DAA Assignment

220001065, 220001066, 220001076

March 2024

1 Introduction

Below is the example of augmenting path.

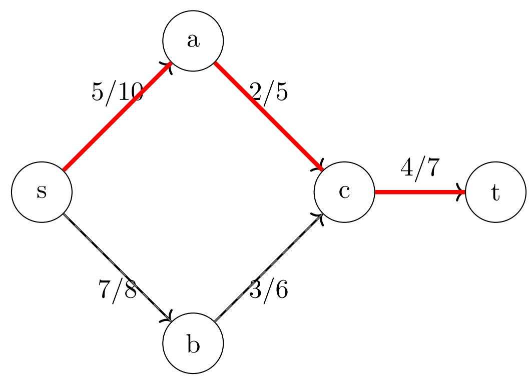

Figure 1: Example of an augmenting path in a network flow

1.1 Augmenting Path

An augmenting path in the context of network flow algorithms is a path from the source node s to the sink node t in a flow network graph. This path is used to increase the flow value from s to t.

Augmenting paths are crucial in algorithms like the Ford-Fulkerson method and its variants (such as the Edmonds-Karp algorithm) for finding the maximum flow in a network. These paths allow for increasing the flow along the edges without violating the capacity constraints of the network.

1.2 Residual Network

In network flow algorithms, a residual network is a modified version of the original flow network used to find augmenting paths. It contains residual capacities for each edge, indicating how much more flow can be pushed through that edge.

The residual capacity of an edge (u, v) is defined as $c_f(u, v) = c(u, v) - f(u, v)$, where $c(u, v)$ is the original capacity and $f(u, v)$ is the current flow.
For a flow network G = (V,E) and a flow f , consider a pair of vertices v,u $\in$ V, then residual capacity c_f(u,v) is defined as

$$c_f = \begin{cases} c(u, v) - f(u, v) & \text{if (u,v)} \in \text{E} \\ f(v, u) & \text{if (v,u)} \in \text{E} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Given a flow network G = (V,E) and flow f , the residual network of G induced by f is G_f = (V,E_f) , where

$$E_f = \{(u, v) \in V \times V : c_f(u, v) > 0\}$$

If f is the flow in G, and f' is a flow of the corresponding residual network G_f , we define $f \uparrow f'$, the augmentation of the flow f by f', to be a function from V $\times$ V to , defined by

$$(f \uparrow f')(u, v) = \begin{cases} f(u, v) + f'(u, v) - f'(v, u) & \text{if (u,v) } in \text{ E,} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

1.3 Lemma

Let G = (V,E) be a flow network with source s and sink t, and let f be a flow in G. Let G_f be the residual network of G induced by f and let f' be a flow in G_f. Then the function $f \uparrow f'$ defined in equation (2) is a flow in G with value $|\, f \uparrow f' \,| = |f| + |\,f'|$.

Proof We first verify that $f \uparrow f'$ obeys the capacity constraint for each edge in E and flow conservation at vertex in V - {s,t}.

For the capacity constraint, first observe that if (u,v) $\in$ E, then c_f(v,u) = $f(u, v)$ from above equation (1). Because f' is flow in G_f , we have $f'(v, u) \leq c_f(v, u)$, which gives $f'(v, u) \leq f(u, v)$. Therefore,

$$\begin{aligned} (f \uparrow f')(u, v) &= f(u, v) + f'(u, v) - f'(v, u) && \text{(from eq. (2))} \quad (3) \\ &\geq f(u, v) + f'(u, v) - f(u, v) && (\because f'(v, u) \leq f(u, v)) \quad (4) \\ &= f'(u, v) && \text{(flows are non negative)} \quad (5) \\ &\geq 0 && (6) \end{aligned}$$

and

$$(f \uparrow f')(u,v) = f(u,v) + f'(u,v) - f'(v,u) \quad \text{(from eq. (2))} \tag{7}$$
$$\leq f(u,v) + f'(u,v) \quad (\because \text{flows are non negative}) \tag{8}$$
$$\leq f(u,v) + c_f(u,v) \quad \text{(from capacity constraint)} \tag{9}$$
$$= f(u,v) + c(u,v) - f(u,v) \quad \text{(from eq. (1))} \tag{10}$$
$$= c(u,v). \tag{11}$$

For the flow constraint, we have to prove incoming flow is equal to outgoing flow of augumented graph

$$\sum_{v \in V} (f \uparrow f')(u,v) = \sum_{v \in V} (f \uparrow f')(v,u) \tag{12}$$

We have from G, Flow conservation in graph G: For all u $\in$ V - {s,t}, graph G satisfies

$$\sum_{v \in V} f(v,u) = \sum_{v \in V} f(u,v) \tag{13}$$

Proof

$$\sum_{v \in V} (f \uparrow f')(u,v) = \sum_{v \in V} [f(u,v) + f'(u,v) - f'(v,u)] \quad \text{(from eq. (2))} \tag{14}$$
$$= \sum_{v \in V} f(u,v) + \sum_{v \in V} f'(u,v) - \sum_{v \in V} f'(v,u) \tag{15}$$
$$= \sum_{v \in V} f(v,u) + \sum_{v \in V} f'(v,u) - \sum_{v \in V} f'(u,v) \quad \text{(from eq.(13))} \tag{16}$$
$$= \sum_{v \in V} [f(v,u) + f'(v,u) - f'(u,v)] \tag{17}$$
$$= \sum_{v \in V} (f \uparrow f')(v,u) \quad \text{(from eq.(2))} \tag{18}$$

Now we are ready to prove $|f \in f'| = |f| + |f'|$, we have the total net flow

$$|f| = \sum_{v \in V} f(s,v) - \sum_{v \in V} f(v,s) \tag{19}$$

$$|f'| = \sum_{v \in V} f'(s,v) - \sum_{v \in V} f'(v,s) \tag{20}$$

and

$$|f \uparrow f'| = \sum_{v \in V} (f \uparrow f')(s,v) - \sum_{v \in V} (f \uparrow f')(v,s) \tag{21}$$

where 's' is the source node. And let
$v1 = \{v \in V | (s, v) \in E\}$
$v2 = V - v1$, then
$v1 \cup v2 = V$ and $v1 \cap v2 = \emptyset$
By the definition of augmentation flow from equation(2) only vertices v $\in$ v1 have positive $(f \uparrow f')(s, v)$ and only vertices v $\in$ v2 have positive $(f \uparrow f')(v, s)$
Then from eq.(21)

$$
\begin{aligned}
|f \uparrow f'| &= \sum_{v \in V} (f \uparrow f')(s, v) - \sum_{v \in V} (f \uparrow f')(v, s) &(22)\\
&= \sum_{v \in v1} [f(s, v) + f'(s, v) - f'(v, s)] - \sum_{v \in v2} [f(v, s) + f'(v, s) - f'(s, v)] &(23)\\
&= [\sum_{v \in v1} f(s, v) - \sum_{v \in v2} f(v, s)] + [\sum_{v \in v1} f'(s, v) + \sum_{v \in v2} f'(s, v)] - [\sum_{v \in v1} f'(v, s) + \sum_{v \in v2} f'(v, s)] &(24)
\end{aligned}
$$

Since $v1 \cap v2 = \emptyset$ and $v1 \cup v2 = V$ then,from principle of Inclusion and Exclusion, we can write

$$\sum_{v \in v1} f'(s, v) + \sum_{v \in v2} f'(s, v) = \sum_{v \in v1 \cup v2} f'(s, v) \tag{25}$$

$$\sum_{v \in v1} f'(v, s) + \sum_{v \in v2} f'(v, s) = \sum_{v \in v1 \cup v2} f'(v, s) \tag{26}$$

continuation of equation 24

$$
\begin{aligned}
|f \uparrow f'| &= |f| + \sum_{v \in v1 \cup v2} f'(s, v) - \sum_{v \in v1 \cup v2} f'(v, s) \quad \text{(from eq.(19) and eq.(25) and eq.(26))} &(27)\\
&= |f| + \sum_{v \in V} f'(s, v) - \sum_{v \in V} f'(v, s) &(28)\\
&= |f| + |f'| \quad \text{(from net flow in the residual network)} &(29)
\end{aligned}
$$

$\therefore \quad |f \uparrow f'| = |f| + |f'|$

1.4 Ford-Fulkerson Method

The Ford-Fulkerson method is an algorithm for computing the maximum flow in a flow network. It repeatedly finds augmenting paths and increases the flow along these paths until no more augmenting paths can be found.

1.5 Algorithms Utilizing Flow Network Methods

Several algorithms utilize flow network methods, including:

- Ford-Fulkerson Algorithm
- Edmonds-Karp Algorithm
- Dinic's Algorithm
- Push-Relabel Algorithm (such as the Goldberg-Tarjan Algorithm)

These algorithms use concepts like augmenting paths, residual networks, and capacity scaling to efficiently compute maximum flows or solve related network flow problems.

1.6 Maximum Flow Minimum Cost Problem

The maximum flow minimum cost problem is an extension of the maximum flow problem where each edge in the network has an associated cost. The goal is to find the flow that maximizes the total flow value while minimizing the total cost of using the edges. This problem has applications in various domains, including transportation, logistics, and network design.

1.7 Algorithm Overview

The Ford-Fulkerson algorithm operates as follows:

1. Start with an initial flow of 0.
2. While there exists an augmenting path from the source to the sink:
   - Find an augmenting path using a search algorithm like Breadth-First Search (BFS) or Depth-First Search (DFS).
   - Compute the maximum flow that can be pushed along the augmenting path.
   - Update the flow in the network by adding the flow along the augmenting path.
3. When no augmenting paths exist, the maximum flow is reached.

1.8 Proof

1.8.1 Correctness of the Algorithm

To prove the correctness of the Ford-Fulkerson algorithm, we need to show that it terminates and returns the maximum flow.

Termination The algorithm terminates because in each iteration, the flow value increases until reaching the maximum flow, and the flow value cannot increase indefinitely.

Maximum Flow Assume f is the maximum flow computed by the Ford-Fulkerson algorithm. We will show that there is no augmenting path from source s to sink t in the residual network G_f.

- Suppose there exists an augmenting path p in G_f from s to t.
- Let $c_f(p)$ be the residual capacity of the augmenting path p.
- Augmenting the flow along p by $c_f(p)$ units increases the flow to $f + c_f(p)$, contradicting the assumption that f is the maximum flow.
- Therefore, no augmenting path exists in G_f, and the algorithm terminates correctly.

1.9 Example Diagram

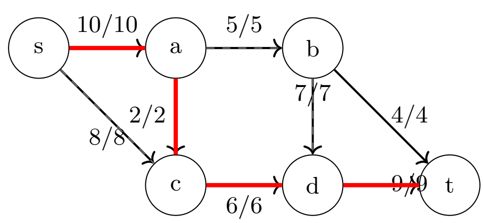

Figure 2: Example of Ford-Fulkerson Algorithm Operation

Figure 2 illustrates an example of the Ford-Fulkerson algorithm operation in a flow network. The red edges represent the augmenting path found by the algorithm.