College of Computer Science
Northeastern University
COM 3391: Advanced Algorithms

Lecture 5, part 2
27 October 1999
Scribe: Mohsen D. Ghassemi

Introduction To Linear Programming

1 The Diet Problem

Many linear programming formulations arise from situations where a decision maker wants to minimize the cost of meeting a set of requirements. In the diet problem we would like to develop a diet using n food items such that it satisfies the daily vitamin requirements. Let the food items be numbered 1 through n and let the m vitamin mineral requirements be given by $b_1, \ldots, b_m$.

From food item j, you get a_{ij} units of mineral i per unit of j. If you decide to have x_j units of item j, you get $x_j a_{ij}$ units of mineral i.

$$\sum_j x_j a_{ij} \geq b_i \quad \forall i$$

If the cost of food item j is c_j per unit. Then the total cost incurred $= \sum_j c_j x_j$.

We would like to minimize $\sum_j c_j\ x_j$
subject to:

$$\sum_{j=1}^{n} x_i a_{i,j} \ \geq \ b_i \quad i = 1,..,m$$

Alternatively, one can write the above optimization problem as:

$$minimize \begin{bmatrix} c_1 & c_2 \ldots c_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$subject\ to: \begin{bmatrix} a_{11} & a_{12} \ldots a_{1n} \\ a_{21} & a_{22} \ldots a_{2n} \\ a_{m1} & a_{m2} \ldots a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \geq \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

The above problem is of the form min $c^T x$ subject to: $Ax \geq b$. Such that a problem is referenced to as a *linear program.* This is because the objective function as well as the constraints are linear combination of the variables.

We now consider some optimization problems that we have studied and see weather we can write them down as a linear program.

1.1 Knapsack Problem

Recall that in the knapsack problem we are given n items of sizes w_1 through w_n and profits p_1 through p_n. The goal is to build the items with total size $\leq B$ such that the total profit is maximized.

The LP representation of the problem can be shown as:

$$\begin{array}{ll} maximize & \sum x_j p_j \\ subject\ to & \sum_{j=1}^{n} x_j w_j \leq B \\ & x_j \in \{0,1\} \end{array}$$

$$c^T = \begin{bmatrix} p_1 \\ \vdots \\ p_n \end{bmatrix}$$

$$A = \begin{bmatrix} w_1 & w_2 \ldots w_n \end{bmatrix}$$

$$b = \begin{bmatrix} B \end{bmatrix}$$

Note that the constraint $x_j \in$ {0,1} is not a linear constraint so that the above program is an integer linear program.

1.2 Minimum Spanning Tree Problem

In the minimum spanning tree roblem, the set of instances is the set of all weighted undirected graphs. For a given instance graph G, the set $S(G)$ of solutions for G is the set of all trees that span every vertex of G. Finally, the value of a solution tree T is simply the sum of the weights of the edges of T.

One LP representation of the problem is:

$$\begin{array}{lll} minimize & \sum_e w_e x_e & \\ subject\ to & \sum_{e\ \in\ c} x_e \geq 1 & for\ all\ cut\ c \\ & x_e = 0\ or\ 1 & for\ all\ edges\ e \end{array}$$

Note that since there are an exponential number of cuts, the number of constraints is exponential.

1.3 Weighted Vertex Cover Problem

Recall that a vertex cover of undirected graph $G = (V.E)$ is a subset $V' \subseteq V$ such that if $(u, v) \in E$, then $u \in V'$ or $v \in V'$ (or both). That is, each vertex covers its incident edges, and a vertex cover for G is a set of vertices that covers all the edges in E.

The LP representation of the problem is:

$$
\begin{array}{lll}
\textit{minimize} & \sum_u x_u w_u & \\
\textit{subject to} & x_u + x_v \geq 1 & \forall (u,v) \in E \\
& x_u, x_v = 0 \ \textit{or} \ 1 &
\end{array}
$$