Analysis of Algorithms

V. Adamchik CSCI 570 Fall 2016

Discussion 11 University of Southern California

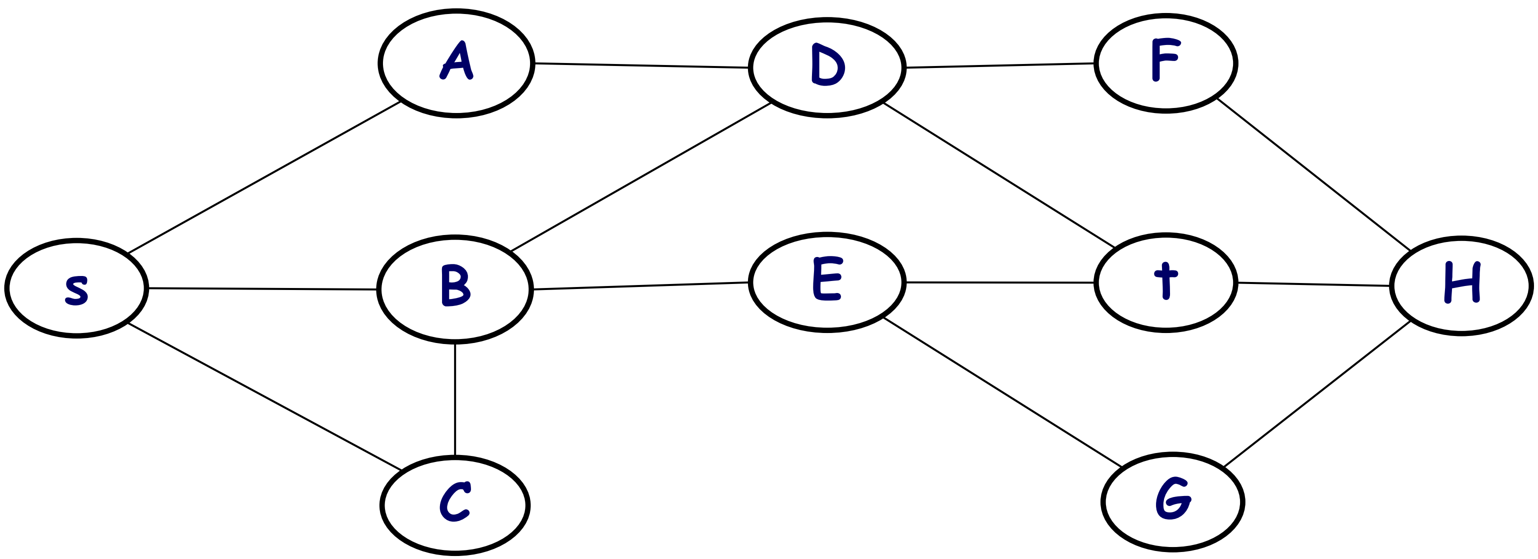
Network Flow-2

Problem 1

Professor Jones has determined that *x* priceless artifacts are located in a labyrinth. The labyrinth can be thought of as a graph, with each edge representing a path and each node an intersection of paths. All of the artifacts are in the same treasure room, which is located at one of the intersections. However, the artifacts are extremely burdensome, so Jones can only carry one artifact at a time. There is only one entrance to the labyrinth, which is also a node in the graph. The entrance serves as the only exit as well. All the paths are protected by human-eating vines, which will be woken up after someone passes the path, so Jones can only go through each path once. Give an algorithm that determines how many artifacts Jones can obtain and how he can do it.

Solution

To obtain a single artifact, Jones would have to find a tour which starts at the entrance, reaches the artifact and leads back again, without getting Jones captured by the vines. If he tries to get another artifact, he must find another tour which does not use the edges he's already traversed.

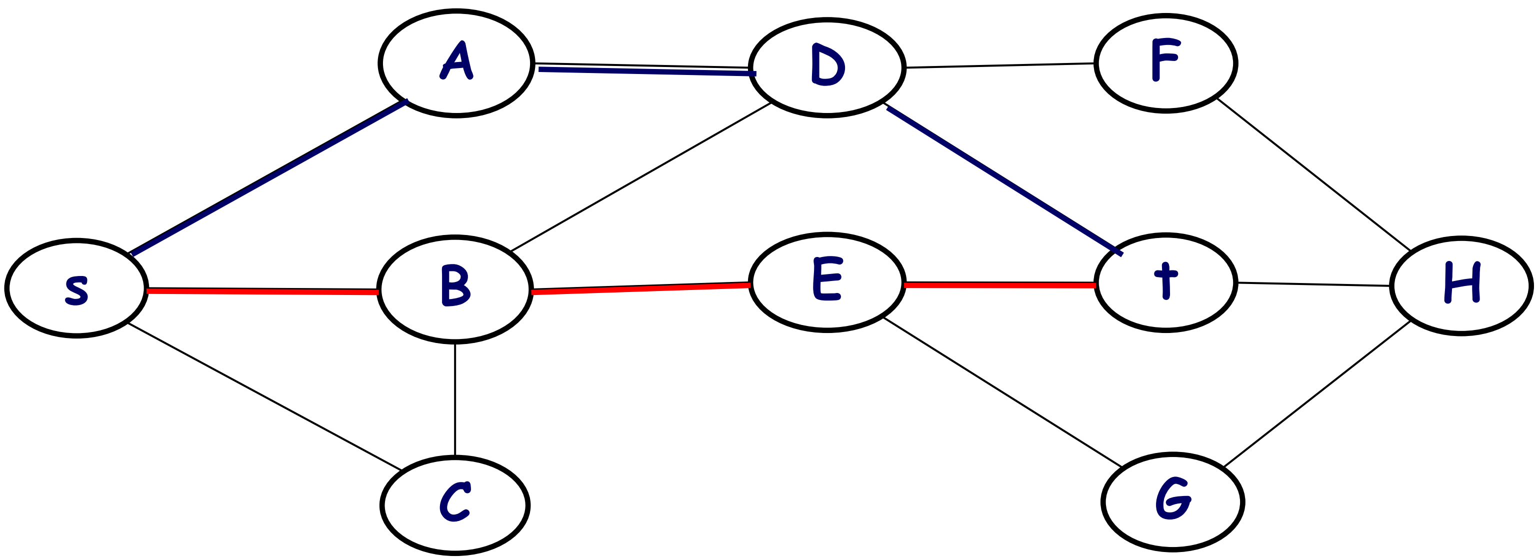


Solution

We need to find how many edge-disjoint paths p exist between s and t.

Then the number of artifacts is twice smaller.

Add capacities, directions and run a max-flow algorithm.



Problem 2

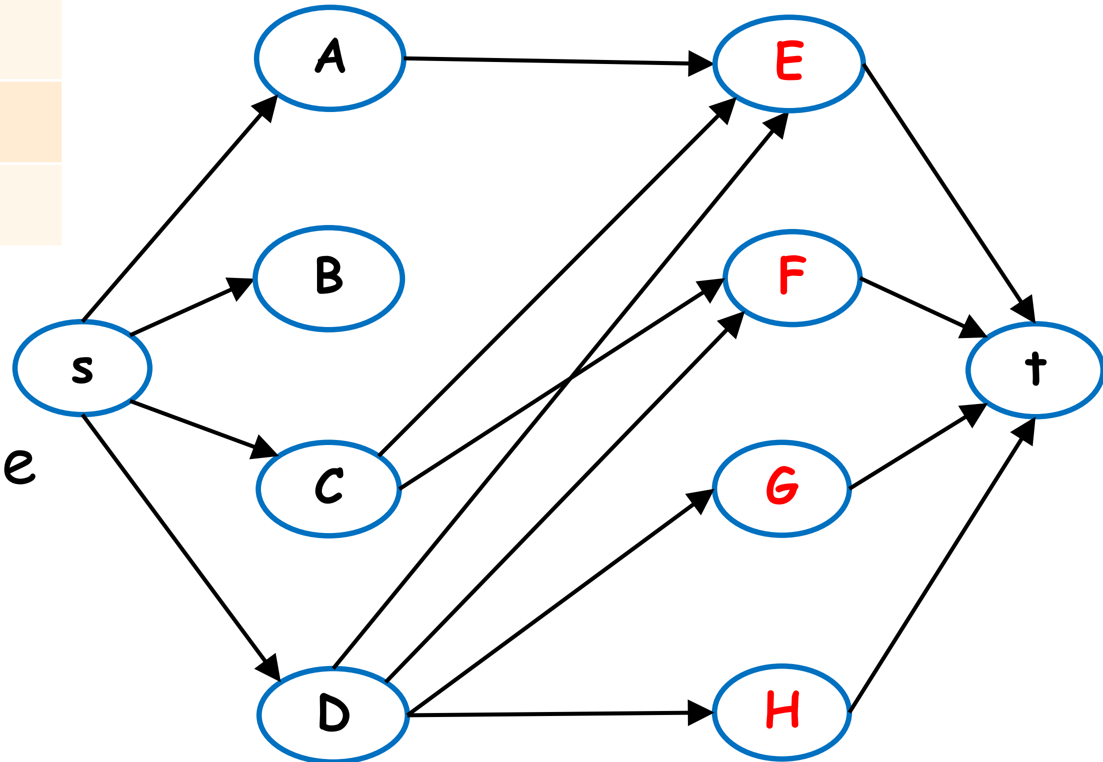
We're asked to help the captain of the USC tennis team to arrange a series of matches against UCLA's team. Both teams have n players; the tennis rating of the i-th member of USC's team is a_i and the tennis rating for the k-th member of UCLA's team is b_k. We would like to set up a competition in which each person plays one match against a player from the opposite school. Our goal is to make as many matches as possible in which the USC player has a higher tennis rating than his or her opponent. Give an algorithm to decide which matches to arrange to achieve this objective.

Solution

Player	Rating	Team
A	10	Trojans
B	5	Trojans
C	15	Trojans
D	20	Trojans
E	7	Bruins
F	14	Bruins
G	16	Bruins
H	19	Bruins

Create a bipartite graph with *n* vertices for Trojan players and *n* vertices for Bruin players. Create source and sink nodes.

Add an edge from Trojans Bruins if the Trojan player has a higher rating than the Bruin player.



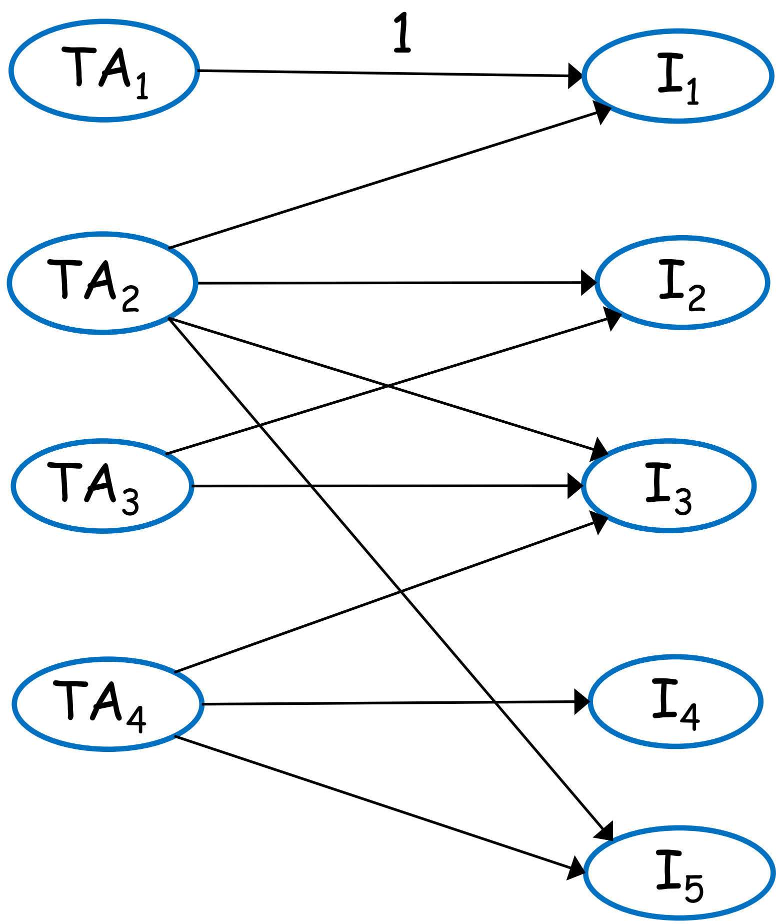
Problem 3

CSCI 570 is a large class with n TAs. Each week TAs must hold office hours in the TA office room. There is a set of hour-long time intervals $I_1, I_2, \ldots I_k$ in which the office room is available. Each TA provides a subset of the time intervals he or she can hold office hours with the minimum requirement of one hour per week. Lastly, the total number of office hours held during the week must be $H>n$. Design an algorithm to determine if there is a valid way to schedule the TA's office hours with respect to these constraints.

Solution

Create a node for each TA and each time interval.

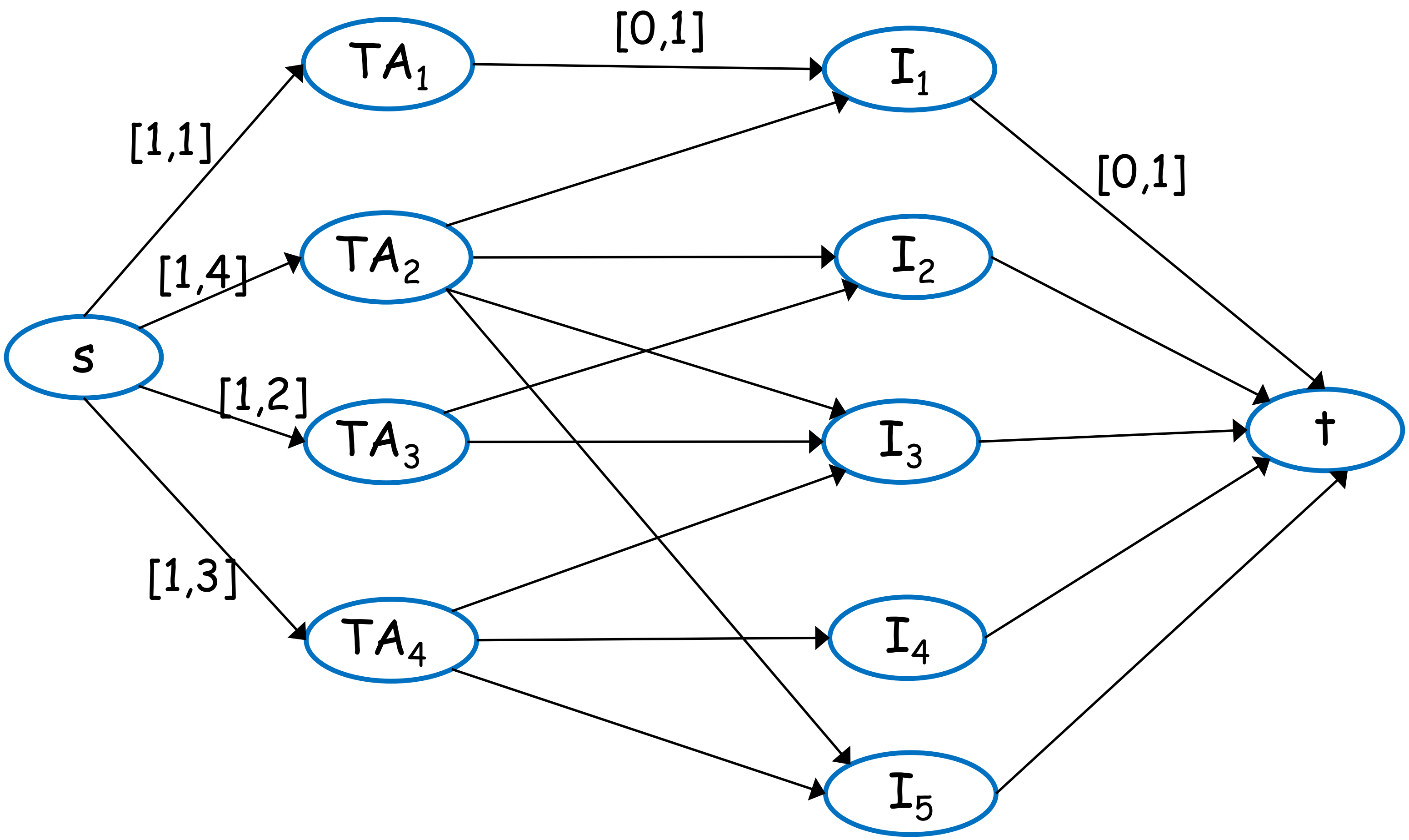
Add an edge between a TA and a time interval node with a capacity of 1 if the TA can cover it.



Solution

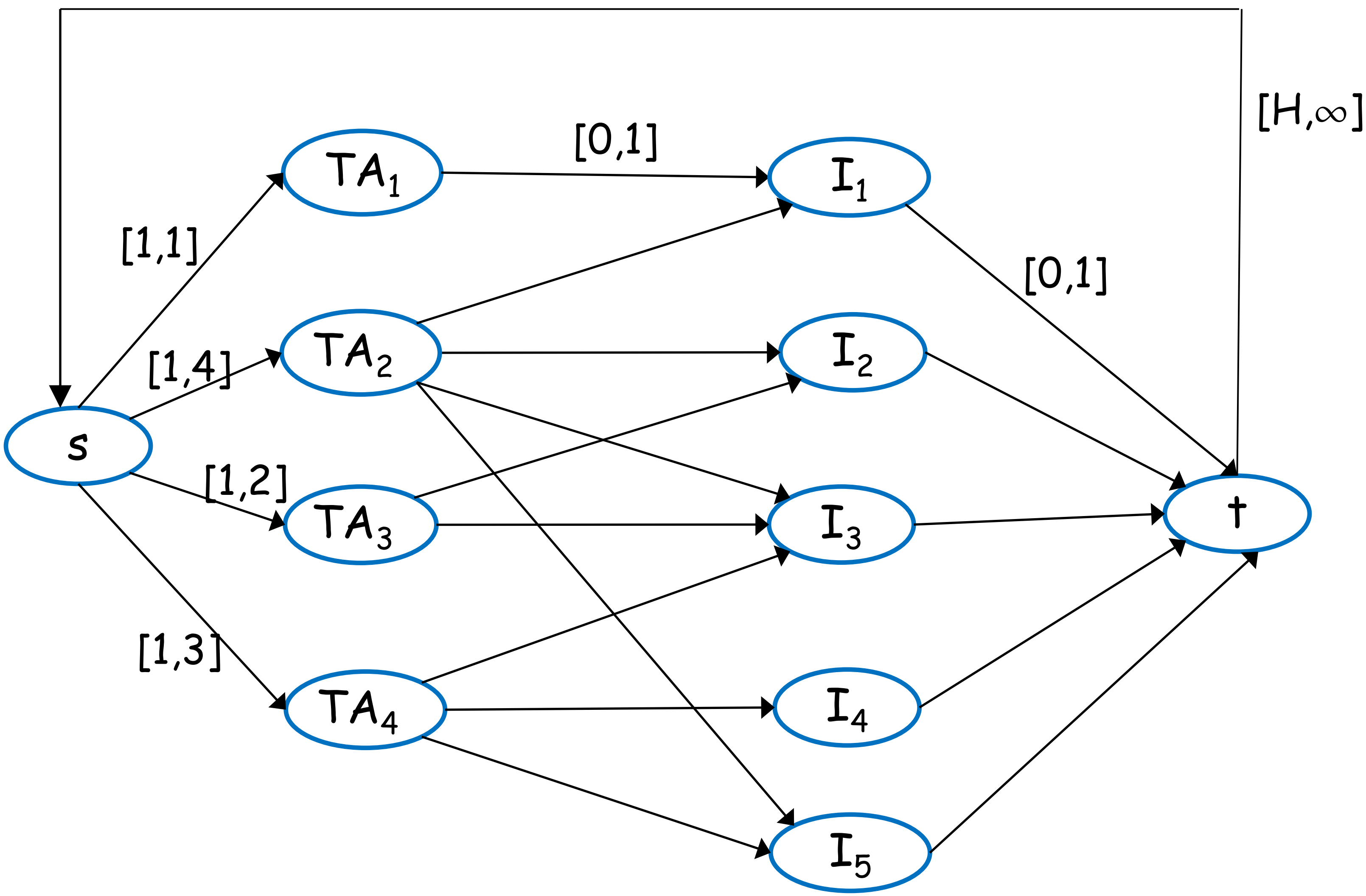
Create a source connected to each TA.

Create a sink node connected each time intervals



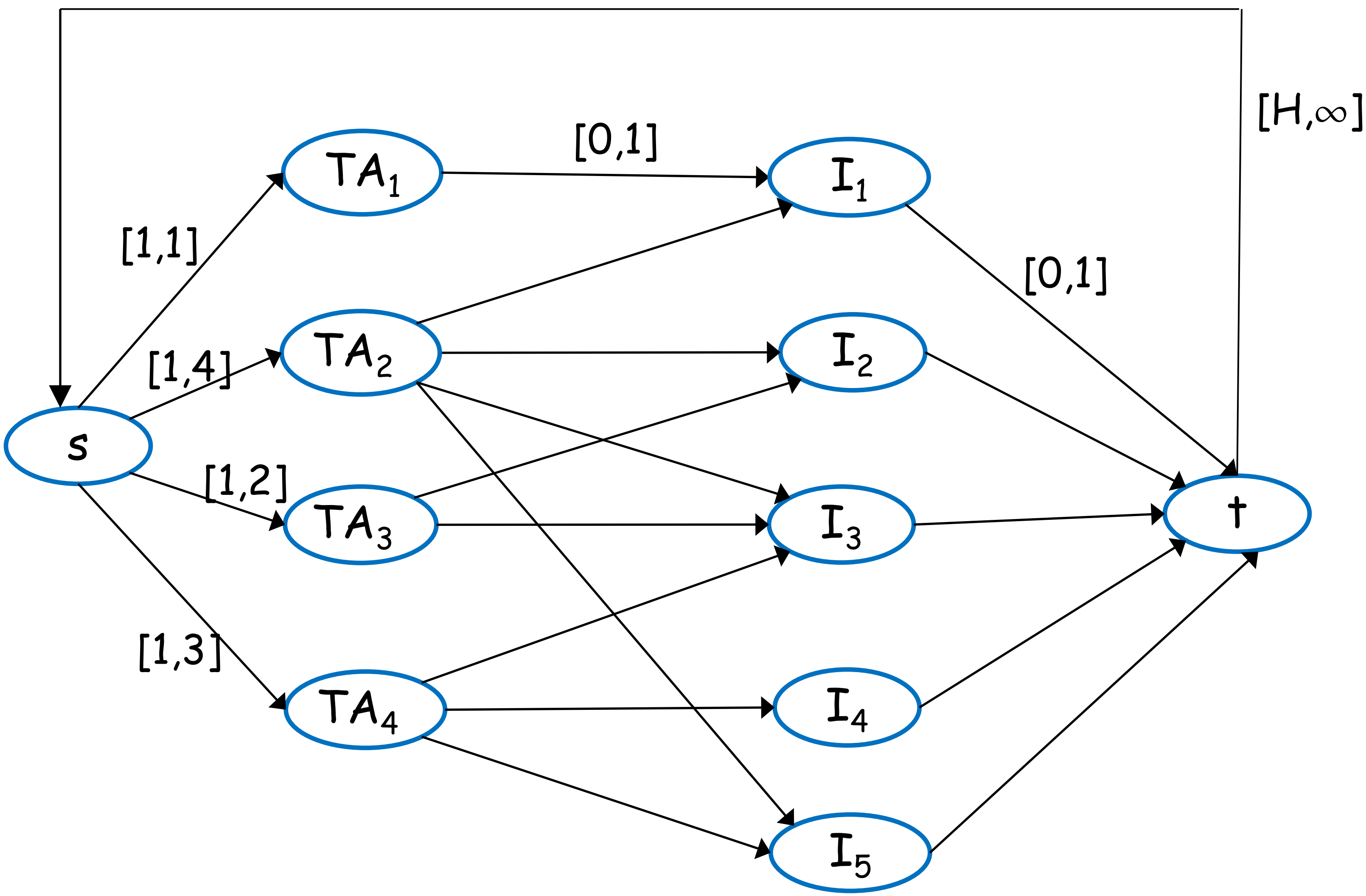
Solution

Connect the sink to the source with minimum capacity of H.



Solution

Does a circulation exist?



Solution

Does a circulation exist?

Step1 : remove lower bounds.

by pushing a flow $f(e) = l(e)$ on each edge.

$L(s)=H-n>0$

$L(TA_i)=1$

$L(I_k)=0$

$L(t)=-H$

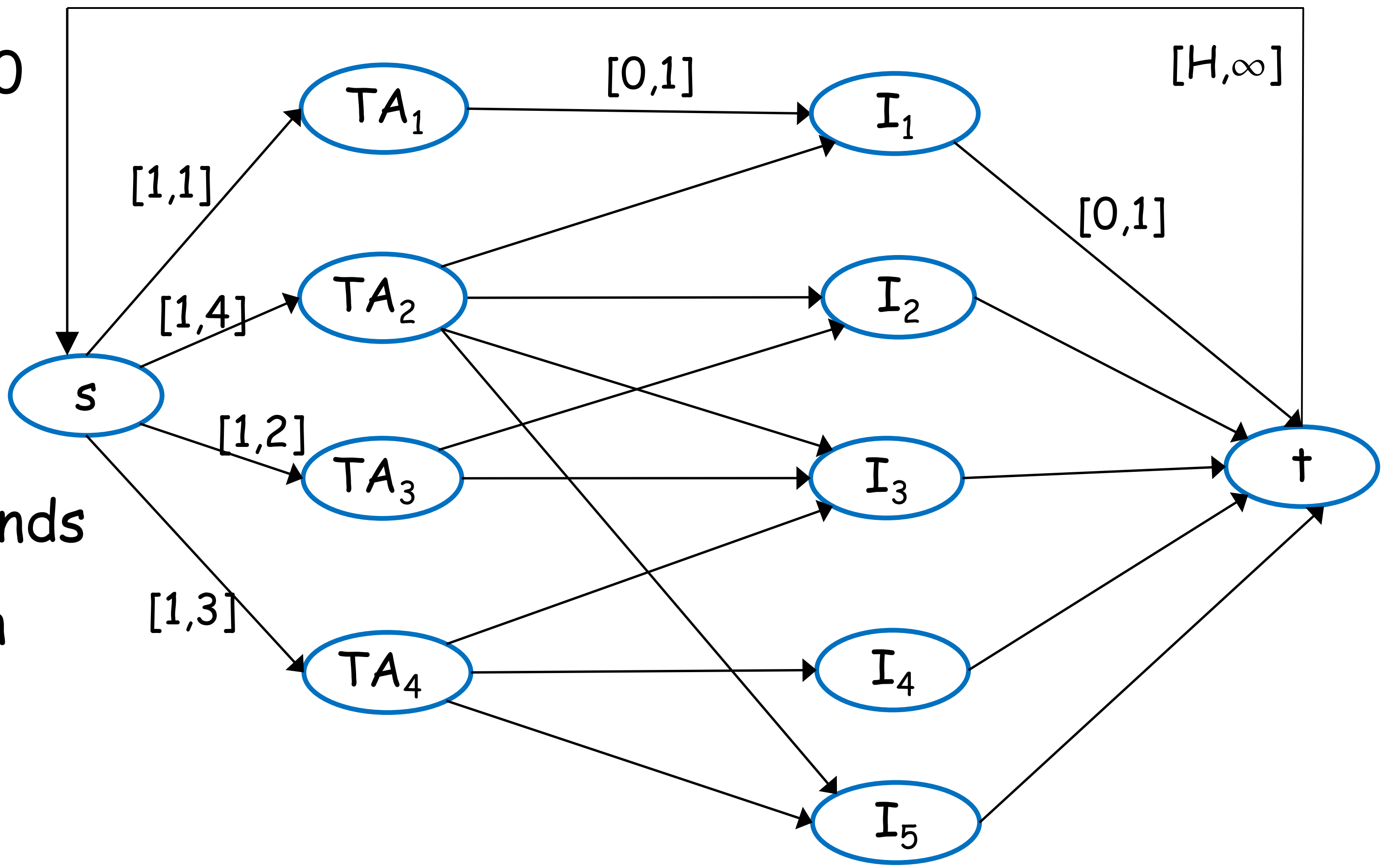
New demands

$d'(s)=-H+n$

$d'(TA_i)=-1$

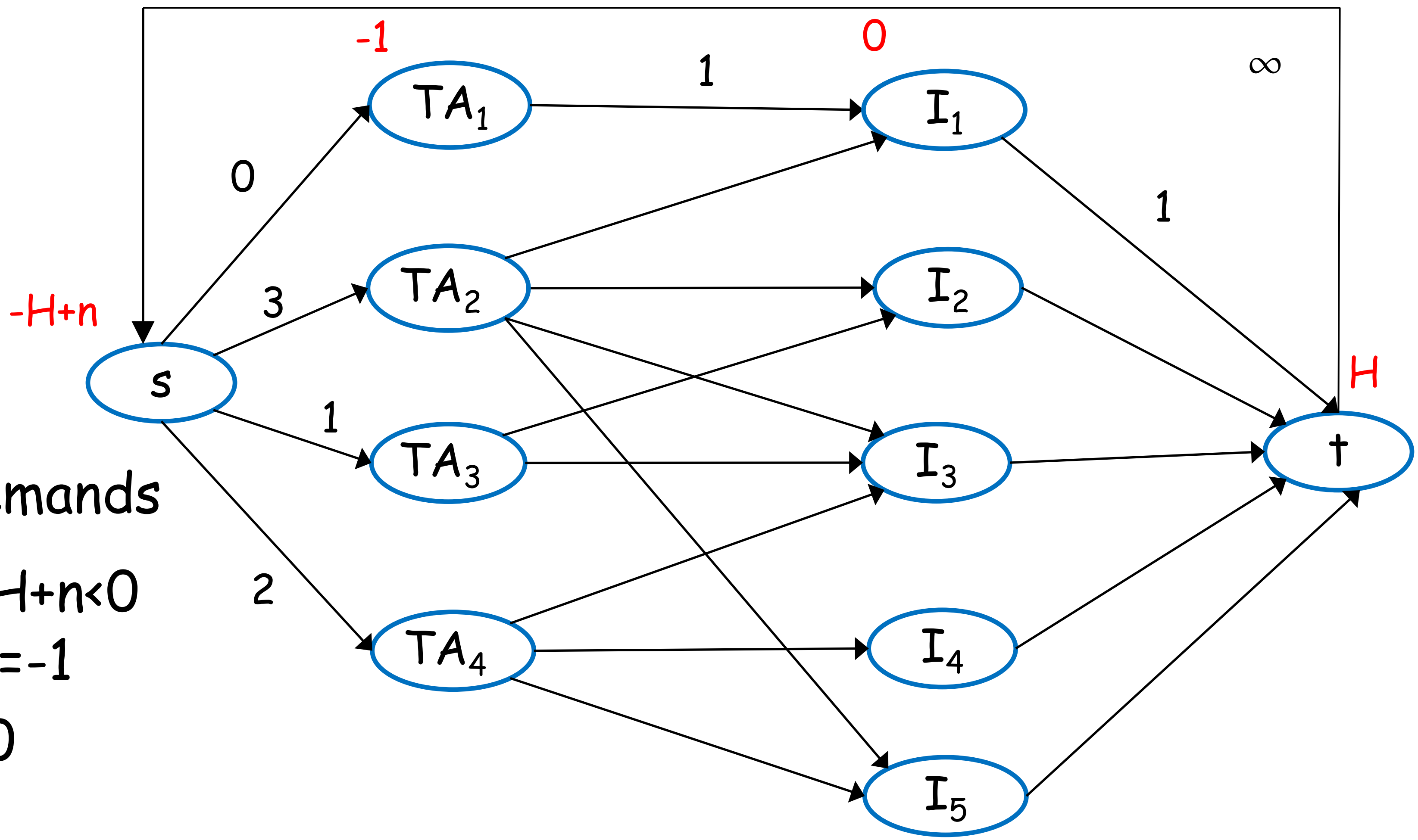
$d'(I_k)=0$

$d'(t)=H$



Solution

Step2 : change demands.



New demands

$d'(s)=-H+n<0$

$d'(TA_i)=-1$

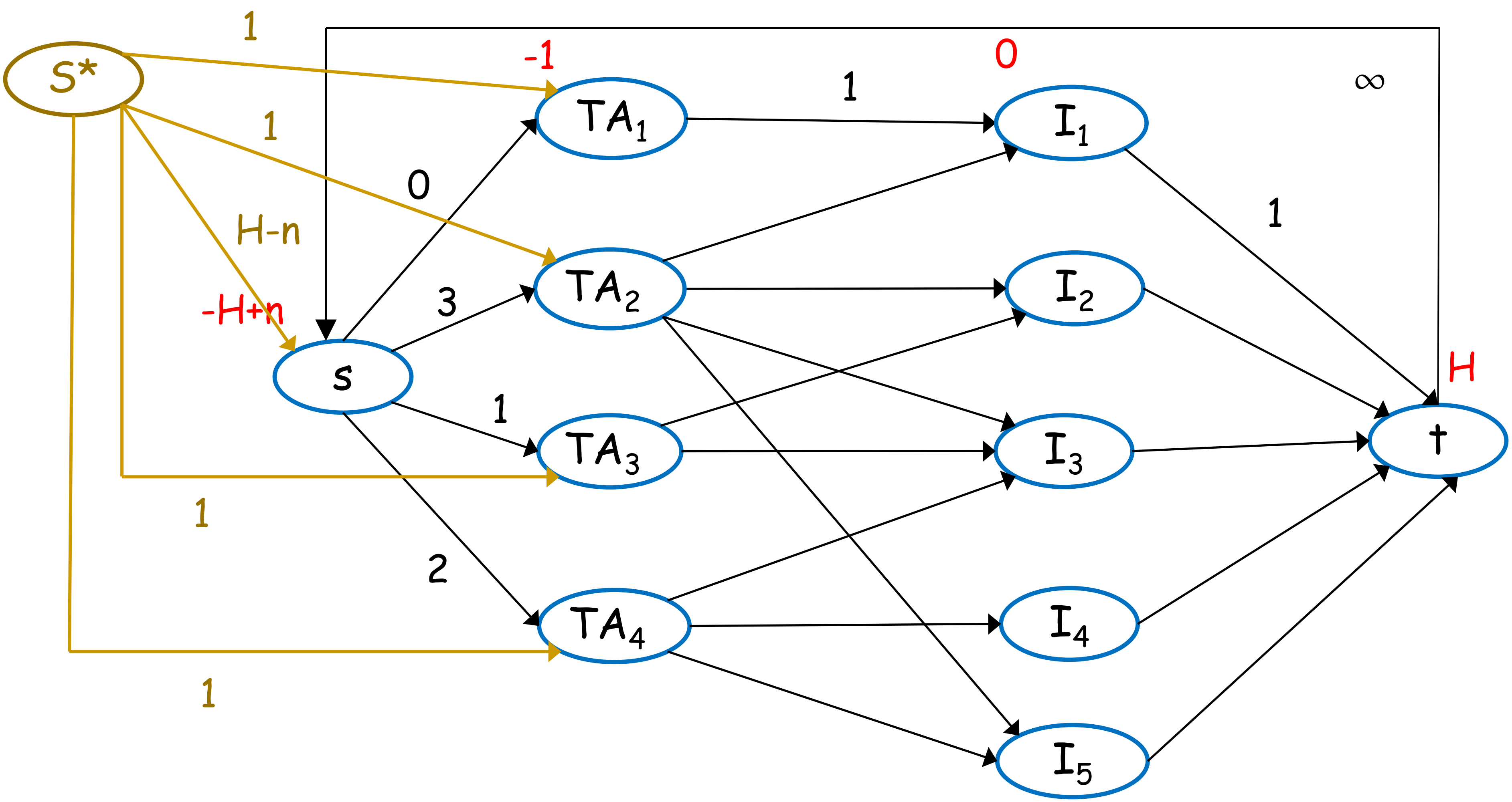
$d'(I_k)=0$

$d'(t)=H$

Solution

Step3 : add new source and connect it to the negative demands

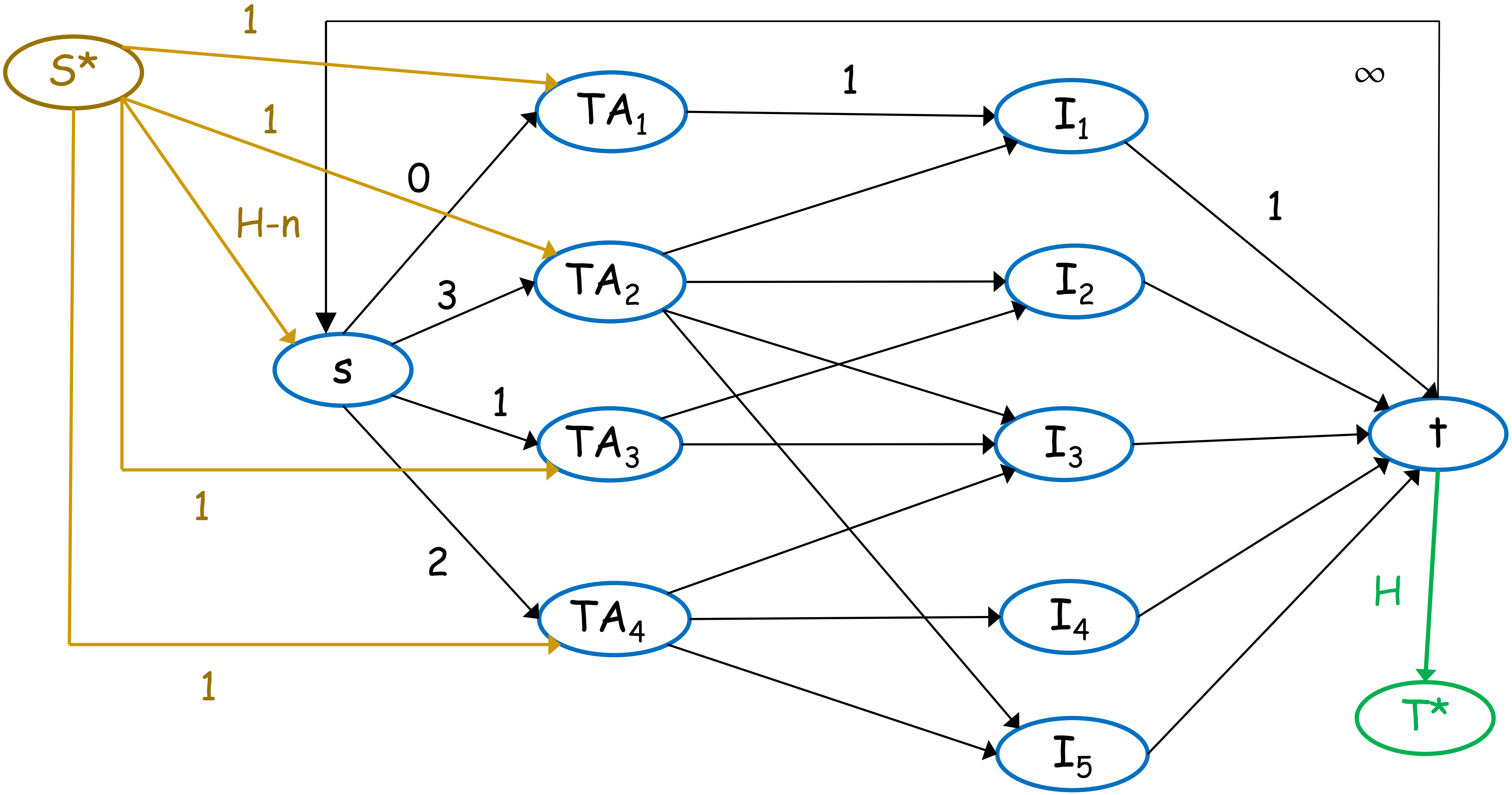
Add weights = - demands



Solution

Step3 : add new sink and connect it to the positive demands

Add weights = demands



Solution

There is a feasible circulation in the original graph if and only if the maximum flow from S* to T* in this graph has value H.