CS 4820, Spring 2017 Homework 6, Problem 2

Name: Yuxiang Peng

NetId: yp344

Collaborator: jl3455, zl542

2. We can construct a specific network flow graph related to the constraints and using Ford-Fulkerson (we will use FF in short for the following paragraphs) to solve this problem.

The vertices for the graph:

source s, sink t

professor nodes Pi (i = 1…p)

transitive relation nodes to show a specific professor belongs to a department and serve on a committee Tij (i = 1…q, j = 1…d)

Committee nodes Ci (i = 1…q)

For the graph, edges(s, Pi) have capacity of c. The flow on this edge shows how many committees this professor i serves. Edges(Pi, Tjk) have a capacity of 1. Each edge shows this professor i belonging to department k and serve the committee j. To satisfy the constraint 4, each j should be in the Li. Edges(Tjk, Ci) shows the auto-built edges based on the committee value and each edge has a capacity of 1. Edges(Ci, t) have capacities related to the specific ri.

Find the maximum flow in this graph by using FF. If there exist an edge(Ci, t) has a flow less than ri. Then it is impossible to staff each committee. Otherwise, assign professors to the committees and record the assignment of professors to committees.

Running time:

The number of vertices in this graph is 2+p+qd+q = O(p+qd). Because of the constraint 3, the number of edges is at most O(pq). The running time for FF is O(mV) where m is the number of edges and V is the maximum flow value. Here we can find the min cut is {s,Pi}and {Tjk, Ci, t}. The bound of maximum flow is pq. So the running time for FF is O((pq)^2). The construction time for the graph is at most the number of all the vertices and edges which is far less than the running time of FF. So the final running time for this algorithm is O((pq)^2).

Proof:

To satisfy the constraint 2, the capacity for each edge(s, Pi) is c, which shows no professor is allowed to serve on more than c committees. There is a list Lj for the construction of edges(Pi, Tjk). We assign the bounded case for these edges is all the professors belonging to its own department and may serve for any committee which is connected to the requirement that each professor belongs to only one department. It guarantees the professor serve the qualified committee. As we mentioned in the running time part, the maximum edges O(pq) and the nodes Tjk (j refers to the committee and k refers to the department) shows that no committee is allowed to have more than one professor from the same department which is the constraint 3. So the capacity for each edge is 1 and the value of either 0 or 1 of the flow displays whether the professor serve this committee. The edges of (Tjk, Ci) are auto-connected based on the committee value. At last we only need to check the value of each edge from Ci to t because we need to satisfy the constraint 1. If there exist an edge(Ci, t) has a flow less than ri. Then it is impossible to staff each committee which is related to the constraint 1. So the graph satisfies all the constraints and once we apply the maximum flow algorithm FF, we will get the valid assignment.