CS 270 Homework 12

Neel Gupta

April 25, 2023

Problem 1. Suppose you are designing a scheduling algorithm for a manufacturing plant that has a set of M machines and a set of N tasks to be completed. Each task can be assigned to only one machine, and each machine can only perform one task at a time. Let p_{ij} be the processing time task i on machine j. The goal is to minimize the overall time it taks to complete all the tasks. How would you formulate this problem as an Integer Linear Program? What decision variables would you use, and what constraints and objective function would you include?

Answer:

To formulate the scheduling problem as an ILP, we can use the following decision variables:

- Let x_{ij} be a binary variable indicating whether task i is assigned to machine j.
- Let C be the time at which the last task is completed, also known as ″makespan″.

Then we can formulate the problem as follows:

$$\text{minimize } C$$

Subject to:

1. (Task assignment constraint) $\sum_{j=1}^{M} x_{ij} = 1, \forall i$
2. (Machine capacity constraint) $C \geq \sum_{i=1}^{N} p_{ij} x_{ij}, \forall j$
3. (Domain of the decision variables) $x_{ij} \in \{0, 1\}, \forall i, j$

Here, the objective function is to minimize the maximum completion time, and the constraints ensure that each task is assigned to exactly one machine and that the completion time of each task is greater than or equal to the total processing time on the machine it is assigned to.

Problem 2. Formulate the problem of finding a Min S-T cut of a directed network with source s and sink t as an Integer Linear Program and explain your program.

Answer:

$$\text{minimize} \sum_{(u,v)\in E} c(u,v) \cdot x(u,v)$$

Subject to:

1. $x_v - x_u + x(u,v) \geq 0 \quad \forall (u,v) \in E$
2. $x_u \in \{0,1\} \quad \forall u \in V : u \neq s, u \neq t$
3. $x(u,v) \in \{0,1\} \quad \forall (u,v) \in E$
4. $x_s = 1$
5. $x_t = 0$

- The variable x_u is used to determine if vertex u is on the side of source s in the cut. $x_u = 1$ if and only if u is on the side of s.
- By setting $x_s = 1$ and $x_t = 0$, we ensure that source s and sink t are separated.
- The variable $x(u,v)$ is used to determine if edge (u,v) crosses the cut.
- In the minimization objective, $c(u,v)$ represents the capacity of edge (u,v).
- The first constraint makes sure that if u is on the side of s and v is on the side of t, then edge (u,v) must be included in the cut.
- Possible scenarios of x_u and x_v:
  1. If u is on the side of s and v is on the side of t: This implies $x_u = 1$ and $x_v = 0$. Thus, for the first constraint to be valid, $x(u,v)$ must be set to 1, indicating that edge (u,v) is part of the cut.
  2. If both u and v are on the same side, then $x_u = x_v$. In this case, the first constraint is already satisfied and it is not necessary for $x(u,v)$ to be 1. Since the ILP tries to minimize $\sum_{(u,v)\in E} c(u,v) \cdot x(u,v)$, all $x(u,v)$ corresponding to this case will eventually be set to 0.

One can also visualize this program as initially starting with the ″s″ node on one side, and all other nodes on the ″T″ side. This would mean that the initial configuration would have all the edges from the source node ″s″ as part of the cut. As the iterations progress, the program checks if each node should be part of the ″S″ set or not and finally reaches the minimal value.

Problem 3. A set of n space stations need your help in building a radar system to track spaceships traveling between them. The ith space station is located in 3D space coodinates (x_i, y_i, z_i). The space stations never move. Each space station i will ahave a radar with power r_i where r_i is to be determined. You want to figure out how powerful to make each space station's radar transmitter, so that whenever any spaceship travels in a straight line from one station to another, it will always be in the radar range f either the first space station (its origin) or the second space station (its destination). A radar with power r is capable of tracking spaceships anywhere in the sphere with radius r centered at itself. Thus, a spaceship is within radar range through its strip from space station i to space station j if every point along the line (x_i, y_i, z_i) to (x_j, y_j, z_j) falls within either the sphere of radius r_i centered at (x_i, y_i, z_i) or the sphere of radius r_j centered at (x_j, y_j, z_j). The cost of each radar transmitter is proportional to its power, and you want to minimize the total cost of all of the radar transmitters. You are given all of the $(x_1, y_1, z_1), ..., (x_n, y_n, z_n)$ values, and your job is to choose values $r_1, ..., r_n$. Express this as a linear program.

Answer:

1. **Variables:** Let r_i = the power of the ith radar transmitter, $i = 1, 2, \ldots, n$
2. **Objective function:** Minimize $\sum_{i=1}^{n} r_i$
3. **Constraints:** $r_i + r_j \geq \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$ or $r_i + r_j \geq d_{i,j}$ for each pair of stations i and j, where $d_{i,j}$ is the distance from station i to station j. We need $\sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}$ constraints of inequality due to the number of unique paths between each pair of space stations.

Problem 4. Recall the maximum-bipartite-matching problem. Write a linear program that solves this problem gven a bipartite graph $G = (V, E)$, where the set of vertices on the left is L, and the set on the right is R. ($L \cup R = V$)

Answer:

We can solve the maximum-bipartite-matching problem by viewing it as a fully connected flow network. We can append a source s and a sink t and connect every node in L to s and every node in R to t. We will give every edge connecting these a capacity of 1.

Let the set of vertices of the new network flow graph be $\mathcal{V}' = s \cup L \cup R \cup t$. Therefore, the maximum flow of this new network flow directly corresponds with the maximum bipartite matchines. The linear programming problem is as follows:

$$\text{maximize} \sum_{v \in L} f(s, v)$$

Subject to:

1. Flow constraints on edges. $f(u, v) \leq 1, \forall u, v \in \mathcal{V}'$
2. Flow constraints on verticles. $\sum_{v \in \mathcal{V}'} f(u, v) = \sum_{v \in \mathcal{V}'} f(v, u), \forall u \in \mathcal{V}'$
3. Positive flow constraints. $f(u, v) \geq 0, \forall u, v \in \mathcal{V}'$