The Maximum Flow Problem

Niharika Reddy Peddinenikalva [1] Adam Letchford [1]

[1]Lancaster University

1. Motivation

Networks exist everywhere in our daily lives - telecommunications, oil and gas distribution, roadways, etc. In network flow problems, we aim to distribute some entity across a network as efficiently as possible. The maximum flow problem is a well-known network flow problem.

2. The Maximum Flow Problem

In this problem, we have a network, denoted by a graph $G = (V, A)$, with sets V of n nodes and A of m arcs. Each arc $a \in A$ has a flow capacity $u_a \in \mathbb{R}_+$. The aim is to maximise the flow of an entity from a source $s \in V$ to a sink $t \in V$, without exceeding the flow capacity of any arc.

Let x_a be the decision variable for the amount of flow through an arc $a \in A$. Let $\delta^+(i) = \{(i,j) \in A \mid j \in V\}$ and $\delta^-(i) = \{(j,i) \in A \mid j \in V\}$ denote the sets of arcs leaving and entering node i, respectively. Then, Ford and Fulkerson [1] formulate the maximum flow problem as an LP,

$$\max \sum_{a \in \delta^+(s)} x_a \tag{1}$$

$$\text{s.t.} \sum_{a \in \delta^+(i)} x_a - \sum_{a \in \delta^-(i)} x_a = 0 \qquad \forall i \in V \backslash \{s,t\}, \tag{2}$$

$$0 \le x_a \le u_a \qquad \forall a \in A. \tag{3}$$

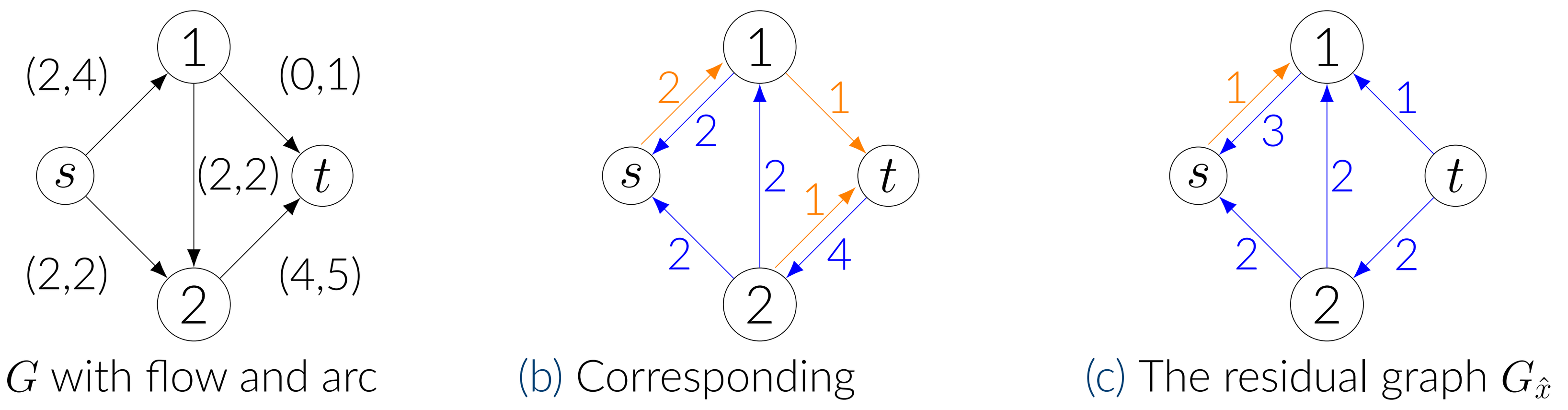
A flow $x \in \mathbb{R}^m$ is a non-negative vector which satisfies flow conservation in Eq. (2) and arc capacities in Eq. (3).

Residual Graphs and Augmenting Paths

Every graph $G = (V, A)$ with flow x has a residual graph $G_x = (V, \bar{A})$. The set of nodes in G_X is the same as in G. The arc set $\bar{A}$ has:

- *Forward arcs*: For each arc $a = (i,j) \in A$ such that $x_a < u_a$, we have an arc $(i,j) \in \bar{A}$ with capacity $r_a \coloneqq u_a - x_a$.
- *Reverse arcs*: For each arc $a = (i,j) \in A$ such that $x_a > 0$, we have an arc $(j,i) \in \bar{A}$ with capacity $r_a \coloneqq x_a$.

An **augmenting path** is an (s,t)-path in the residual graph (e.g., path $s \to 1 \to t$ in Figure 1b). A flow x is a **blocking flow** if the residual graph G_x does not contain any augmenting paths (e.g., Figure 1c). A flow x is maximal if and only if the residual graph G_x contains no augmenting paths [1]. All maximum flows are blocking, but the converse is not true.



(a) G with flow and arc capacities (x_a, u_a). (b) Corresponding residual graph G_x. (c) The residual graph $G_{\hat{x}}$ of a maximal flow $\hat{x}$.

Figure 1. Example of a graph $G = (V, A)$ with flow x, the corresponding residual graph G_x with forward and reverse arcs and the residual graph $G_{\hat{x}}$ for a maximal flow $\hat{x}$.

3. Solution Method: The Ford-Fulkerson Algorithm

Ford and Fulkerson [1] designed an augmenting path algorithm which exploits the special structure of maximum flow problems.

Algorithm 1: Ford-Fulkerson Algorithm

input : $G = (V, A)$; arc capacities $u_a,\ a \in A$; source/sink $s, t \in V$;
output: Flow x from s to t with maximum flow value
Set initial flow $x \leftarrow 0$;
while G_x *contains an augmenting* (s,t)*-path* **do**
  Identify an augmenting path P in G_x from node s to node t;
  Find minimum residual capacity $\delta \coloneqq \min\{r_a : a \in P\}$;
  Set $x_a \leftarrow x_a + \delta$ for each $a \in P$ and update G_x;
end

4. Application: Evacuation Route Planning

Suppose we have an emergency at the SAT Building, Lancaster University, UK (source) and we wish to divert people via vehicles to the Royal Lancaster Infirmary (sink). We can maximise the flow of vehicles from the source of disaster to a sink of safety and/or care through the road network in Lancashire, UK.

We model the road network using the OpenStreetMap dataset [2]. Arcs and nodes represent road segments and intersections, respectively. Arc capacities are the number of vehicles that can flow through an arc per hour. This is now a maximum flow problem with the LP formulation in Eqs. (1) - (3). The flows inform the number of vehicles per hour along the routes from the source to sink.

We implement the Ford-Fulkerson (FF) algorithm and three general LP solution methods (primal-simplex (PS), dual-simplex (DS), and an interior point method (IPM)) to solve the maximum flow problem. The four methods are compared for solution quality and run times.

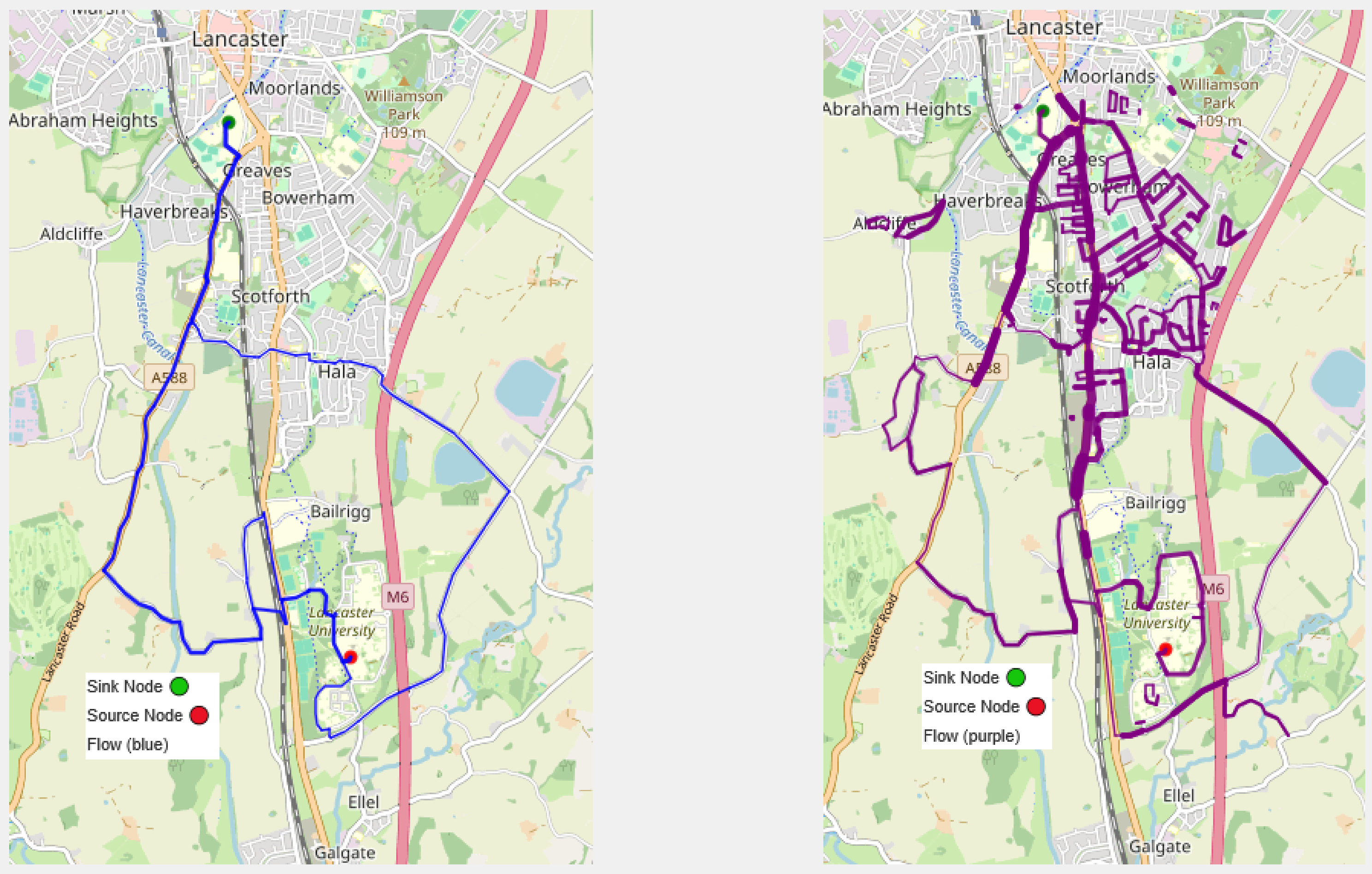
Results

The maximum flow of vehicles from the SAT building to Royal Lancaster Building was found to be 500 vehicles per hour with all four methods. Table 1 shows that the Ford-Fulkerson method is much quicker than the general LP solution methods.

Solution Method	FF	PS	DS	IPM
Average run time (in seconds)	0.0021	0.0024	0.0030	0.0042

Table 1. Average run time (in seconds) for each solution method.

The maximum flow algorithms also find better quality solutions (Figure 2a) than general LP solution methods. PS yields feasible solutions with unnecessary loops (Figure 2b). This is not desirable or realistic during an emergency evacuation. DS and IPM yield similar solutions to PS. Thus, algorithms tailored for maximum flow yield more suitable flows than general LP solution methods.



(a) FF flows (in blue). (b) PS Flows (in purple).

Figure 2. Flows from the SAT Building to Royal Lancaster Infirmary.

5. Further work

Adjustments to the LP formulation may improve the LP solutions obtained. We could investigate the performance of the four methods for large-scale problems and/or including more efficient maximum flow algorithms. We could also include multiple source/sink nodes to evacuate people to/from more locations. Including minimised time as an objective could yield more realistic solutions.

References

[1] L. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962. ISBN 9780691625393.

[2] OpenStreetMap contributors. Data retrieved from openstreetmap via the overpass api. https://www.openstreetmap.org, 2025. Accessed via OSMnx in April 2025.

https://www.lancaster.ac.uk/stor-i-student-sites/niharika-peddinenikalva/ n.peddinenikalva@lancaster.ac.uk