Data Structures and Algorithms Spring 2024 — Problem Sets

by Kirill Efimovich

June 24, 2025

Week 14. Problem set

1 Task 1

1.1 Statement 1

Consider an algorithm for finding the maximum flow in a network that follows the Ford-Fulkerson method [Cormen, § 24.2] but uses a variation of the Dijkstra's algorithm to find augmenting paths of maximum residual capacity in the residual network on each iteration.

(a) Give an asymptotic time complexity of such an algorithm in terms of the number of vertices $|V|$, number of edges $|E|$, and the maximum flow of the network f_*. You may assume that edge capacities are positive integers. Provide full justification.

(b) (+2% extra credit)

Prove the asymptotic time complexity of the algorithm is:
$O\left(|E||V|\log(|V|)\cdot\log(f_*)+|E|^2\log(f_*)\right)$.

1.2 Solution

(a):
Dijkstra's algorithm runs in $O(|E|+|V|log|V|)$ time when implemented with a Fibonacci heap. Ford-Fulkerson algorithm runs f_* iterations. So, total time complexity will be: $O(f_*(|E|+|V|log|V|))$.
(a) Answer: $O(f_*(|E|+|V|log|V|))$

(b):
In the Ford-Fulkerson method, the maximum flow f_* is at most: $|E|\cdot c$, because each edge can at maximum contribute its capacity c. So we can rewrite it:
$f_* \leq |E|\cdot c$
$\log(f_*) \leq \log(|E|\cdot c)$
$\log(f_*) \leq \log|E| + \log c$, Hence:

$(f_*) = O(|E| \cdot \log c)$
After substitution in formula with Dijkstra's algorithm:
$O(|E| \cdot \log c \cdot |E||V| \log |V|)$
$O(|V||E|^2 \log c \log |V|)$
We can transform it to:
$O(|E||V| \log(|V|) \cdot \log(f_*))$

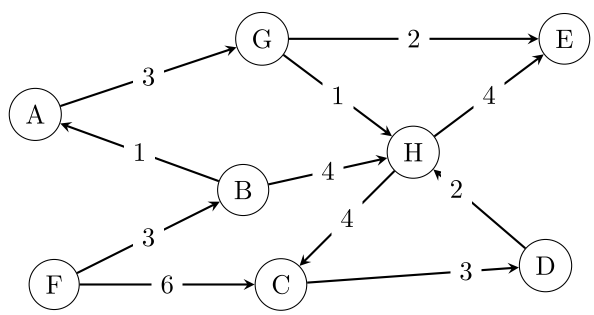
Each edge can be saturated and unsaturated f_* times, and to update residual network we need $O(\log(f_*))$. Combined, it gives: $O(|E|^2 \log(f_*))$

So, overall formula will be: $O(|E||V| \log(|V|) \cdot \log(f_*) + |E|^2 \log(f_*))$.
Q.E.D.

2 Task 2

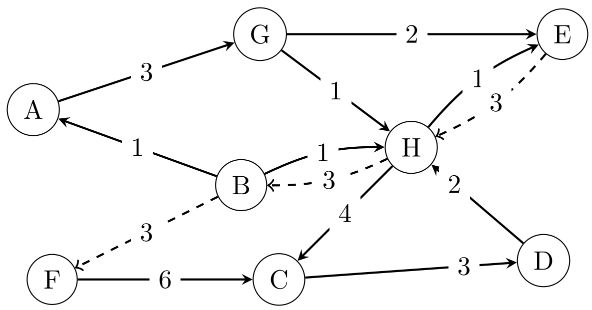
2.1 Statement

Run Edmonds-Karp algorithm [Cormen, § 24.2] on the given network:
(a) Identify the source and the sink of the network.
(b) For every iteration of the algorithm
i. write down the augmenting path (sequence of vertices),
ii. show the residual network after the iteration
(c) Write down the maximum flow value after the last iteration.
(d) Show that the flow is maximum by demonstrating a minimum cut of the network (as two sets of vertices).

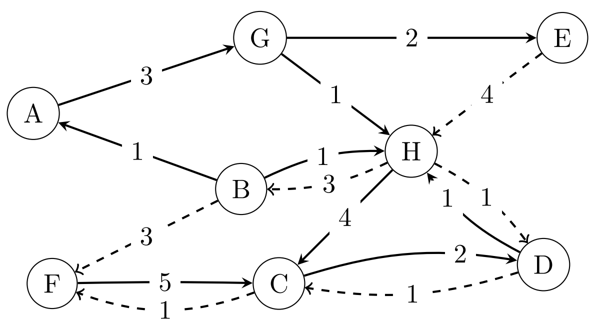


2.2 Solution

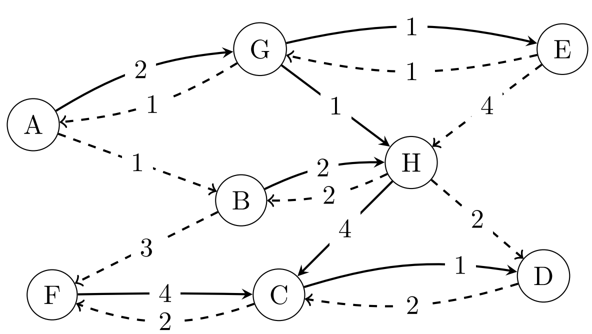
(a) Answer:
Source: F, Sink: E.
(b) Answer:
1st Iteration:
Augmenting path: $FBHE$
Residual network: (augmenting path is dotted)



2nd Iteration:
Augmenting path: *FCDHE*
Residual network: (augmenting path is dotted)



3rd Iteration:
Augmenting path: *FCDHBAGE*
Residual network: (augmenting path is dotted)



(c) Answer:
Maximum flow value after the last iteration: *5*.
(d) Answer:
Minimum cut: *FCD* and *AGEBH*.

3 Task 3

3.1 Statement

Consider a flow network $\mathcal{N}$ and a minimum cut $C_{\mathcal{N}} = \langle S, T \rangle$ such that there exist two vertices $u \in S$ and $v \in T$ and an edge $\langle v, u \rangle$ with capacity $c > 0$ and there is no edge $\langle u, v \rangle$ in $\mathcal{N}$. Let the maximum flow in $\mathcal{N}$ be $f_{\mathcal{N}}$.

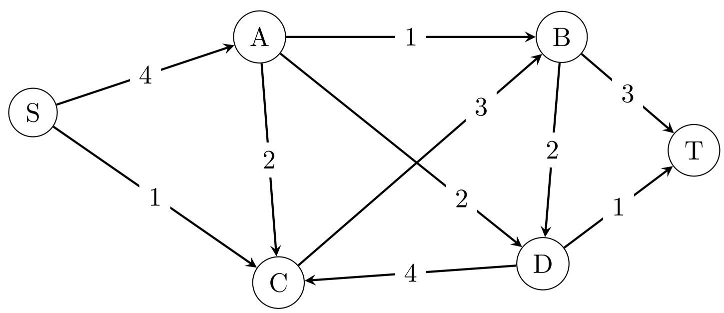
Now suppose that we invert the edge $\langle v, u \rangle$ in $\mathcal{N}$, obtaining a new flow network $\mathcal{N}^{\langle u,v \rangle}$. Let the maximum flow in $\mathcal{N}^{\langle u,v \rangle}$ be $f_{\mathcal{N}^{\langle u,v \rangle}}$.

Is it true that $f_{\mathcal{N}^{\langle u,v \rangle}} = f_{\mathcal{N}} + c$? If yes, prove it. If no, provide a counterexample with justification.

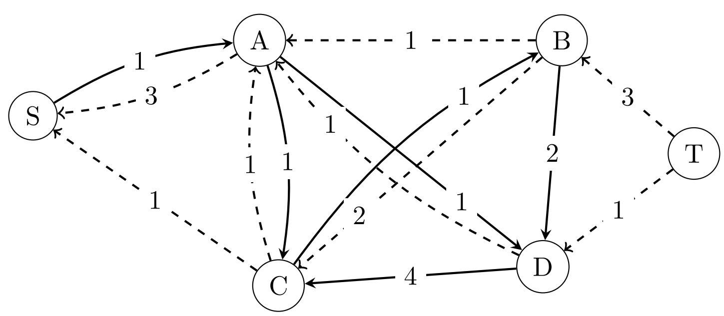
3.2 Solution

Answer: No, it is false

If we consider this network as $\mathcal{N}$:

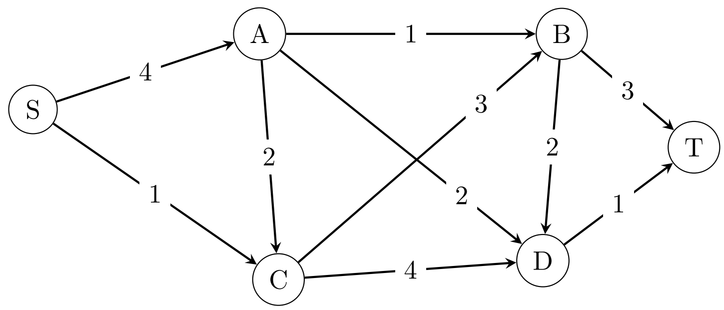


After Ford-Fulkerson method:

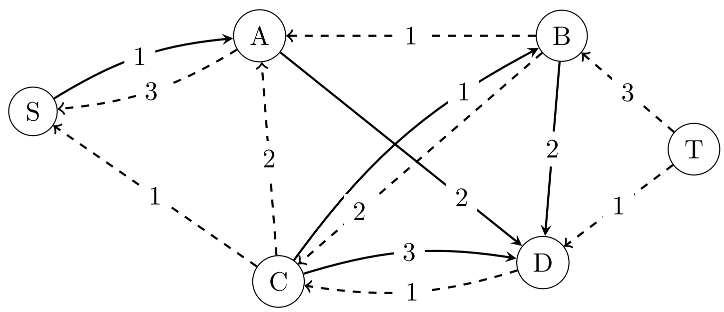


$f_{\mathcal{N}} = 4$.

If we pick edge $\langle D, C \rangle$ and reverse it to $\langle C, D \rangle$ we get $\mathcal{N}^{\langle D,C \rangle}$:



After Ford-Fulkerson method:



$f_{\mathcal{N}\langle D,C\rangle} = 4$.

We must find if $f_{\mathcal{N}\langle D,C\rangle} = f_{\mathcal{N}} + c$ is true.
$f_{\mathcal{N}\langle D,C\rangle} = 4$; $f_{\mathcal{N}} = 4$; $c = 4$; Hence:
$4 = 4 + 4$ - <u>FALSE</u>.
Answer: No, it is false

References

[Cormen] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein. Introduction to Algorithms, Fourth Edition. The MIT Press 2022