

# Maximum flow

In the 1950s, T.E. Harris at the RAND Corporation in Santa Monica (California) called attention for the following problem:

Consider a rail network connecting two cities by way of a number of intermediate cities, where each link of the network has a number assigned to it representing its capacity. Assuming a steady state condition, find a maximal flow from one given city to the other.

This question raised a stream of research. The problem can be formalized as follows.

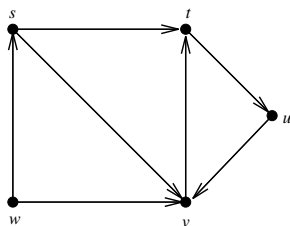
**Definition.** A directed graph  $D$  is a pair  $(V, A)$  where  $V$  is a finite set and  $A$  is a collection of ordered pairs of elements of  $V$ .

Any element of  $V$  is called a *vertex* of  $D$  and any element of  $A$  is called an *arc* of  $D$ .

An example of a directed graph is:

$$(1) \quad (\{s, t, u, v, w\}, \{(s, t), (s, v), (t, u), (u, v), (v, t), (w, v), (w, s)\}).$$

A directed graph  $D = (V, A)$  can be represented by a picture by choosing a point for each vertex, and drawing an arrow from  $v'$  to  $v''$  whenever  $(v', v'')$  is an arc of  $D$ . Thus the graph given in (1) can be represented as:



If  $a = (v', v'')$  is an arc, then one says that  $a$  *leaves*  $v'$  and *enters*  $v''$ . Thus in the graph above, arc  $(s, v)$  leaves  $s$  and enters  $v$ . One also says that arc  $a = (v', v'')$  *runs from*  $v'$  to  $v''$ , and that  $v'$  is the *tail* of  $a$  and  $v''$  is the *head* of  $a$ .

Let now a directed graph  $D = (V, A)$  be given with two special vertices, a 'source'  $r$  and a 'sink' or 'terminal'  $s$ . Then we have the following definition:

**Definition.** An  $r - s$  flow is a function  $f : A \rightarrow \mathbb{R}_+$  such that for each vertex  $v \neq r, s$  the following holds:

$$(2) \quad \sum_{a \in \delta^{\text{in}}(v)} f(a) = \sum_{a \in \delta^{\text{out}}(v)} f(a).$$

Here  $\delta^{\text{in}}(v)$  denotes the set of arcs entering  $v$ , while  $\delta^{\text{out}}(v)$  denotes the set of arcs leaving  $v$ .

Condition (2) is called the *flow conservation law*. It says that at any vertex  $v \neq r, s$ , the amount of ingoing flow is equal to the amount of outgoing flow.

If  $f$  is an  $r - s$  flow, the *value* of  $f$  is by definition equal to the net flow leaving  $r$ ; that is:

$$(3) \quad \text{value}(f) := \sum_{a \in \delta^{\text{out}}(r)} f(a) - \sum_{a \in \delta^{\text{in}}(r)} f(a).$$

**Exercise 1.** Prove that this value is equal to the net flow entering  $s$ :

$$(4) \quad \sum_{a \in \delta^{\text{in}}(s)} f(a) - \sum_{a \in \delta^{\text{out}}(s)} f(a).$$

Flows are particularly interesting if a function  $c : A \rightarrow \mathbb{R}_+$  is given, which serves as a ‘capacity’. One says that a flow  $f$  is *subject to*  $c$  if

$$(5) \quad f(a) \leq c(a)$$

for each arc  $a$ .

Now the **maximum flow problem** can be formulated:

- (6) given: a directed graph  $D = (V, A)$ , vertices  $r, s \in V$ , and a ‘capacity’ function  $c : A \rightarrow \mathbb{R}_+$ ;  
 find: an  $r - s$  flow  $f$  subject to  $c$  maximizing  $\text{value}(f)$ .

Around 1955, L.R. Ford, Jr and D.R. Fulkerson at RAND designed a fast method to solve this problem ([6]), and proved a theorem, the *max-flow min-cut theorem*, that characterizes the solution of the maximum flow problem ([5]).

For any subset  $W$  of  $V$  define  $\delta^{\text{out}}(W)$  to be the set of arcs leaving  $W$  (i.e., with tail in  $W$  and head in  $V \setminus W$ ). Similarly, define  $\delta^{\text{in}}(W)$  to be the set of arcs entering  $W$  (i.e., with tail in  $V \setminus W$  and head in  $W$ ).

If  $r \in W$  and  $s \notin W$  then  $\delta^{\text{out}}(W)$  is called an  $r - s$  cut. The *capacity* of  $\delta^{\text{out}}(W)$  is the sum of the  $c(a)$  for all  $a \in \delta^{\text{out}}(W)$ :

$$(7) \quad \text{cap}(\delta^{\text{out}}(W)) := \sum_{a \in \delta^{\text{out}}(W)} c(a).$$

The capacity of any  $r - s$  cut is an upper bound on the maximal value of  $r - s$  flows:

**Proposition 1.** For every  $r - s$  flow  $f$  and every  $r - s$  cut  $\delta^+(W)$  one has:

$$(8) \quad \text{value}(f) \leq \text{cap}(\delta^+(W)).$$

**Proof.**

$$\begin{aligned} (9) \quad \text{value}(f) &= \sum_{a \in \delta^+(r)} f(a) - \sum_{a \in \delta^-(r)} f(a) \\ &= \sum_{a \in \delta^+(r)} f(a) - \sum_{a \in \delta^-(r)} f(a) + \sum_{v \in W \setminus \{r\}} \left( \sum_{a \in \delta^+(v)} f(a) - \sum_{a \in \delta^-(v)} f(a) \right) \\ &= \sum_{v \in W} \left( \sum_{a \in \delta^+(v)} f(a) - \sum_{a \in \delta^-(v)} f(a) \right) = \sum_{a \in \delta^+(W)} f(a) - \sum_{a \in \delta^-(W)} f(a) \\ &\stackrel{*}{\leq} \sum_{a \in \delta^+(W)} f(a) \stackrel{**}{\leq} \sum_{a \in \delta^+(W)} c(a) = \text{cap}(\delta^+(W)). \quad \blacksquare \end{aligned}$$

Moreover, the following holds:

$$(10) \quad \text{equality holds in (8)} \iff \begin{aligned} \forall a \in \delta^-(W) : f(a) = 0 \text{ and} \\ \forall a \in \delta^+(W) : f(a) = c(a). \end{aligned}$$

This follows directly from the inequalities  $\star$  and  $\star\star$  in (9).

We now describe the algorithm of Ford and Fulkerson to determine a maximum flow. We assume that  $c(a) > 0$  for each arc  $a$ . First we describe an important subroutine. To this end, an  $r - s$  path is a sequence  $(v_0, a_1, v_1, \dots, a_t, v_t)$  where  $v_0, v_1, \dots, v_t$  are vertices, with  $v_0 = r, v_t = s$ , and  $a_1, \dots, a_t$  are arcs such that  $a_i = (v_{i-1}, v_i)$  for  $i = 1, \dots, t$ .

### Flow augmenting algorithm

**input:** an  $r - s$  flow  $f \leq c$ .

**output:** either (i) an  $r - s$  flow  $f' \leq c$  with  $\text{value}(f') > \text{value}(f)$ ,  
or (ii) an  $r - s$  cut  $\delta^+(W)$  with  $\text{cap}(\delta^+(W)) = \text{value}(f)$ .

**description of the algorithm:** For any pair  $a = (v, w)$  define  $a^{-1} := (w, v)$ . Make an auxiliary graph  $D_f = (V, A_f)$  by the following rule: for any arc  $a \in A$ ,

$$(11) \quad \begin{aligned} \text{if } f(a) < c(a) \text{ then } a \in A_f, \\ \text{if } f(a) > 0 \text{ then } a^{-1} \in A_f. \end{aligned}$$

So if  $0 < f(a) < c(a)$  then both  $a$  and  $a^{-1}$  are arcs of  $A_f$ .

Now there are two possibilities:

$$(12) \quad \begin{aligned} \text{Case 1: } & \text{There exists an } r - s \text{ path in } D_f, \\ \text{Case 2: } & \text{There is no } r - s \text{ path in } D_f. \end{aligned}$$

**Case 1:** There exists an  $r - s$  path  $P = (v_0, a_1, v_1, \dots, a_t, v_t)$  in  $D_f = (V, A_f)$ .

So  $v_0 = r$  and  $v_t = s$ . We may assume that  $v_0, v_1, \dots, v_t$  are all distinct.

As  $a_1, \dots, a_t$  belong to  $A_f$ , we know by (11) that for each  $i = 1, \dots, t$ :

$$(13) \quad \begin{aligned} \text{either (i) } & a_i \in A \text{ and } \sigma_i := c(a_i) - f(a_i) > 0 \\ \text{or (ii) } & a_i^{-1} \in A \text{ and } \sigma_i := f(a_i^{-1}) > 0. \end{aligned}$$

Define  $\varepsilon := \min\{\sigma_1, \dots, \sigma_t\}$ . So  $\varepsilon > 0$ . Let  $f' : A \rightarrow \mathbb{R}_+$  be defined by, for  $a \in A$ :

$$(14) \quad \begin{aligned} f'(a) & := f(a) + \varepsilon, & \text{if } a = a_i \text{ for some } i = 1, \dots, t; \\ & := f(a) - \varepsilon, & \text{if } a = a_i^{-1} \text{ for some } i = 1, \dots, t; \\ & := f(a), & \text{for all other } a. \end{aligned}$$

Then  $f'$  again is an  $r - s$  flow subject to  $c$ . The inequalities  $0 \leq f'(a) \leq c(a)$  hold because of our choice of  $\varepsilon$ . It is easy to check that also the flow conservation law is maintained.

Moreover,

$$(15) \quad \text{value}(f') = \text{value}(f) + \varepsilon,$$

since either  $(v_0, v_1) \in A$ , in which case the outgoing flow in  $r$  is increased by  $\varepsilon$ , or  $(v_1, v_0) \in A$ , in which case the ingoing flow in  $r$  is decreased by  $\varepsilon$ .

Path  $P$  is called a *flow augmenting path*.

**Case 2:** *There is no path in  $D_f = (V, A_f)$  from  $r$  to  $s$ .*

Now define:

$$(16) \quad W := \{w \in V \mid \text{there exists a path in } D_f \text{ from } r \text{ to } w\}.$$

Then  $r \in W$  while  $s \notin W$ , and so  $\delta^+(W)$  is an  $r - s$  cut.

By definition of  $W$ , if  $u \in W$  and  $v \notin W$ , then  $(u, v) \notin A_f$  (as otherwise also  $v$  would belong to  $W$ ). Therefore:

$$(17) \quad \begin{aligned} &\text{if } (u, v) \in \delta^+(W), \text{ then } (u, v) \notin A_f, \text{ and so (by (11)): } f(u, v) = c(u, v), \\ &\text{if } (u, v) \in \delta^-(W), \text{ then } (v, u) \notin A_f, \text{ and so (by (11)): } f(u, v) = 0. \end{aligned}$$

Then (10) gives:

$$(18) \quad \text{cap}(\delta^+(W)) = \text{value}(f). \quad \blacksquare$$

This finishes the description of the Flow augmenting algorithm. The description of the Maximum flow algorithm is now simple:

### Maximum flow algorithm

**input:** directed graph  $D = (V, A), r, s \in V, c : A \rightarrow \mathbb{R}_+$ .

**output:** a maximum  $r - s$  flow  $f$  and an  $r - s$  cut  $\delta^+(W)$  of minimum capacity, with  $\text{value}(f) = \text{cap}(\delta^+(W))$ .

**description of the algorithm:** Let  $f_0$  be the ‘null flow’ (that is,  $f_0(a) = 0$  for each arc  $a$ ). Determine with the Flow augmenting algorithm flows  $f_1, f_2, \dots, f_N$  such that  $f_{i+1} = f'_i$ , until, in the  $N$ th iteration, say, we obtain output (ii) of the Flow augmenting algorithm. Then we have flow  $f_N$  and a cut  $\delta^+(W)$  with the given properties.  $\blacksquare$

The following is easy to prove:

**Theorem 1.** *If all capacities  $c(a)$  are integer, the algorithm terminates.*

**Proof.** In the flow augmentation iterations, every flow  $f_i$  and every  $\varepsilon$  is integer-valued. So at each iteration, the flow value increases by at least 1. Since the flow value cannot exceed  $\text{cap}(\delta^+(\{r\}))$ , we can have only finitely many iterations.  $\blacksquare$

Note that in the Maximum flow algorithm, if all capacities are integer, then the maximum flow will also be integer-valued.

**Exercise 2.** Derive from Theorem 1 that also if all capacities are rational then the algorithm terminates.

On the other hand, the algorithm does not need to terminate for general real-valued capacities (see Ford and Fulkerson [7]).

Yet also for generally real-valued capacities the following basic theorem of Ford and Fulkerson [5] can be proved:

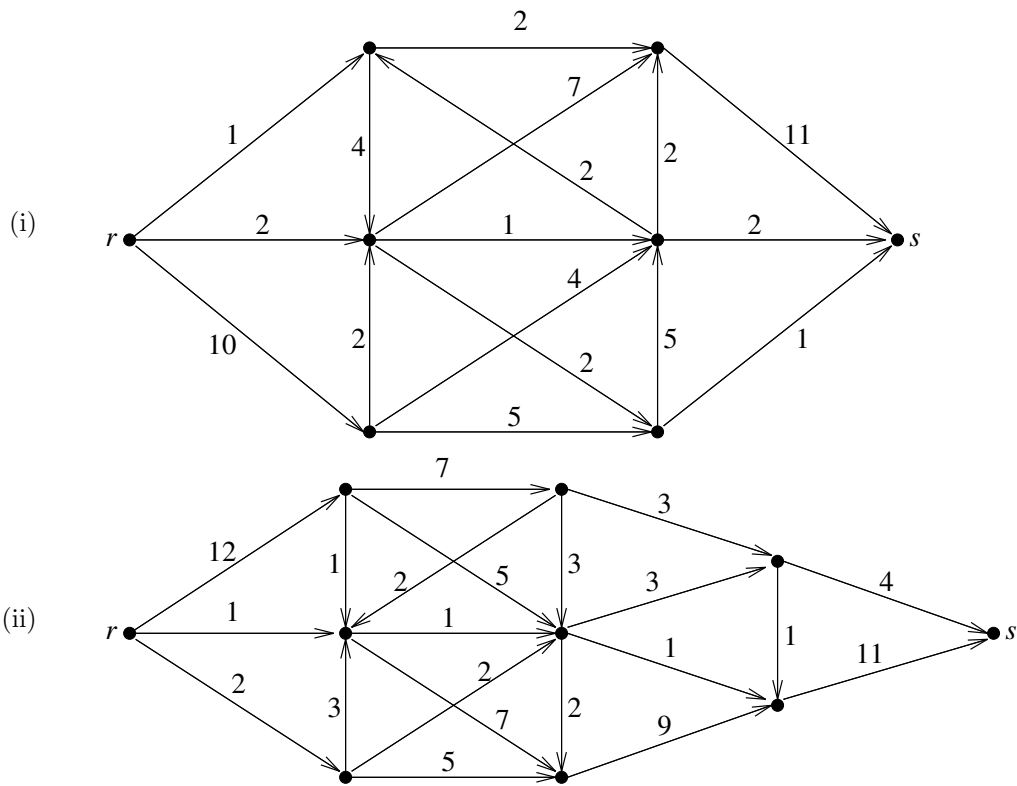
**Theorem 2** (Max-flow min-cut theorem). *The maximal value of an  $r - s$  flow subject to  $c$  is equal to the minimal capacity of an  $r - s$  cut. In formula:*

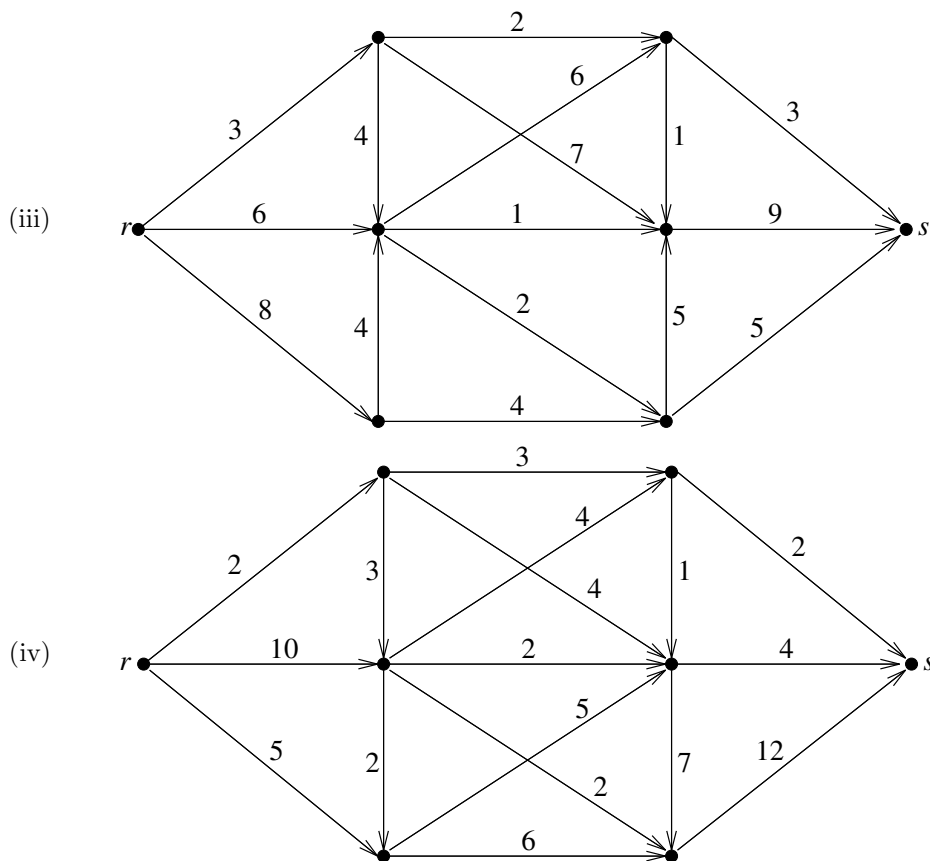
$$(19) \quad \max_{f \text{ } r-s \text{ flow subject to } c} \text{value}(f) = \min_{\delta^+(U) \text{ } r-s \text{ cut}} \text{cap}(\delta^+(U)).$$

*If the capacities are integer, there exists an integer-valued maximum flow.*

If we choose always a shortest  $r - s$  path in  $D_f$  as our flow augmenting path  $P$  (that is, with a minimum number of arcs), then the number of iterations is at most  $|V| \cdot |A|$ . This was shown by Dinits [2] and Edmonds and Karp [3].

**Exercise 3.** Determine with the maximum flow algorithm an  $r - s$  flow of maximum value and an  $r - s$  cut of minimum capacity in the following graphs (where the numbers at the arcs give the capacities):





For background literature on network flows we refer to the books by Ford and Fulkerson [7] and Ahuja, Magnanti, and Orlin [1].

### References

- [1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, *Network Flows — Theory, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
- [2] E.A. Dinitz, Algorithm for solution of a problem of maximum flow in a network with power estimation (in Russian), *Doklady Akademii Nauk SSSR* 194 (1970) 754–757 [English translation: *Soviet Mathematics Doklady* 11 (1970) 1277–1280].
- [3] J. Edmonds and R.M. Karp, Theoretical improvements in algorithmic efficiency for network flow problems, *Journal of the Association for Computing Machinery* 19 (1972) 248–264.
- [4] P. Elias, A. Feinstein, and C.E. Shannon, A note on the maximum flow through a network, *IRE Transactions on Information Theory* IT-2 (1956) 117–119.
- [5] L.R. Ford, Jr and D.R. Fulkerson, Maximal flow through a network, *Canadian Journal of Mathematics* 8 (1956) 399–404.
- [6] L.R. Ford, Jr and D.R. Fulkerson, A simple algorithm for finding maximal network flows and an application to the Hitchcock problem, *Canadian Journal of Mathematics* 9 (1957) 210–218.
- [7] L.R. Ford, Jr and D.R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, New Jersey, 1962.