CSCI 570 - Fall 2019 - HW 9

Junzhe Liu, 2270250947

November 7, 2019

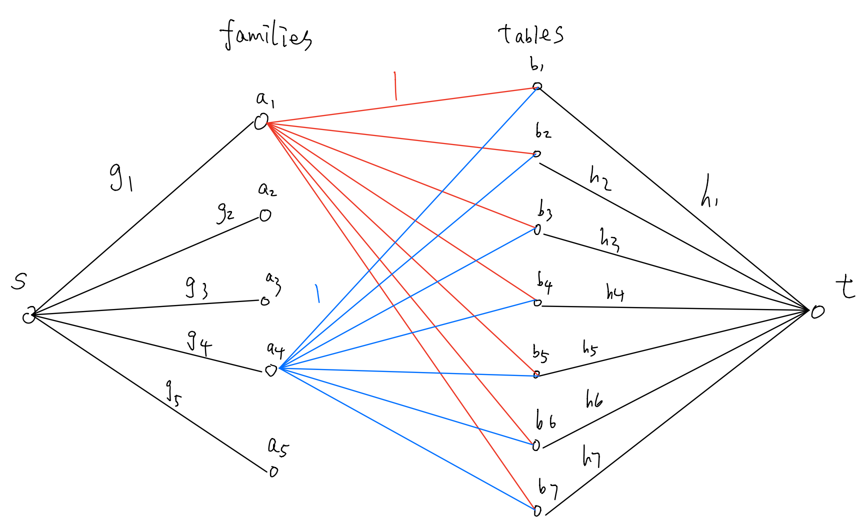
1 At a dinner party, there are n families $a_1, ..., a_n$ and m tables $b_1, ..., b_m$. The i-th family a_i has g_i members and the j-th table b_j has h_j seats. Everyone is interested in making new friends and the dinner party planner wants to seat people such that no two members of the same family are seated in the same table. Design an algorithm that decides if there exists a seating assignment such that everyone is seated and no two members of the same family are seated at the same table.

First of all, if:

$$\sum_i g_i > \sum_j h_j \tag{1}$$

Then there's no enough seats to accommodate everybody. Secondly, if there exist a family a_i such that $g_i > m$, then there's not enough tables for this family. According to Pigeonhole principle, the seating assignment does not exist.

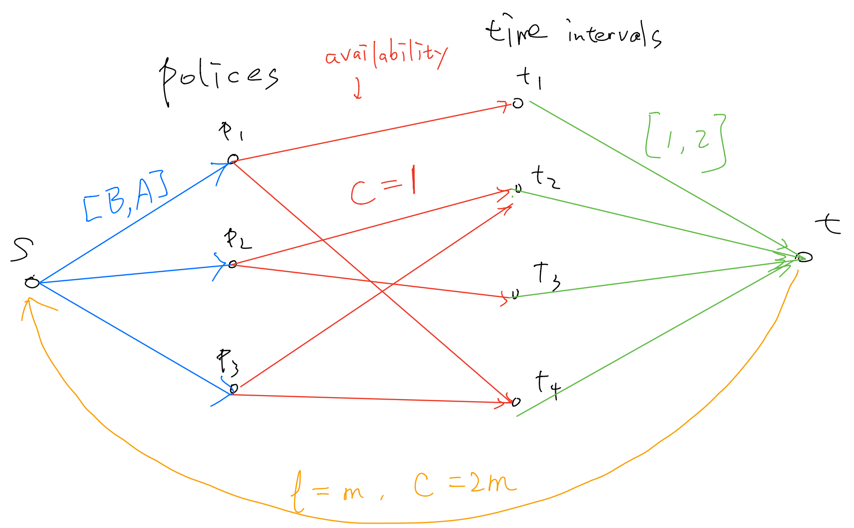
Finally, we build such graph:



where a_i and b_j are fully-connected(meaning there is an edge between all $a_i - b_j$ pair, we only show two examples in the image: the red and blue lines), and their weights are all 1. The edges going out from source to the families a_i have the weight of g_i, and the weights of edges going from tables to the sink t equal to h_j. We run the max-flow algorithm on this graph, if $v(f) < \sum_i g_i$, then the seating assignment does not exist; Otherwise, $v(f) \geq \sum_i g_i$, then such assignment do exist.

2 There is a precious diamond that is on display in a museum at m disjoint time intervals. There are n security guards who can be deployed to protect the precious diamond. Each guard has a list of intervals for which he/she is available to be deployed. Each guard can be deployed to at most A time slots and has to be deployed to at least B time slots. Design an algorithm that decides if there is a deployment of guards to intervals such that each interval has either exactly one or exactly two guards deployed.

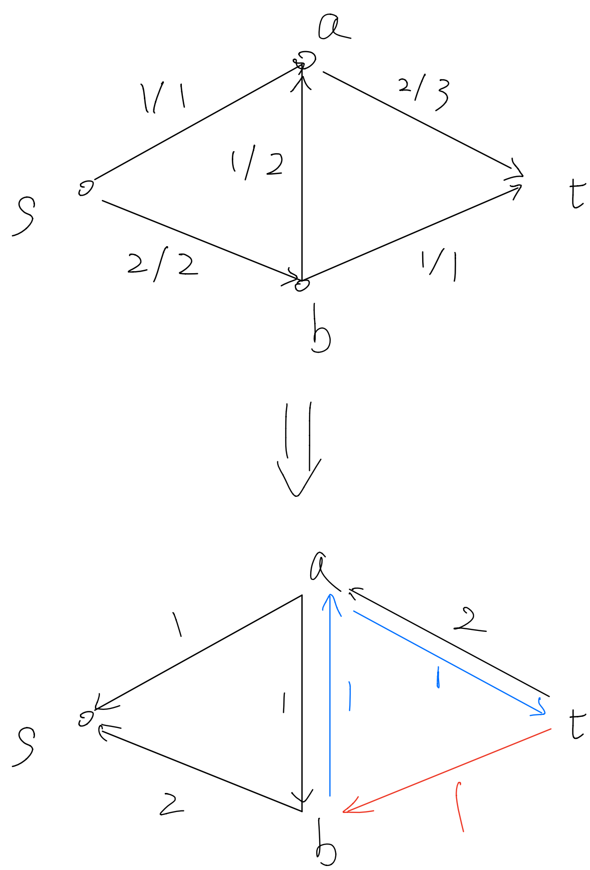
We construct such graph:



In the above graph, $n = 3$ and $m = 4$, polices connect with time intervals according to their time availability: Given the time list of p_i, if t_j is in accordance with this time list, then there's an edge between p_i and t_j; The edge weights between polices and time intervals are all 1, meaning the policy can devote 1 unit of time slot in this interval. The edges between the source and the policy have the weight lower bound of B and a capacity of A, this is exactly the time constraint of each policeman. And the weight of the edges connecting the time intervals and the sink has a lower bound of 1 and a capacity of 2, restricting the deployed police number to be exactly 1 or 2. Furthermore, there's an edge goes from sink t back to source s with a lower bound of m and a capacity of $2m$. Use max-flow to find a possible circulation in this circulation graph.

3 An edge of a flow network G is called critical if decreasing the capacity of this edge results in a decrease in the maximum flow. Is it true that with respect to a maximum flow of G, any edge whose flow is equal to its capacity is a critical edge? Give an efficient algorithm that finds a critical edge in a flow network.

A critical edge must be an edge whose flow equals to its capacity, but reversely, it's not necessary. Consider the following flow graph:



Here the edge $< b, t >$ has a flow equals to its capacity, but it's not a critical edge, because if we reduce 1 unit of flow from it, we can re-routed this 1 unit flow via $b \to a \to t$ and compensate the lost. Therefore, for a saturated edge $< u, v >$ in the original graph, if there exist a path from u to v with available capacity(except the direct edge), then the edge $< u, v >$ is not a critical edge.

Algorithm: for a given max-flow f, construct its residual graph G_f, examine all saturated edge $< u, v >$ in the graph G_f: (meaning $< u, v >$ does not exist in the original graph G and $< v, u >$ is not an edge in the residual graph G_f). Examine: If there's a path routing from u to v in the residual graph G_f, then this edge $< u, v >$ is not a critical edge, otherwise it is.