Lecture 19b: Applications of Max Bipartite Matching

Version of April 2, 2019

Applications of Max Bipartite Matching

- We just saw how to solve the Max-Bipartite Matching problem by reduction to (integral) Max-Flow and then using the Ford-Fulkerson Max-Flow algorithm.

- We will now see how to solve various scheduling problem by recasting them as Max-Bipartite Matching problems

1. Feasible Schedules

2. Balanced Assignments

3. Constrained Assignments

Max Flows: Feasible Schedule

Assume n roommates $r_1,\ldots r_n$.

For fairness, every day $d_1,\ldots d_n$ a different roommate is supposed to cook dinner.

However, due to other obligations, some roommates are unable to cook on certain days.

Let $C_{i,j}$=true, if r_i can cook on day d_j.

Describe an algorithm to determine if is possible to have a feasible schedule such that each roommate cooks exactly once during the n days.

Max Flows: Feasible Schedule

$C_{i,j}$=true, if r_i can cook on day d_j.

Describe an algorithm to determine if is possible to have a feasible schedule such that each roommate cooks exactly once during the n days.

Solution: This is a matching problem.

Create **bipartite graph** in which each roommate $r_1,\ldots r_n$ and each day $d_1,\ldots d_n$ are nodes.

Construct edge (r_i, d_j) iff $C_{i,j}$=true.

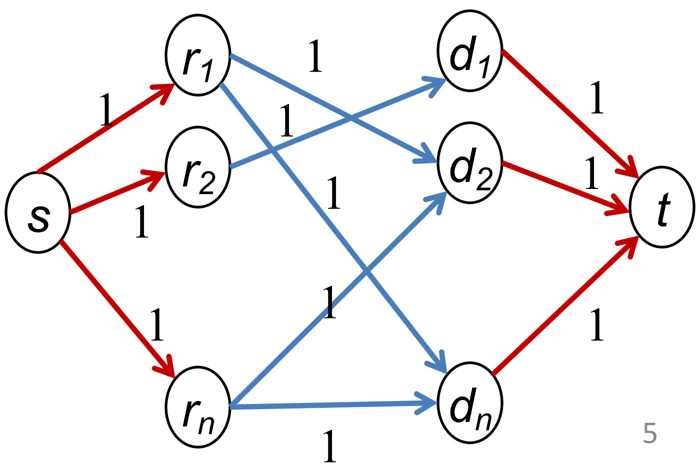
Add source node s with outgoing edges to all roommates $r_1,\ldots r_n$,
and sink t with incoming edges from all days $d_1,\ldots d_n$.
Set all edge capacities equal 1.

A feasible schedule exists if and only if

The bipartite graph has a perfect matching, i.e.,

A matching touching every vertex.

This happens iff
the max s-t flow has value n.



1. Feasible Schedules
2. Balanced Assignments
3. Constrained Assignments

Max Flows: Balanced Assignment

Your company wishes to assign n customers $c_1,\ldots c_n$ to k facilities $f_1,\ldots f_k$.

Each customer can only be served by some facility in his vicinity:
$C_{i,j}$=true means that customer c_i can be served by facility f_j.

An **assignment** of customers to facilities is balanced,
if each facility serves the same number n/k of customers
(assume that n/k is integer).

Given the constraints $C_{i,j}$, describe an algorithm to determine if is possible to construct a balanced assignment

Max Flows: Balanced Assignment

$C_{i,j}$=true means that customer c_i can be served by facility f_j.

Given constraints $C_{i,j}$, describe an algorithm to determine if is possible to construct a balanced assignment

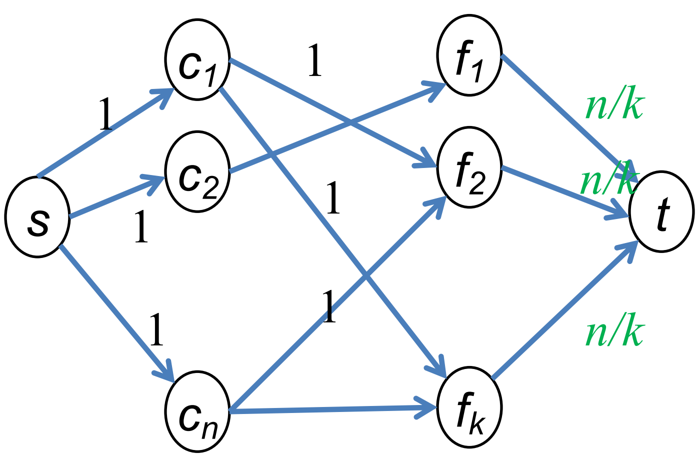
Solution: Create a bipartite graph.

Each customer $c_1,\ldots c_n$ and each facility $f_1,\ldots f_k$ are nodes.

Edge (c_i, f_j) exists iff $C_{i,j}$=true.

Add source s connected to all customers $c_1,\ldots c_n$,
and sink t with incoming edges from all facilities $f_1,\ldots f_k$.
All edge capacities = 1,
except for the edges (f_j, t) whose capacity is n/k.

A balanced assignment exists
if and only if
maximum s-t flow has value n.



1. Feasible Schedules

2. Balanced Assignments

3. Constrained Assignments

Max Flows: Constrained Assignment

Your company now wishes to assign n customers $c_1,\ldots c_n$ to k facilities $f_1,\ldots f_k$.

Each customer can only be served by some facility in his vicinity:
$C_{i,j}$=true means that customer c_i can be served by facility f_j

An **assignment** of customers to facilities is constrained,
so that facility f_i can serve n_i customers where $\sum_{i=1}^{k} n_i = n$.

Given the constraints $C_{i,j}$ and the n_i, describe an algorithm to determine if is possible to construct a constrained assignment that serves all of the customers and, if such an assignment exists, to construct it.

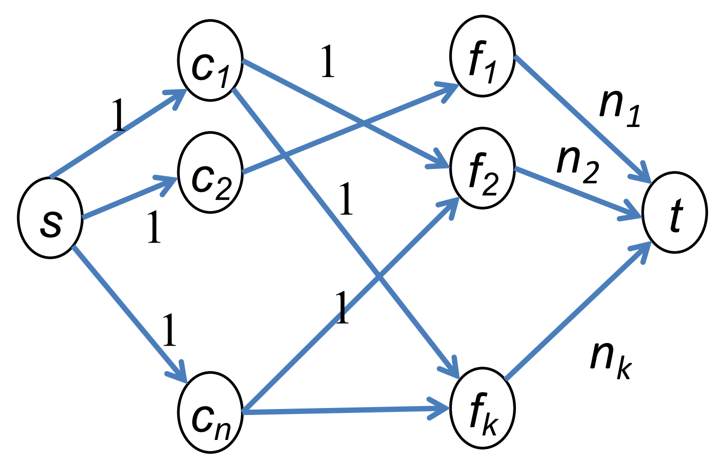
Max Flows: Constrained Assignment

$C_{i,j}$=true means that customer c_i can be served by facility f_j.
Facility f_i serves at most n_i customers where $\sum_{i=1}^{k} n_i = \mathrm{n}$

Describe an algorithm to determine if is possible to construct a constrained assignment given the constraints $C_{i,j}$ and values n_i

Solution: Create a bipartite graph in which each customer $c_1,\ldots c_n$ and each facility $f_1,\ldots f_k$ are nodes.
Edge (c_i, f_j) exists iff $C_{i,j}$=true.

Add
source s with outgoing edges to customers $c_1,\ldots c_n$
sink t with incoming edges from all facilities $f_1,\ldots f_k$
All edge capacities equal 1,
except for the edges (f_j, t) whose capacity is n_i



A constrained assignment exists
if and only if
maximum s-t flow has value n.