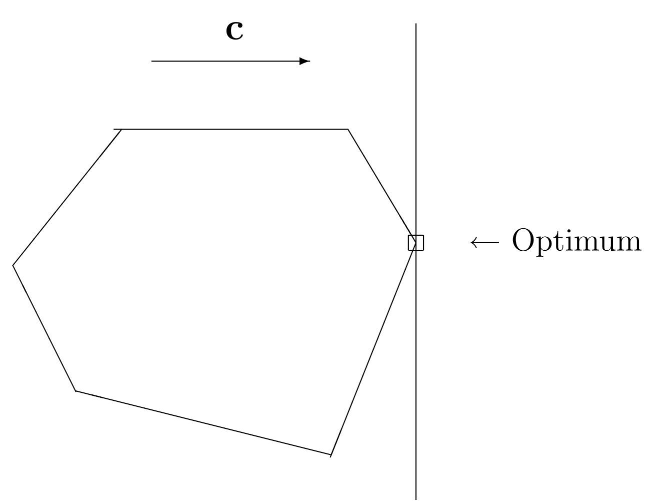
Solving Linear Programs

Geometric Interpretation

The constraints determine the set of feasible solutions. This is a *polyhedron*, the higher dimensional generalization of a 2-dimensional polygon.

Finding the maximum of a linear objective function of the form $Z = \mathbf{cx}$ over this polyhedron essentially means to find a vertex of the polyhedron that is the farthest in the direction determined by the vector $\mathbf{c}$:

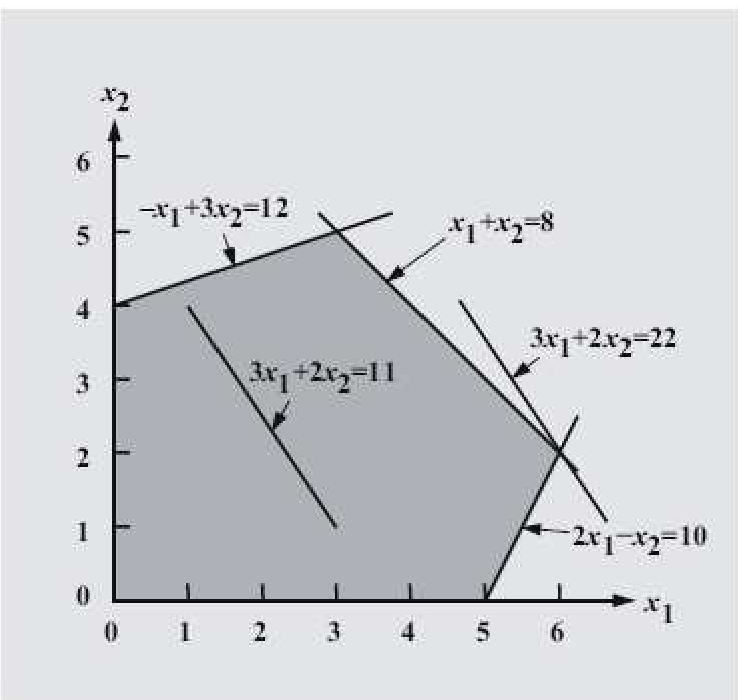


In case there are only two variables, this can be graphically represented and solved in the plane.

To illustrate, consider the following problem:

$$\begin{aligned} \text{maximize} \quad & 3x_1 + 2x_2 \\ \text{subject to} \quad & -x_1 + 3x_2 \le 12 \\ & x_1 + x_2 \le 8 \\ & 2x_1 - x_2 \le 10 \\ & x_1,\ x_2 \ge 0. \end{aligned}$$

Graphical solution: After finding the polygonal boundary of the feasible domain D, as illustrated in the figure below, we "push" the line $3x_1+3x_2 = a$, representing the objective function, as far as possible, so that it still intersects D. The optimum will be attained at a vertex of the polygon.



If, however, as typical in applications, there are many variables, this simple graphical approach does not work, one needs more systematic methods.

Comments on LP Algorithms

Finding the optimal solution in Linear Programming takes relatively complex algorithms. Studying the details of LP algorithms is beyond the scope of this course, since in most cases the network designer can apply off-the-shelf commercial software. A lot of freeware is also available on the Internet.

Some historical points about solving linear programs:

- The first and most widely used LP algorithm has been the Simplex method of Dantzig, published in 1951. The key idea of the method is to explore the vertices of the polyhedron, moving along edges, until an optimal vertex is reached.
- There are many variants of the Simplex Method, and they usually work fast in practice. In pathological worst cases, however, they may take exponential running time.
- It was a long standing open problem whether linear programming could be solved by a polynomial-time algorithm at all, in the worst case. The two most important discoveries in this area were the following:
  - The first polynomial-time LP algorithm was published by Khachiyan in 1979. This result was considered a

theoretical breakthrough, but was not very practical.

  - A practically better algorithm was found by Karmarkar in 1984. This is a so called interior point method that starts from a point in the polyhedron and proceeds towards the optimum in a step-by-step descent fashion. Later many variants, improvements and implementations were elaborated, and now it has similar practical peformance as the Simplex Method, while guaranteeing polynomially bounded worst-case running time.

- It is interesting that, after more than a half century, a major problem is still open in the world of LP algorithms: does there exist an algorithm that solves the LP, such that the worst-case running time is bounded by a polynomial in terms of the number of variables and constraints only, *independently* of how large are the numbers that occur in them? (Counting elementary arithmetic operations as single steps.) Such an algorithm is called a *strongly polynomial time* algorithm. Khachiyan's and Karmarkar's methods do not have this feature.

- LP solvers of different sorts are available as commercial software, or even as freeware. Thus, the network designer typically does not have to develop his/her own LP solver. Once a problem is formulated as a linear programming task, off-the-shelf software can readily be used.