COMP 261 (2024) - Tutorial 6 (Model Solution)

Tutorial questions for Week 6.

These are questions to work on in groups during your tutorial session. We do not expect all groups to complete all the questions. The goal is to ensure that you understand the concepts and techniques introduced in the lectures in order to be able to do the assignment and the test.

Network flow

1. The first decision to be made when implementing an algorithm for graph processing is what data structure to use. Your task is to explore the use of different data structures to implement Edmond Karp's algorithm for finding max flows in a graph. In order to figure out a "good" data structure, what points should you keep in mind?

Answer:

a) Ability to easily access the forward edges and the forward neighbours from a given node
b) Ability to easily access the reverse edge for a given forward edge
c) Ability to trace out the corresponding reverse path once a path has been found

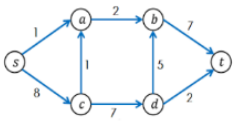
2. Evaluate the following data structures to achieve the objectives identified in question 1 above.
    - Adjacency matrix
    - Adjacency list: consisting of a collection of nodes, a collection of edges and for each node a collection of forward edges and backward edges

Answer:

Adjacency matrix:

a) Forward edge, forward neighbours: Follow the corresponding row
b) Reverse edge of a given edge: element (i, j) of the matrix represents the edge (i, j) of the graph - its reverse will be the element (j, i)
c) Trace out the path: Need to maintain a back-pointers Map

Nodes	Forward edges	Back edges
s	(s,a,1) , (s,c,8)	(a,s,0),(c,s,0)
a	(a,b,2)	(b,a,0)
c	(c,a,1) , (c,d,7)	(a,c,0) , (d,c,0)
b	(b,t,7)	(t,b,0)
d	(d,b,5) , (d,t,2)	(b,d,0), (t,d,0)
t	null	null



Adjacency list:

a) Forward edge, forward neighbours: Follow the collection of forward edges of the corresponding node
b) Reverse edge of a given edge (i, j) : Requires traversing the collection of reverse edges of i to find the corresponding reverse edge

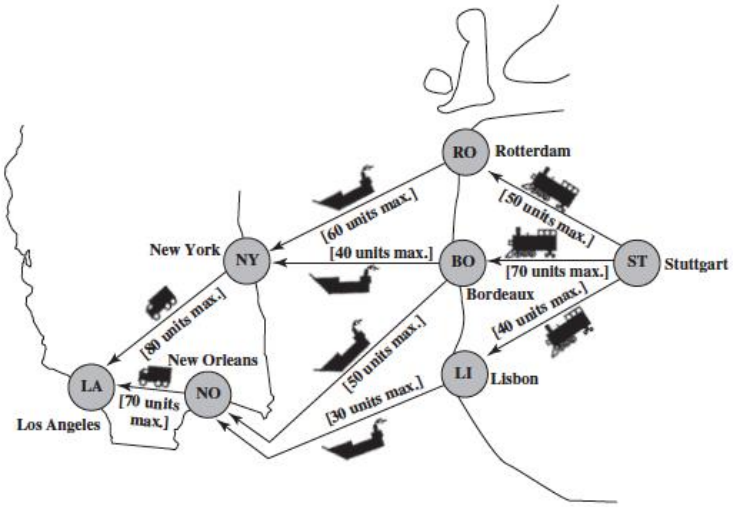
c) Trace out the path: Need to maintain a back-pointers Map

Do you see a problem with the adjacency list method above? How does the method used in the algorithm in the lecture slides help solve this problem.

Answer:

We are able to find the reverse edges in a constant time - As a result, we get the advantage of an adjacency matrix which takes constant time to find the reverse edge, and the advantage of an adjacency list in which we do not need to traverse the entire row to find a node's neighbours.

Q3. Use the Edmond Karp algorithm (as in the lecture slides, and repeated below) to identify the maximum flow from **ST to LA** through the network depicted in the graph below.

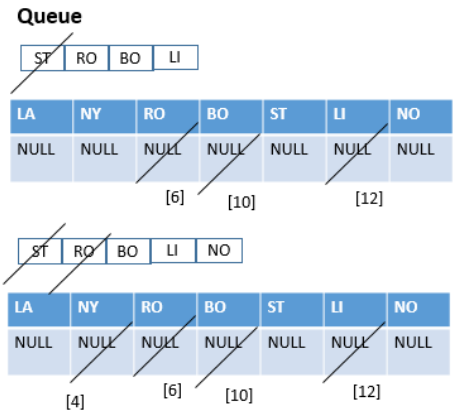


Node List

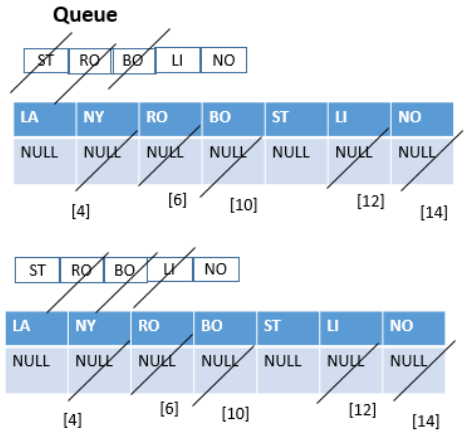
LA	1, 3
NY	0, 5, 9
RO	4, 7
BO	8,11, 14
ST	6, 10, 12
LI	13, 16
NO	2, 15, 17

Edge List

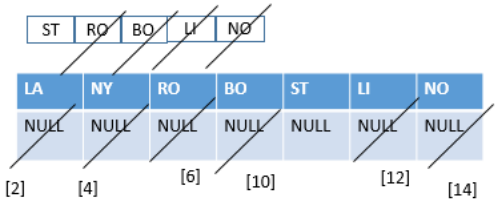
0	NY LA 80 0
1	LA NY 0 0
2	NO LA 70 0
3	LA NO 0 0
4	RO NY 60 0
5	NY RO 0 0
6	ST RO 50 0
7	RO ST 0 0
8	BO NY 40 0
9	NY BO 0 0
10	ST BO 70 0
11	BO ST 0 0
12	ST LI 40 0
13	LI ST 0 0
14	BO NO 50 0
15	NO BO 0 0
16	LI NO 30 0
17	NO LI 0 0



0	NY LA 80 0
1	LA NY 0 0
2	NO LA 70 0
3	LA NO 0 0
4	RO NY 60 0
5	NY RO 0 0
6	ST RO 50 0
7	RO ST 0 0
8	BO NY 40 0
9	NY BO 0 0
10	ST BO 70 0
11	BO ST 0 0
12	ST LI 40 0
13	LI ST 0 0
14	BO NO 50 0
15	NO BO 0 0
16	LI NO 30 0
17	NO LI 0 0



0	NY LA 80 0
1	LA NY 0 0
2	NO LA 70 0
3	LA NO 0 0
4	RO NY 60 0
5	NY RO 0 0
6	ST RO 50 0
7	RO ST 0 0
8	BO NY 40 0
9	NY BO 0 0
10	ST BO 70 0
11	BO ST 0 0
12	ST LI 40 0
13	LI ST 0 0
14	BO NO 50 0
15	NO BO 0 0
16	LI NO 30 0
17	NO LI 0 0



[2] [14] [10]

[10] [14] [2]

70 50 70

ST → BO →NO → LA : pathFlow: 50

0	NY LA 80 0
1	LA NY 0 0
2	NO LA 20 50
3	LA NO 50 0
4	RO NY 60 0
5	NY RO 0 0
6	ST RO 50 0
7	RO ST 0 0
8	BO NY 40 0
9	NY BO 0 0
10	ST BO 20 50
11	BO ST 50 0
12	ST LI 40 0
13	LI ST 0 0
14	BO NO 0 50
15	NO BO 50 0
16	LI NO 30 0
17	NO LI 0 0

Repeat the steps till no augmentation path is found with capacities >0

Note: Depending upon how Nodelist and Edgelist is prepared you may find some other augmentation paths, but the max flow value will remain the same

Max Flow: 150