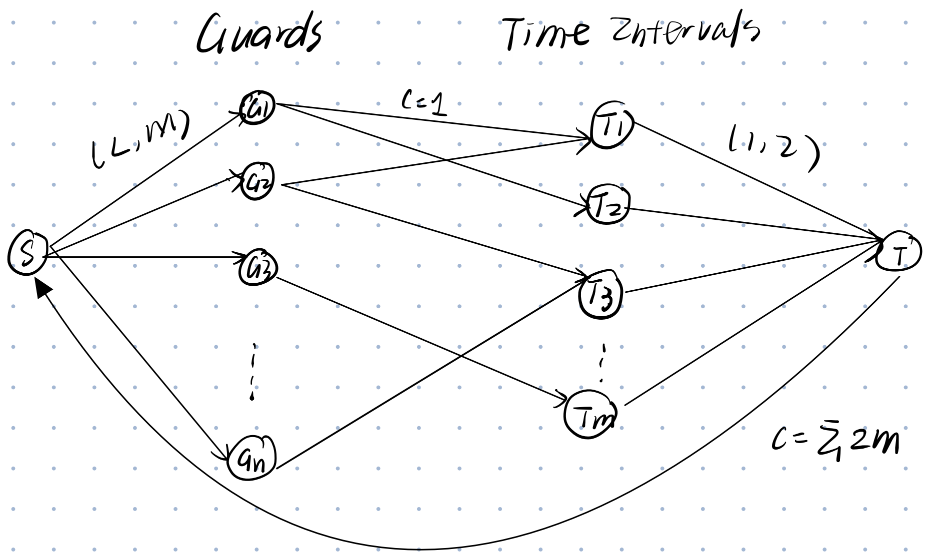
1)

Reduce this problem to Circulation with lower bounds.

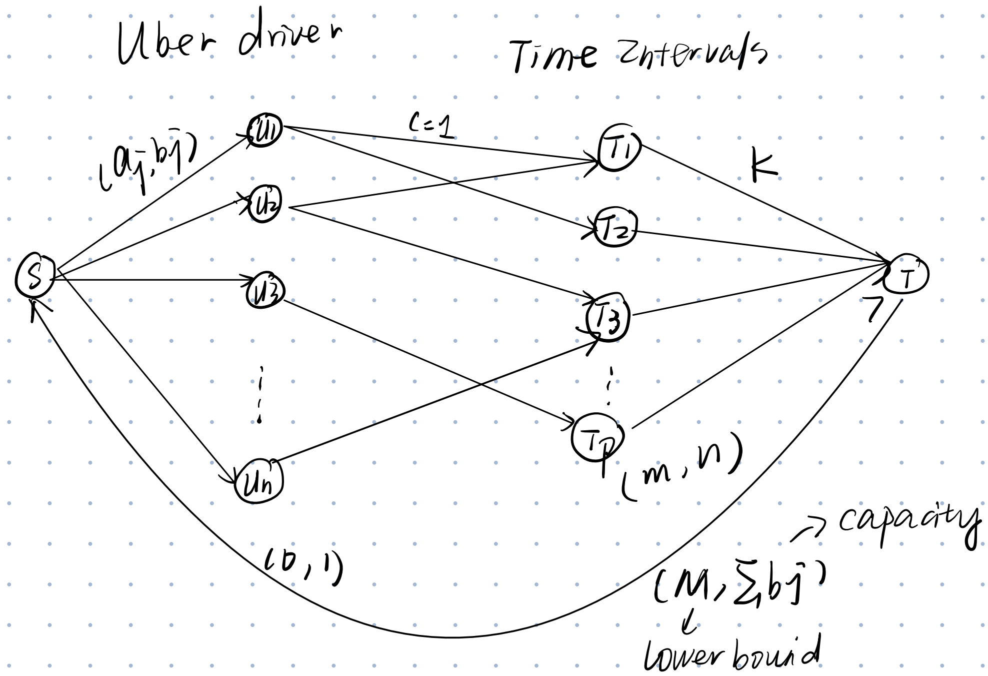
We will construct a circulation network G such that G can have a feasible flow iff we can schedule guards with all given constraints.



We will then solve the feasible circulation problem. If there is a feasible circulation in G, then we can find a feasible assignment of guards to time intervals. Otherwise, we will not be a feasible assignment.

2) Reduce this problem to Circulation with lower bounds.

We will construct a circulation network G such that G can have a feasible flow iff we can schedule Ubers with all given constraints.



If a feasible circulation exists in the network, then there is a valid Uber driver schedule that meets all the constraints, including the minimum number of drivers per day.

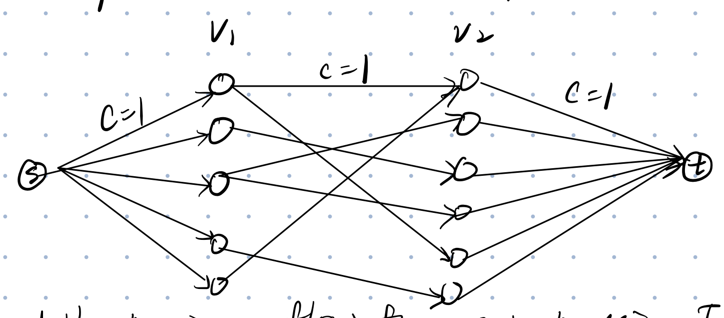
3)

We can reduce the problem to Maximum Bipartite Matching problem, which can be solved by a network flow algorithm.

Let's create a bipartite graph $G = (V_1, V_2, E)$

- For each S cell in the grid, create a node in V_1.
- For each Y cell in the grid, create a node in V_2.
- For each P cell in the grid, create two nodes p_in and p_out. Add p_in to V_1 and p_out to V_2.
- For each S node, add an edge to the p_out node of each neighbouring P cell.
- For each p_in node, add an edge to its corresponding p_out node.
- For each p_in node, add an edge to the Y node of each corresponding Y cell.

Now, we have a bipartite graph where every path from a node in V_1 to a node in V_2 represents a possible "SPY" in the grid. To find the maximum number of disjoint SPYs, we can run a Maximum Bipartite Matching algorithm on this graph. Now, reduce this problem to a network flow problem.



Now, find the maximum flow from s to t using Ford-Fulkerson algorithm. The value of the maximum flow will be equal to the maximum number of disjoint SPYs in the original grid.