Maximum Bipartite Matching

Abrar Jahin Sarker[1]
Abdullah Muhammed Amimul Ehsan[2]
Mostafa Rifat Tazwar[3]

[1]2005012

[2]2005017

[3]2005020

[1,2,3]Department of Computer Science and Technology, BUET

February 11, 2024

Table Of Contents

Outline

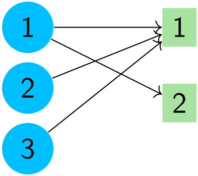
Introduction

Bipartite Graph

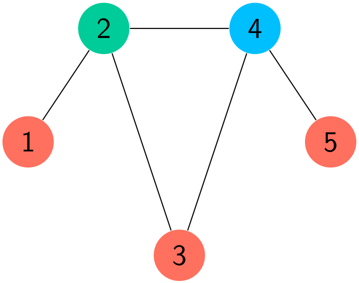
A bipartite graph is one whose vertices can be split into two independent groups U,V such that every edge connects vertices of different groups.

Note

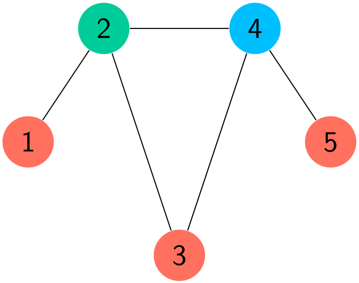
There can not be any edge between two vertices of U or two vertices of V .
The graph is two colourable and doesn't have cycles of odd length.



Odd length and Even length cycle

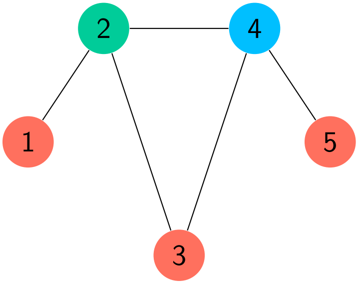


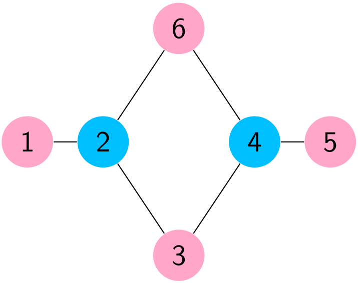
Odd length and Even length cycle



Odd length cycle
Not 2 colorable
Not a bipartite graph

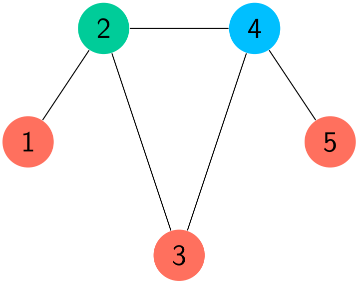
Odd length and Even length cycle



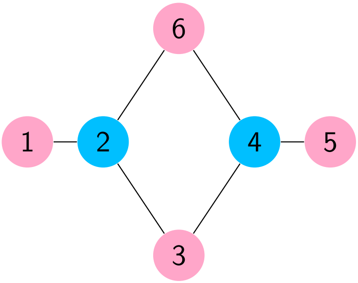


Odd length cycle
Not 2 colorable
Not a bipartite graph

Odd length and Even length cycle



Odd length cycle
Not 2 colorable
Not a bipartite graph



Even length cycle
2 colorable
Bipartite graph

Outline

The Problem

Maximum Bipartite Matching

Given a bipartite graph $G = (A \cup B, E)$, find an $\{\ S \subseteq A \times B : S$ is a matching and is as large as possible. $\}$ [KT06]

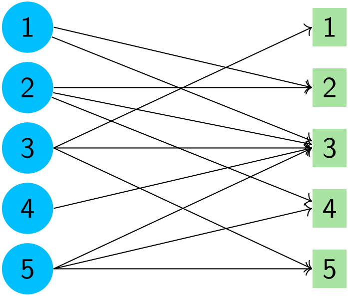
Outline

Example

In a picnic,there are 5 people and 5 food items.Some people express interest in some of the items.How can we satisfy maximum number of people while wasting minimum number of items?

Example

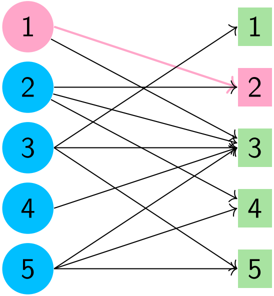
In a picnic,there are 5 people and 5 food items.Some people express interest in some of the items.How can we satisfy maximum number of people while wasting minimum number of items?



Outline

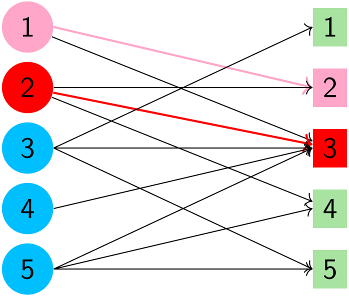
Greedy Approach

Person 1 starts by taking the first available item



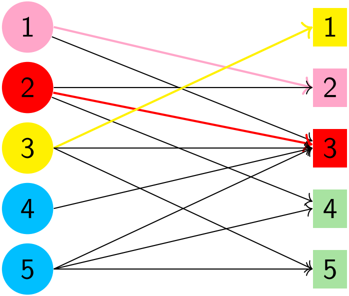
Greedy Approach

Making item 3, the only valid choice for person 2



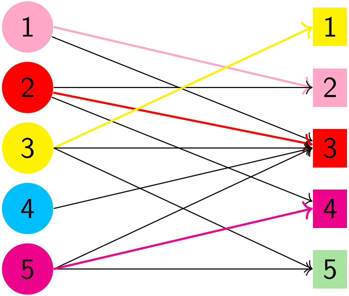
Greedy Approach

Here,item 1 was luckily unoccupied



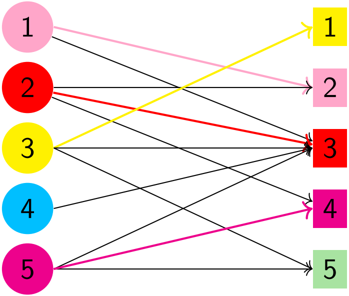
Greedy Approach

Here,person 4 can not have anything and person 5 got a match luckily.



Greedy Approach

Here, person 4 can not have anything and person 5 got a match luckily.



We could only satisfy four guests.One guest is unhappy.

Outline

Approach to solve

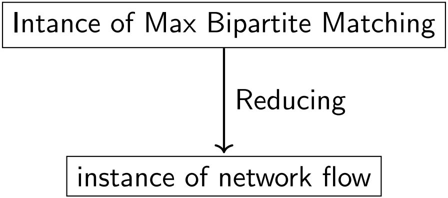
- Given an instance of bipartite matching

Approach to solve

- Given an instance of bipartite matching
- Reduce it to Max Flow Problem.

Approach to solve

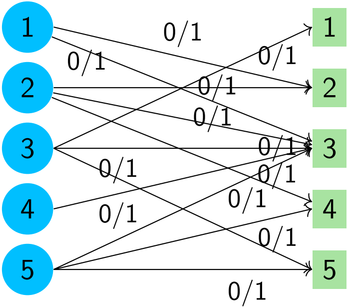
- Given an instance of bipartite matching
- Reduce it to Max Flow Problem.
- Where the solution to the network flow problem can easily be used to find the solution to the bipartite matching.



Reduction

Step 1

Make all the edges directed if not and add 0 flow and 1 capacity for all edges.Expressed as 0/1 in Flow/Capacity format

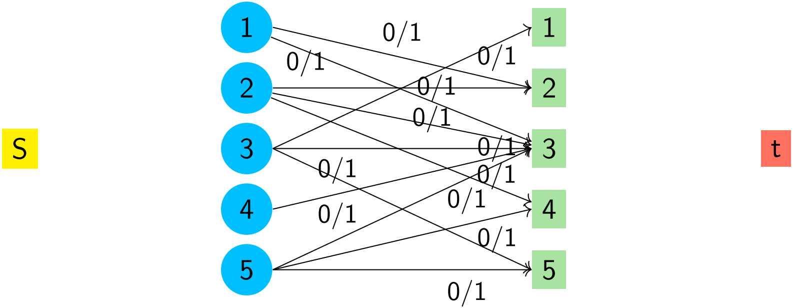


Reduction

Step 2

Add two new nodes:

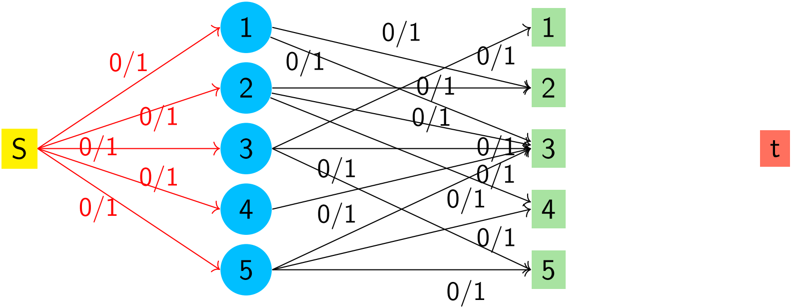
1. Source(S)
2. Destination(t)



Reduction

Step 3

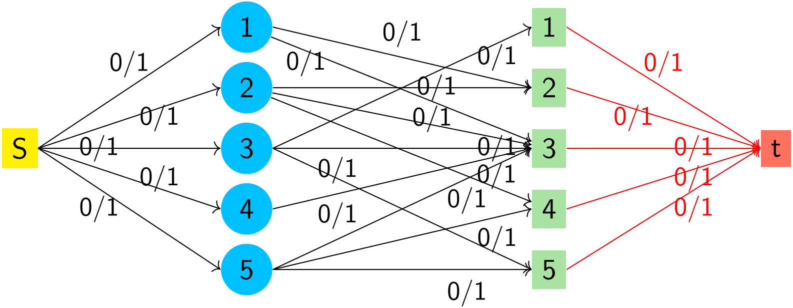
Add nodes from source to person with flow/capacity=0/1.



Reduction

Step 4

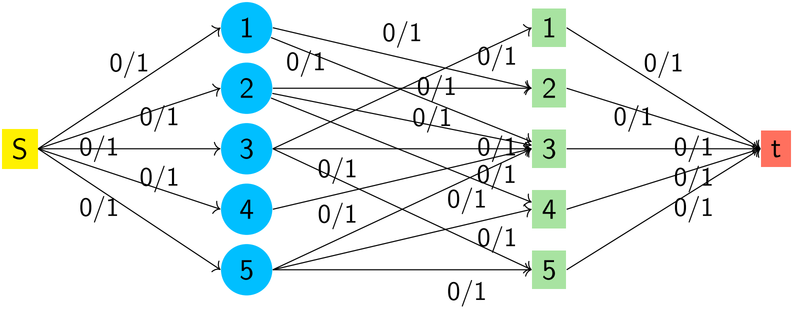
Add nodes from food items to destination with 0 flow and 1 capacity



Reduced to Network Flow Problem

Finally

It is reduced to a network flow problem where we have to apply network flow algorithm.Here, flow > 0 between the pairs indicates a matching.

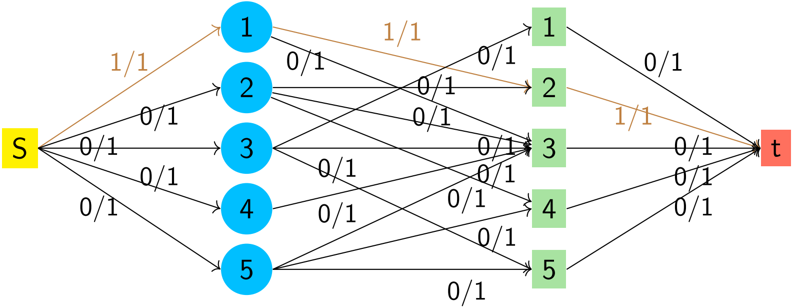


Edmonds-Karp Algorithm to Solve Network Flow Problem

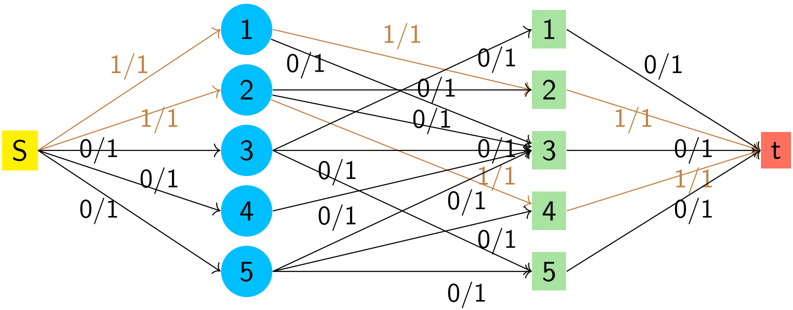
- Initialize flow $f(u, v) = 0$ for all edges (u, v) in the graph.
- Repeat the following steps until no augmenting paths can be found:
  - Use BFS to find the shortest augmenting path from source to sink.
  - If no augmenting path is found, terminate.
  - Let P be the augmenting path found by BFS.
  - Let $cf(P)$ be the minimum residual capacity along path P.
  - For each edge (u, v) in P:
    - Update flow $f(u, v) = f(u, v) + cf(P)$.
    - Update flow $f(v, u) = f(v, u) - cf(P)$ for the reverse edge.
- The value of the maximum flow is the sum of flow values leaving the source.

Outline

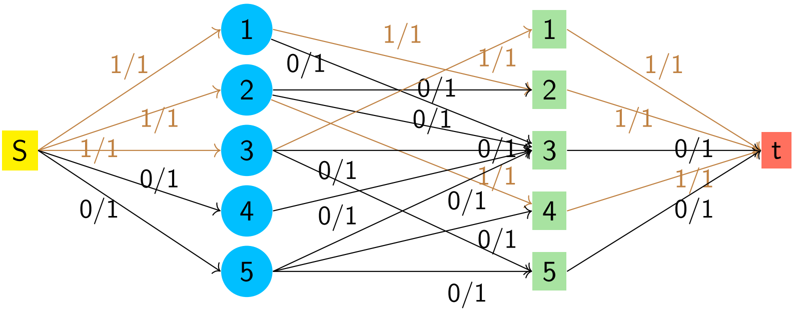
Solution to Network Flow Problem



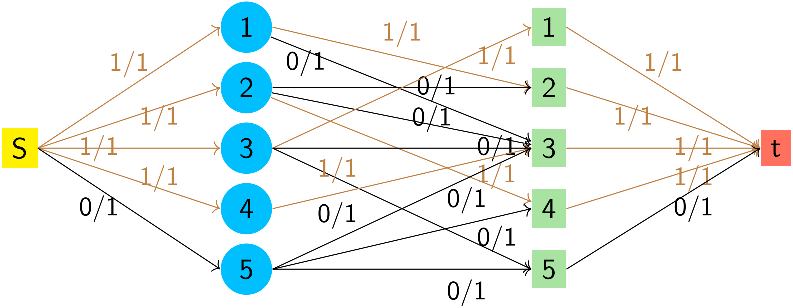
Solution to Network Flow Problem



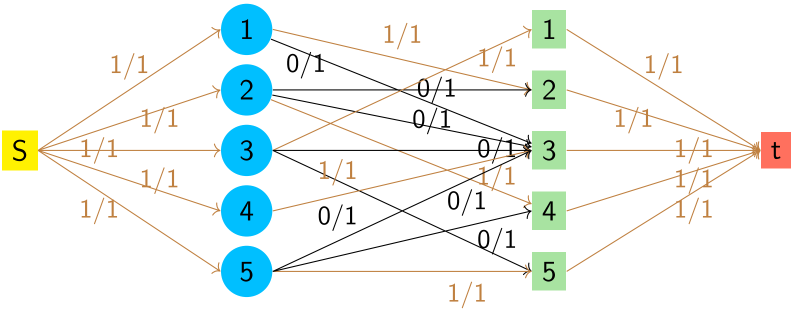
Solution to Network Flow Problem



Solution to Network Flow Problem



Solution to Network Flow Problem



Analyzing the solution

Here,we can not augment the paths anymore.We have already found the max flow of the network.The max flow=5.Hence,Number of matching=5.

Pairs are :

1. Person 1,Item 2

Analyzing the solution

Here,we can not augment the paths anymore.We have already found the max flow of the network.The max flow=5.Hence,Number of matching=5.

Pairs are :

1. Person 1,Item 2
2. Person 2,Item 4

Analyzing the solution

Here,we can not augment the paths anymore.We have already found the max flow of the network.The max flow=5.Hence,Number of matching=5.

Pairs are :

1. Person 1,Item 2
2. Person 2,Item 4
3. Person 3,Item 1

Analyzing the solution

Here,we can not augment the paths anymore.We have already found the max flow of the network.The max flow=5.Hence,Number of matching=5.

Pairs are :

1. Person 1,Item 2
2. Person 2,Item 4
3. Person 3,Item 1
4. Person 4,Item 3

Analyzing the solution

Here,we can not augment the paths anymore.We have already found the max flow of the network.The max flow=5.Hence,Number of matching=5.

Pairs are :

1. Person 1,Item 2
2. Person 2,Item 4
3. Person 3,Item 1
4. Person 4,Item 3
5. Person 5,Item 5

Analyzing the solution

Here,we can not augment the paths anymore.We have already found the max flow of the network.The max flow=5.Hence,Number of matching=5.

Pairs are :

1. Person 1,Item 2
2. Person 2,Item 4
3. Person 3,Item 1
4. Person 4,Item 3
5. Person 5,Item 5

Using greedy approach,we found 4 pairs which is not maximum.

Outline

Running time analysis

Running time analysis

1. How long does it take to solve the network flow problem on the new graph?

Running time analysis

1. How long does it take to solve the network flow problem on the new graph?
2. The Edmond Karp's algorithm uses BFS to find the shortest augmenting path from source to sink.

Running time analysis

1. How long does it take to solve the network flow problem on the new graph?
2. The Edmond Karp's algorithm uses BFS to find the shortest augmenting path from source to sink.
3. Running time of BFS is $O(V+E)$ where V is the number of vertices and E is the number of edges in the graph.

Running time analysis

1. How long does it take to solve the network flow problem on the new graph?
2. The Edmond Karp's algorithm uses BFS to find the shortest augmenting path from source to sink.
3. Running time of BFS is $O(V+E)$ where V is the number of vertices and E is the number of edges in the graph.
4. Each path search can take $O(VE)$ time in the worst case.

Running time analysis

1. How long does it take to solve the network flow problem on the new graph?
2. The Edmond Karp's algorithm uses BFS to find the shortest augmenting path from source to sink.
3. Running time of BFS is $O(V+E)$ where V is the number of vertices and E is the number of edges in the graph.
4. Each path search can take $O(VE)$ time in the worst case.
5. Hence, the running time of the algorithm is $O((V+E)VE) = O((E)VE) = O(VE^2)$.

Outline

Extension of Maximum Bipartite Matching

Allowing multiple items for a person

What if a person is allowed to take multiple items?

Extension of Maximum Bipartite Matching

Allowing multiple items for a person

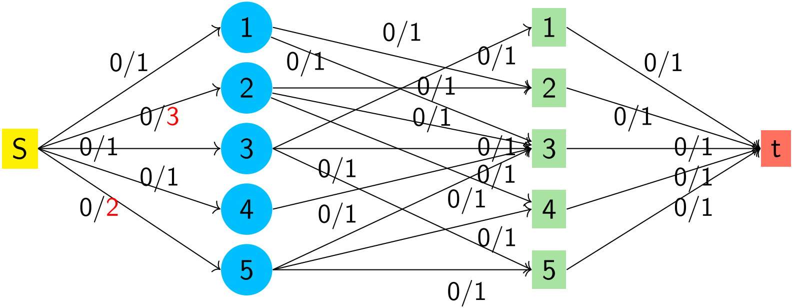
What if a person is allowed to take multiple items?

Solution

The capacity of edge connected the source and a person would be equal to the number of items that person is allowed to take.Then Network Flow algorithm should be applied as usual.

Extension of Maximum Bipartite Matching

For example if Person 2 is allowed to take 3 items and person 5 is allowed to take 2 items:



Extension of Maximum Bipartite Matching

Multiple copies of item available

What if an item has multiple instances?

Extension of Maximum Bipartite Matching

Multiple copies of item available

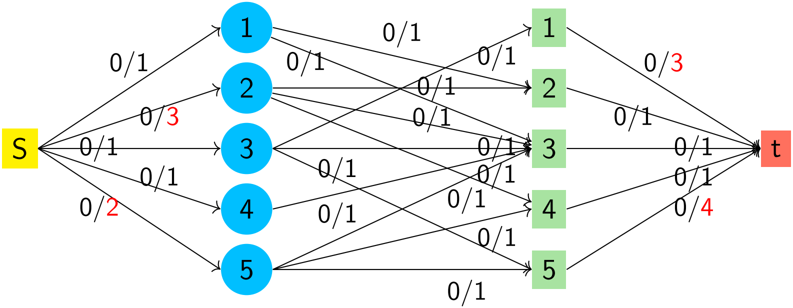
What if an item has multiple instances?

Solution

The capacity of edge connected the item and Destination would be equal to the number of copies of the item. Then Network Flow algorithm should be applied as usual.

Extension of Maximum Bipartite Matching

For example if item 1 has 3 copies and item 5 has 4 copies:



Extension of Maximum Bipartite Matching

Taking same item multiple times?

What if an item can be taken multiple times by a person?

Extension of Maximum Bipartite Matching

Taking same item multiple times?

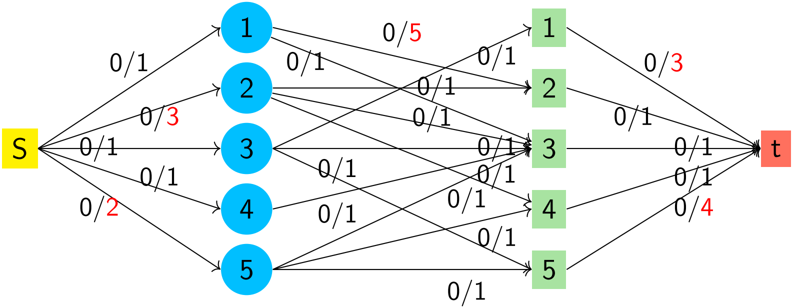
What if an item can be taken multiple times by a person?

Solution

The capacity of edge connected between the person and the item would be equal to the number of times the item can be taken by that person. Then Network Flow algorithm should be applied as usual.

Extension of Maximum Bipartite Matching

For example if Person1 can take Item 2 upto 5 times:



Outline

References

Jon Kleinberg and Eva Tardos, *Algorithm design*, Pearson Education India, 2006.

Outline

Conlusion

We hope you understood

Maximum Bipartite Matching

Feel free to ask any question...