Linear Programming

Éric Schost

November 6, 2007

1 What to read

My main reference was [1]; linear programming is explained in Chapter 29. The discussion of duality for flow algorithms is from Chapter 12 of [2].

2 Problems

1. Give all the steps of the simplex algorithm applied to the following problem: maximize $x + 3y$ with the constraints $x \geq 0$, $y \geq 0$, $x + 3y \leq 3$, $2x - y \leq 1$. You can use $x = 0, y = 0$ as a starting point.

2. Give a clean proof that you can always turn a linear problem (with any kind of equalities and inequalities) into one in *slack form*, where all inequalities are of the form $x_i \geq 0$, and that solving the optimization problem for the system in slack form enables you to solve the original problem.

3. (not required for the assignment!) Same question, for the standard form, where you have no equality, only inequalities of the form $a_{i,1}x_1 + \cdots + a_{i,n}x_n \leq b_i$ and $x_i \geq 0$, and you have to solve a maximization problem.

4. A company sells wooden boards. The boards are first produced as pieces of length 100; however, the customers's orders are

   - 85 boards of length 44 each,
   - 75 boards of length 36 each,
   - 121 boards of length 31 each,
   - 151 boards of length 17.

   The company can subdivide the length-100 boards to satisfy the customers's needs (e.g., you can get 3 boards of length 31 out of one board of length 100), and wants to do this with a minimal loss.

Show that there are 12 ways to subdivide a board of length 100 into boards of lengths {44,36,31,17} (without leaving a usable leftover).

Rewrite the optimization problem as a linear problem in 12 variables, where you look for an optimal solution with integer coefficients.

Give an optimal solution obtained with the simplex algorithm (you can find applets online, google "simplex applet"). You may find an optimal with non-integer solutions. In that case, what can you do with it?

5. (not required for the assignment!) We consider a non-oriented graph G with n vertices, where all pairs of vertices are connected. The length between two vertices i and j is $c_{i,j}$. How many edges are there?

   A *cycle* C is a sequence of vertices $C = (i_1, i_2, \ldots, i_\ell = i_1)$, with $\ell > 2$, and where $i_r \neq i_1$ for $r = 2, \ldots, \ell - 1$ (this means that there is no "self-intersection"). A family of cycles $C_1, \ldots, C_k$ *covers* G if every vertex of G belongs to one, and only one, of these cycles. For $n = 4$, give the list of all cycles.

   We want to find a family of cycles that covers G and has a minimal total length. We will write this as a linear problem in some variables $x_{i,j}$ $(i \neq j)$, where (as in the previous problem) we look for integer solutions.

   The constraints are

$$\sum_{i=1}^{n} x_{i,j} = 1 \text{ (for each } j\text{)}, \quad \sum_{j=1}^{n} x_{i,j} = 1 \text{ (for each } i\text{)}, \quad x_{i,j} \geq 0 \text{ for all } i, j.$$

   Prove that whenever these constraints are satisfied, assuming that all $x_{i,j}$ are integers:

   - for all i, there is a unique j such that $x_{i,j} = 1$ (and all other ones are 0);
   - for all j, there is a unique i such that $x_{i,j} = 1$ (and all other ones are 0).

   Using these facts, explain how we can find a family of cycles that covers G from the values $x_{i,j}$. What objective function do we want to optimize?

References

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms (Second Edition)*. MIT Press, 2001.

[2] V. Vazirani. *Approximation algorithms*. Springer, 2001.