $z^{IP} \in \mathbb{Z}$ , and we can write  $z^{IP} \geq \lceil z^{LP} \rceil$ . In our example, this means that  $z^{IP} \geq \lceil \frac{55}{3} \rceil = 19$ .

**Observation 2.** In a cutting-stock problem, all constraints are of the form “ $\geq$ ” and  $A_{ij} \geq 0$  for all  $i, j$ . Therefore, any solution  $x'$  such that  $x' \geq x^*$  is LP-feasible. In particular  $x' := \lceil x^* \rceil$  is LP-feasible and integral, so it is IP-feasible. In the example,  $x' = [0 \ 1 \ 0 \ 15 \ 3]^T$  has objective function value 19, so it is optimal. Having found this integer solution  $x'$  and having proven that it is optimal, there is no need to perform branch-and-bound.

Our optimal solution corresponds to taking  $x_2^* = 1$  times the pattern  $Q_2 = [0 \ 3 \ 0]^T$ ,  $x_4^* = 15$  times the pattern  $Q_4 = [1 \ 1 \ 1]^T$ , and  $x_5^* = 3$  times the pattern  $Q_5 = [4 \ 1 \ 0]^T$ :

1×	5	5	5	
15×	3	5	9	
3×	3	3	3	5

giving:

- 27 pieces of length 3,
- 21 pieces of length 5,
- 15 pieces of length 9,

and some waste (length 2), using a total of 19 stocks.

## 2 Second example

We now consider a second, smaller example, that lets us solve all LPs and knapsacks by hand. The data are given by:

$$\begin{array}{ll} \text{input: } & \boxed{b = 11} \\ \\ \text{output: } & \boxed{a_1 = 3} \qquad d_1 = 18 \\ & \boxed{a_2 = 4} \qquad d_2 = 16 \end{array}$$

**Iteration 1: LP.** We start with the two simple patterns  $[3 \ 0]^T$  and  $[0 \ 2]^T$ .

$$\begin{array}{ll} \min & x_1 + x_2 \\ \text{s.t.} & \begin{bmatrix} 3 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 2 \end{bmatrix} x_2 \geq \begin{bmatrix} 18 \\ 16 \end{bmatrix} \\ & x \in \mathbb{R}_+^2 \end{array}$$

The only way to satisfy the first constraint is to have  $3x_1 \geq 18$ . Similarly, for the second, we need  $2x_2 \geq 16$ . We thus immediately know the optimal solution  $x^* = [6 \ 8]^T$ . However, we still need a corresponding dual solution, so we write down our initial problem algebraically

$$\begin{array}{ll} c = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix} \\ \\ A = \begin{bmatrix} 3 & 0 & -1 & 0 \\ 0 & 2 & 0 & -1 \end{bmatrix} & b = \begin{bmatrix} 18 \\ 16 \end{bmatrix} \end{array}$$

and set our basis to  $\mathcal{B} = \{1, 2\}$ . We get a basis matrix  $B$  compute its inverse:

$$B = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \qquad B^{-1} = \begin{bmatrix} 1/3 & 0 \\ 0 & 1/2 \end{bmatrix}$$

Because we will need it later, we write the tableau corresponding to  $\mathcal{B}$ :

$$\begin{array}{ll} \bar{c} = \begin{bmatrix} 0 & 0 & 1/3 & 1/2 \end{bmatrix} \\ \\ \bar{A} = \begin{bmatrix} 1 & 0 & -1/3 & 0 \\ 0 & 1 & 0 & -1/2 \end{bmatrix} & \bar{b} = \begin{bmatrix} 6 \\ 8 \end{bmatrix} \end{array}$$

Observe that the first two columns of  $\bar{A}$  are an identity because  $x_1$  and  $x_2$  are basic. Furthermore, the two subsequent columns are  $B^{-1}(-I) = -B^{-1}$ . We can compute  $\bar{c}$  after  $\bar{A}$ , so that we can use  $\bar{c}_j = c_j - [1 \ 1]\bar{A}$ . Again,  $\bar{c}_0$  and  $\bar{c}_1$  are zero because  $x_1$  and  $x_2$  are basic. Finally, we compute the dual solution

$$y^{*T} = c_{\mathcal{B}} B^{-1} = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1/3 & 0 \\ 0 & 1/2 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/2 \end{bmatrix}.$$

**Iteration 1: pricing.** The pricing problem looks for  $\bar{c}_p = c_p - y^* Q_p < 0$ . It corresponds to the knapsack given below:

$$\begin{aligned} \min_p \bar{c}_p &= \min \quad 1 - \frac{1}{3}q_1 - \frac{1}{2}q_2 \\ &\text{s.t.} \quad 3q_1 + 9q_2 \leq 11 \\ &\quad q \in \mathbb{Z}_+^2 \\ &= 1 - \frac{1}{6} \max \quad 2q_1 + 3q_2 \\ &\quad \text{s.t.} \quad 3q_1 + 9q_2 \leq 11 \\ &\quad q \in \mathbb{Z}_+^2 \end{aligned}$$

	$r = 1$	$r = 2$
$\lambda = 0$	0	0
1	0	0
2	0	0
3	2	2
4	2	3
5	2	3
6	4	4
7	4	5
8	4	6
9	6	6
10	6	7
11	6	8

We find  $\min_p \bar{c}_p = 1 - \frac{1}{6}8 = -\frac{1}{3}$ , with a pattern  $q^* = [1 \ 2]^T$ .

**Iteration 2: LP.** Once we add the pattern  $[1 \ 2]^T$ , we get the restricted LP

$$\begin{aligned} \min \quad & x_1 + x_2 + x_3 \\ \text{s.t.} \quad & \begin{bmatrix} 3 \\ 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 2 \end{bmatrix} x_2 + \begin{bmatrix} 1 \\ 2 \end{bmatrix} x_3 \geq \begin{bmatrix} 18 \\ 16 \end{bmatrix} \\ & x \in \mathbb{R}_+^3 \end{aligned}$$

which corresponds to adding a column to the algebraic representation

$$\begin{aligned} c &= \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \end{bmatrix} \\ A &= \begin{bmatrix} 3 & 0 & 1 & -1 & 0 \\ 0 & 2 & 2 & 0 & -1 \end{bmatrix} \quad b = \begin{bmatrix} 18 \\ 16 \end{bmatrix} \end{aligned}$$

If we reuse the basis  $\mathcal{B} = \{1, 2\}$  that was optimal at iteration 1, we immediately get the tableau

$$\bar{c} = \begin{bmatrix} 0 & 0 & -1/3 & 1/3 & 1/2 \end{bmatrix}$$

$$\bar{A} = \begin{bmatrix} 1 & 0 & -1/3 & 0 \\ 0 & 1 & 0 & -1/2 \end{bmatrix} \quad \bar{b} = \begin{bmatrix} 6 \\ 8 \end{bmatrix}$$

where  $\bar{c}_3 = -\frac{1}{3}$  is given by the dynamic programming algorithm at the end of iteration 1, and  $\bar{A}_3 = B^{-1}Q_3 = B^{-1} \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ . Clearly,  $x_3$  should enter the basis. Applying one pivot of the primal simplex method,  $x_2$  leaves the basis, yielding  $\mathcal{B} = \{1, 3\}$  and

$$B = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix} \quad B^{-1} = \begin{bmatrix} 1/3 & -1/6 \\ 0 & 1/2 \end{bmatrix}.$$

The corresponding tableau is computed as for the first iteration:

$$\bar{c} = \begin{bmatrix} 0 & 1/3 & 0 & 1/3 & 1/3 \end{bmatrix}$$

$$\bar{A} = \begin{bmatrix} 1 & -1/3 & 0 & -1/3 & -1/6 \\ 0 & 1 & 1 & 0 & -1/2 \end{bmatrix} \quad \bar{b} = \begin{bmatrix} 10/3 \\ 8 \end{bmatrix}$$

and we now have an optimal solution (because  $\bar{c} \geq 0$ ) to the restricted LP

$$x^* = \begin{bmatrix} \frac{10}{3} & 0 & 8 \end{bmatrix}^T$$

with objective function value  $\frac{34}{2} \approx 11.33$ . We also have a dual solution

$$y^{*T} = c_{\mathcal{B}} B^{-1} = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1/3 & -1/6 \\ 0 & 1/2 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/3 \end{bmatrix}.$$

**Iteration 2: pricing.** The pricing problem is the same as previously, only with a new value of  $y^*$ :

$$\begin{aligned} \min_p \bar{c}_p &= \min \quad 1 - \frac{1}{3}q_1 - \frac{1}{3}q_2 \\ &\text{s.t.} \quad 3q_1 + 9q_2 \leq 11 \\ &\quad q \in \mathbb{Z}_+^2 \\ &= 1 - \frac{1}{3} \max_{\text{s.t. } 3q_1 + 9q_2 \leq 11} q_1 + q_2 \\ &\quad q \in \mathbb{Z}_+^2 \end{aligned}$$



	$r = 1$	$r = 2$
$\lambda = 0$	0	0
1	0	0
2	0	0
3	1	1
4	1	1
5	1	1
6	2	2
7	2	2
8	2	2
9	3	3
10	3	3
11	3	3

We find  $\min_p \bar{c}_p = 1 - \frac{1}{3}3 = -\frac{1}{3}$ , with  $q^* = [3 \ 0]^T$ . We conclude that  $\bar{c}_p \geq 0$  for all patterns, including those not in the restricted LP. Our solution  $x^* = \left[ \frac{10}{3} \ 0 \ 8 \right]^T$  is thus optimal for the original LP (the one that considers all possible patterns).

**Integer solution** Because the objective function value of  $x^*$  is  $\frac{34}{2} \approx 11.33$ , any integer solution will have objective function value 12 or more. We construct

$$x' = \lceil x^* \rceil = \left\lceil \left[ \frac{10}{3} \ 0 \ 8 \right]^T \right\rceil = \left[ 4 \ 0 \ 8 \right]^T$$

It is IP-feasible and has objective function value 12 so it is optimal.

Our optimal solution corresponds to taking  $x_1^* = 4$  times the pattern  $Q_1 = [3 \ 0]^T$ ,  $x_3^* = 8$  times the pattern  $Q_3 = [1 \ 2]^T$ :

4×	3	3	3	
8×	3	4	4	

giving:

20 pieces of length 3,

16 pieces of length 4,

and some waste (length 2), using a total of 12 stocks.