L’algoritmo del simplesso - fondamenti

Ricerca Operativa [035IN]

Lorenzo Castelli

30 Settembre 2021

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Corner-point solutions

- A constraint boundary is a line that forms the boundary of what is permitted by the corresponding constraint.
- The points of intersection are the corner-point solutions (vertice) of the problem. The five that lie on the corners of the feasible region—(0, 0), (0, 6), (2, 6), (4, 3), and (4, 0)—are the corner-point feasible solutions (CPF solutions). The other three—(0, 9), (4, 6), and (6, 0)—are called corner-point infeasible solutions.

In this example, each corner-point solution lies at the intersection of two constraint boundaries. For a LP problem with n decision variables, each of its corner-point solutions lies at the intersection of n constraint boundaries.

Finding all vertices - example

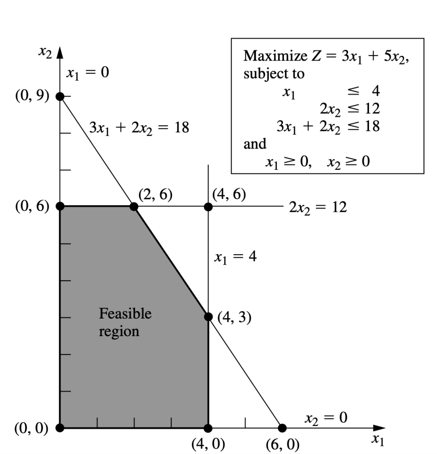


Figura 1: Constraint boundaries and corner-point solutions

Adjacent CPF solutions

Certain pairs of the CPF solutions in Fig. 1 share a constraint boundary, and other pairs do not. It will be important to distinguish between these cases by using the following general definitions.

Definitions

For any LP problem with n decision variables, two CPF solutions are adjacent to each other if they share $n-1$ constraint boundaries. The two adjacent CPF solutions are connected by a line segment that lies on these same shared constraint boundaries. Such a line segment is referred to as an edge (spigolo) of the feasible region.

Adjacent CPF solutions

Since $n = 2$ in the example, two of its CPF solutions are adjacent if they share one constraint boundary; for example, (0, 0) and (0, 6) are adjacent because they share the $x_1 = 0$ constraint boundary. The feasible region in Fig. 1 has five edges, consisting of the five line segments forming the boundary of this region. Note that two edges emanate from each CPF solution. Thus, each CPF solution has two adjacent CPF solutions (each lying at the other end of one of the two edges), as enumerated in Table 1.

CPF solution	Its adjacent CPF solutions
(0,0)	(0,6) and (4,0)
(0,6)	(2,6) and (0,0)
(2,6)	(4,3) and (0,6)
(4,3)	(4,0) and (2,6)
(4,0)	(0,0) and (4,3)

Tabella 1: Adjacent CPF solutions

The optimal solution is on a vertex

Property 1

1. If there is exactly one optimal solution, then it must be a CPF solution.
2. If there are multiple optimal solutions (and a bounded feasible region), then at least two must be adjacent CPF solutions.

The optimal solution is on a vertex

Proof of 1: By contradiction, by assuming that there is exactly one optimal solution and that it is not a CPF solution. Recall the definition of CPF solution (a feasible solution that does not lie on any line segment connecting two other feasible solutions). Since we have assumed that the optimal solution x^* is not a CPF solution, this implies that there must be two other feasible solutions such that the line segment connecting them contains the optimal solution. Let the vectors $x^{'}$ and $x^{''}$ denote these two other feasible solutions, and let Z_1 and Z_2 denote their respective objective function values. Like each other point on the line segment connecting $x^{'}$ and $x^{''}$,

$$x^* = \alpha x^{''} + (1-\alpha)x^{'}$$

for some value of α such that $0 < \alpha < 1$ Thus,

$$Z^* = \alpha Z_2 + (1-\alpha)Z_1$$

Since the weights α and $1-\alpha$ add to 1, the only possibilities for how Z^*, Z_1, and Z_2 compare are

1. $Z^* = Z_1 = Z_2$
2. $Z_1 < Z^* < Z_2$
3. $Z_1 > Z^* > Z_2$

The first possibility implies that $x^{'}$ and $x^{''}$ also are optimal, which contradicts the assumption that there is exactly one optimal solution. Both the latter possibilities contradict the assumption that x^* (not a CPF solution) is optimal. The resulting conclusion is that it is impossible to have a single optimal solution that is not a CPF solution.

The optimal solution is on a vertex

Proof of 2: What happens when you are solving graphically is that the objective function line keeps getting raised until it contains the line segment connecting the two CPF solutions.

The same thing would happen in higher dimensions except that an objective function hyperplane would keep getting raised until it contained the line segment(s) connecting two (or more) adjacent CPF solutions.

As a consequence, all optimal solutions can be obtained as weighted averages of optimal CPF solutions.

A finite number of vertices

Property 2

There are only a finite number of CPF solutions.

Each CPF solution is the simultaneous solution of a system of n out of the $m+n$ constraint boundary equations. The number of different combinations of $m+n$ equations taken n at a time is

$$\binom{m+n}{n} = \frac{(m+n)!}{m!n!},$$

which is a finite number.

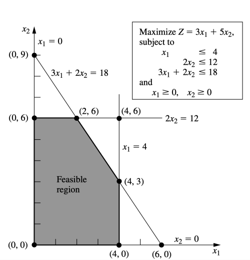
Exhaustive enumeration might not be possible: A rather small linear programming problem with only $m = 50$ and $n = 50$ would have $\frac{100!}{(50!)2} \approx 10^{29}$ systems of equations to be solved!

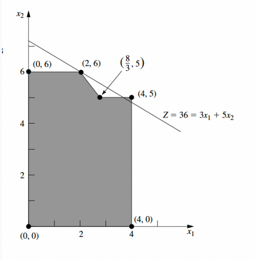
By contrast, the simplex method would need to examine only approximately 100 CPF solutions for a problem of this size.

The feasible region is convex

Property 3

If a CPF solution has no adjacent CPF solutions that are better (as measured by Z), then there are no better CPF solutions anywhere. Therefore, such a CPF solution is guaranteed to be an optimal solution (by Property 1), assuming only that the problem possesses at least one optimal solution (guaranteed if the problem possesses feasible solutions and a bounded feasible region).





Optimality test

Consider any linear programming problem that possesses at least one optimal solution. If a CPF solution has no adjacent CPF solutions that are better (as measured by Z), then it must be an optimal solution

Thus, for the example, $(2, 6)$ must be optimal simply because its $Z = 36$ is larger than $Z = 30$ for $(0, 6)$ and $Z = 27$ for $(4, 3)$. This optimality test is the one used by the simplex method for determining when an optimal solution has been reached.

The simplex algorithm - i

Initialisation Choose $(0,0)$ as the initial CPF solution to examine. (This is a convenient choice because no calculations are required to identify this CPF solution.)

Optimality Test Conclude that $(0,0)$ is not an optimal solution. (Adjacent CPF solutions are better.)

Iteration 1 Move to a better adjacent CPF solution, $(0,6)$, by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from $(0,0)$, choose to move along the edge that leads up the x_2 axis. (With an objective function of $Z = 3x_1 + 5x_2$, moving up the x_2 axis increases Z at a faster rate than moving along the x_1 axis.)
2. Stop at the first new constraint boundary: $2x_2 = 12$. (Moving farther in the direction selected in step 1 leaves the feasible region; e.g., moving to the second new constraint boundary hit when moving in that direction gives $(0,9)$, which is a corner-point infeasible solution.)
3. Solve for the intersection of the new set of constraint boundaries: $(0,6)$. (The equations for these constraint boundaries, $x_1 = 0$ and $2x_2 = 12$, immediately yield this solution.)

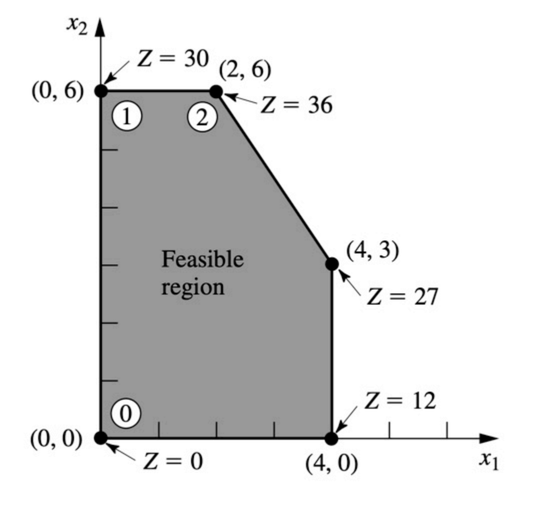
Optimality Test Conclude that $(0,6)$ is not an optimal solution. (An adjacent CPF solution is better.)

Iteration 2 Move to a better adjacent CPF solution, $(2, 6)$, by performing the following three steps.

1. Considering the two edges of the feasible region that emanate from $(0, 6)$, choose to move along the edge that leads to the right. (Moving along this edge increases Z, whereas backtracking to move back down the x_2 axis decreases Z.)
2. Stop at the first new constraint boundary encountered when moving in that direction: $3x_1 + 2x_2 = 18$. (Moving farther in the direction selected in step 1 leaves the feasible region.)
3. Solve for the intersection of the new set of constraint boundaries: $(2, 6)$. (The equations for these constraint boundaries, $3x_1 + 2x_2 = 18$ and $2x_2 = 12$, immediately yield this solution.)

Optimality Test Conclude that $(2, 6)$ is an optimal solution, so stop. (None of the adjacent CPF solutions are better.)

The simplex algorithm - iii



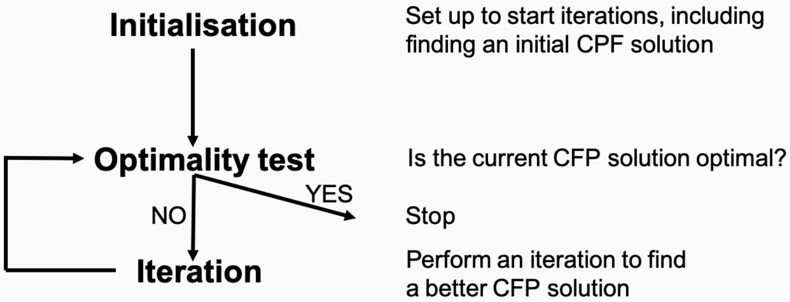
Relationships between optimal and CPF solutions

The simplex method focuses solely on CPF solutions. For any problem with at least one optimal solution, finding one requires only finding a best CFP solution.

Since the number of feasible solutions generally is infinite, reducing the number of solutions that need to be examined to a small finite number (just three in our example) is a tremendous simplification.

The flow of the simplex method

The simplex method is an iterative algorithm with the following structure



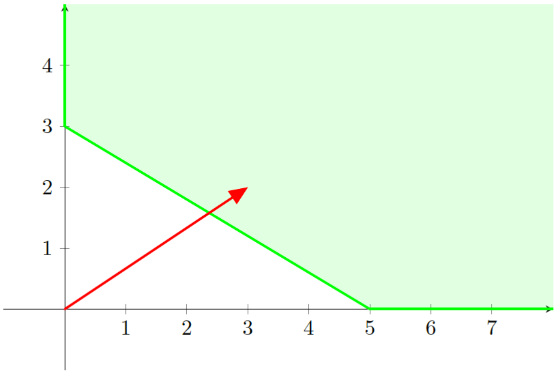
When the example was solved, this flow diagram was followed through two iterations until an optimal solution was found.

How to get started

Whenever possible, the initialisation of the simplex method chooses the origin (all decision variables equal to zero) to be the initial CPF solution. When there are too many decision variables to find an initial CPF solution graphically, this choice eliminates the need to use algebraic procedures to find and solve for an initial CPF solution.

Choosing the origin commonly is possible when all the decision variables have nonnegativity constraints, because the intersection of these constraint boundaries yields the origin as a corner-point solution. This solution then is a CPF solution unless it is infeasible because it violates one or more constraints. If it is infeasible, special procedures are needed to find the initial CPF solution.

Concept 3 - $(0,0)$ is not a CPF

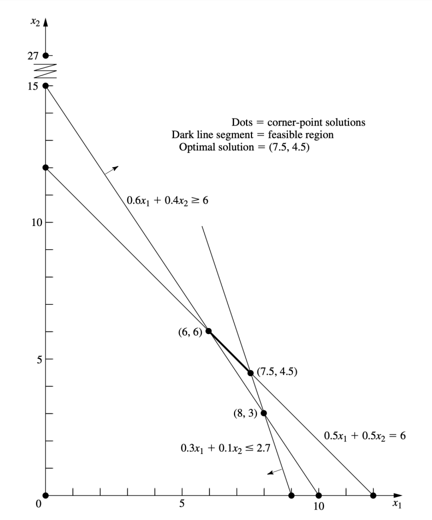


The corner-point $(0,0)$ is not a feasible solution (i.e., a CPF)

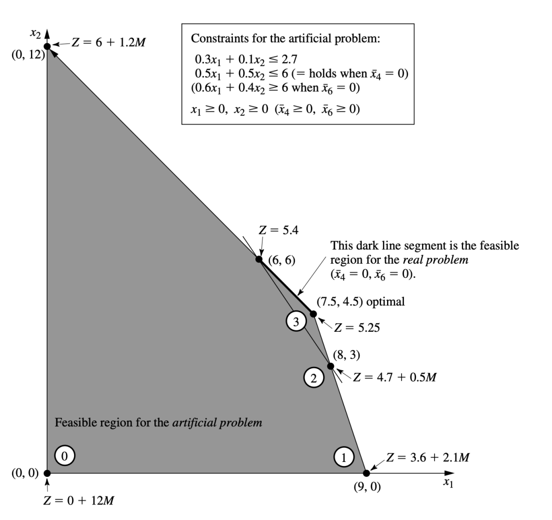
Concept 3 - $(0, 0)$ is not a CPF

$$
\begin{aligned}
\min z = & 0.4x_1 + 0.5x_2 \\
& 0.3x_1 + 0.1x_2 \leq 2.7 \\
& 0.5x_1 + 0.5x_2 = 6 \\
& 0.6x_1 + 0.4x_2 \geq 6 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Concept 3 - $(0, 0)$ is not a CPF



Concept 3 - $(0, 0)$ is not a CPF



The choice of a better CPF soution at each iteration

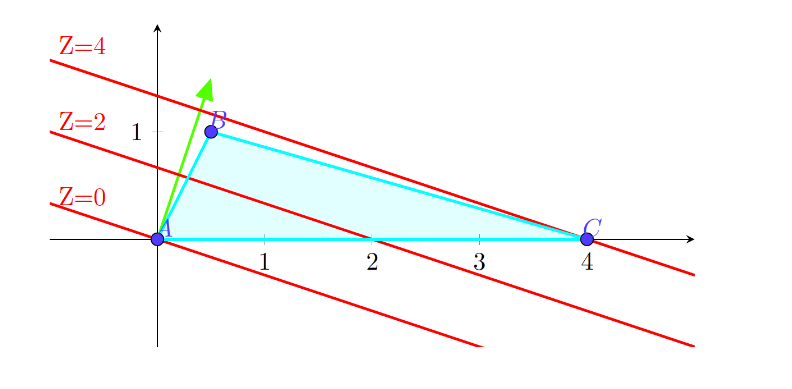
Given a CPF solution, it is much quicker computationally to gather information about its adjacent CPF solutions than about other CPF solutions. Therefore, each time the simplex method performs an iteration to move from the current CPF solution to a better one, it always chooses a CPF solution that is adjacent to the current one. No other CPF solutions are considered. Consequently, the entire path followed to eventually reach an optimal solution is along the edges of the feasible region.

Which adjacent CFP solution to choose at each iteration

After the current CPF solution is identified, the simplex method examines each of the edges of the feasible region that emanate from this CPF solution and identifies the rate of improvement in Z that would be obtained by moving along the edge. Among the edges with a positive rate of improvement in Z, it then chooses to move along the one with the largest rate of improvement in Z. The iteration is completed by first solving for the adjacent CPF solution at the other end of this one edge and then relabelling this adjacent CPF solution as the current CPF solution for the optimality test and (if needed) the next iteration.

At the first iteration of the example, moving from $(0, 0)$ along the edge on the x_1 axis would give a rate of improvement in Z of 3 (Z increases by 3 per unit increase in x_1), whereas moving along the edge on the x_2 axis would give a rate of improvement in Z of 5 (Z increases by 5 per unit increase in x_2), so the decision is made to move along the latter edge. At the second iteration, the only edge emanating from $(0, 6)$ that would yield a positive rate of improvement in Z is the edge leading to $(2, 6)$, so the decision is made to move next along this edge.

In this case it is not a good idea to move along the edge with the largest rate of improvement in Z. In fact $Z = x_1 + 3x_2$. If we move along the edge on the x_2 axis we need two iterations ($A \to B \to C$), whereas if we move along the x_1 axis we just need one iteration ($A \to C$).



How the optimality test is performed efficiently

A positive rate of improvement in Z implies that the adjacent CPF solution is better than the current CPF solution, whereas a negative rate of improvement in Z implies that the adjacent CPF solution is worse. Therefore, the optimality test consists simply of checking whether any of the edges give a positive rate of improvement in Z. If none do, then the current CPF solution is optimal.

In the example, moving along either edge from $(2, 6)$ decreases Z. Since we want to maximise Z, this fact immediately gives the conclusion that $(2, 6)$ is optimal.

Variation of Z

CPF $(2, 6)$ is the intersection point between $2x_2 = 12$ (i.e., $x_2 = 6$) and $3x_1 + 2x_2 = 18$. The latter constraint boundary can also be written as $x_1 = 6 - (2/3)x_2$. Hence, when $Z = 3x_1 + 5x_2$ moves along this constraint, we have that $Z = 3 * (6 - 2/3)x_2 + 5x_2$, i.e., $Z = 18 + 3x_2$. In $x_2 = 6, Z = 36$. Hence, if $x_2 < 6$, Z decreases and therefore this is not a viable option. Since also moving along $x_2 = 6$ does not allow to increase Z, it follows that $(2, 6)$ is the optimal solution.

Similarly, if we consider $x_2 = 9 - 3/2x_1$ then $Z = 45 - 9/2x_1$. In $x_1 = 2$ $Z = 36$ and this value decreases as long as x_1 increases.

Check that if $Z = 3x_1 + x_2$ then $(2, 6)$ is not optimal.

Thank you for your attention