NDMI011: Combinatorics and Graph Theory 1

Lecture #6
Flows and cuts in networks

Irena Penev

1 Network flows and cuts

A *network* is an ordered four-tuple (G, s, t, c), where G is an oriented graph, s and t are two distinct vertices of this graph (called the *source* and *sink*, respectively), and $c : E(G) \to [0, +\infty)$ is a function, called the *capacity function* (see Figure 1.1 for an example). The *capacity* of an edge $e \in E(G)$ is the number $c(e)$.

Networks can be used to model, for example, a system of pipes used to transport some resource, such as water or oil; capacities would be the number of units of volume that a given pipe can transport per unit time.

A *feasible flow* (or simply *flow*) in a network (G, s, t, c) is a function $f : E(G) \to [0, +\infty)$ that satisfies the following two properties (see Figure 1.2 for an example):

- $f(e) \leq c(e)$ for all $e \in E(G)$;[1]
- for all $v \in V(G) \setminus \{s, t\}$, we have $\sum\limits_{(x,v) \in E(G)} f(x, v) = \sum\limits_{(v,y) \in E(G)} f(v, y)$.[2]

The *value* of a flow f is

$$val(f) = \Big(\sum_{(s,x) \in E(G)} f(s, x) \Big) - \Big(\sum_{(x,s) \in E(G)} f(x, s) \Big).$$

A *maximum flow* in (G, s, t, c) is a flow f^* that has maximum value, i.e. one that satisfies $val(f) \leq val(f^*)$ for all flows f.

Theorem 1.1. *Every network (G, s, t, c) has a maximum flow.*

[1] This means that flow cannot be higher than capacity.

[2] This means that, for each vertex other than the source and the sink, the in-flow is equal to the out-flow. This condition is called the *conservation of flow* condition.

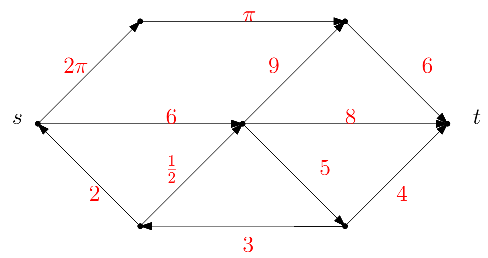


Figure 1.1: A network with capacities in red.

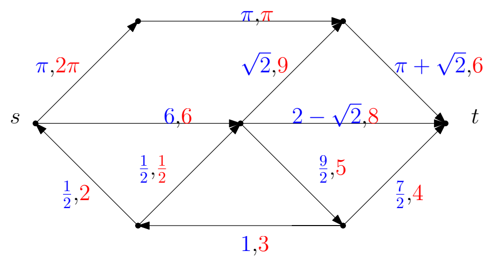


Figure 1.2: A network flow. Flows are in blue and capacities are in red.

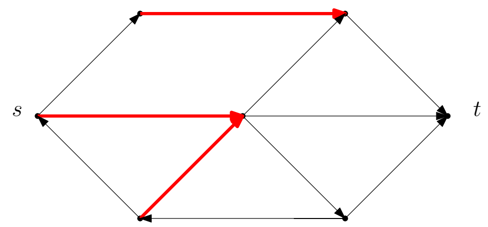


Figure 1.3: A cut in a network. (The edges of the cut are in red.)

Proof. Omitted. □

Theorem 1.1 should certainly seem plausible, and yet it is not entirely obvious how one might prove it (since the number of flows is, typically, infinite). The proof relies on certain results from analysis, which we omit.

An s,t-*cut*, or simply *cut*, in a network (G,s,t,c) is a set $R \subseteq E(G)$ such that $G \setminus R$ contains no directed path from s to t (see Figure 1.3 for an example). The *capacity* of the cut R is $c(R) = \sum_{e \in R} c(e)$.

Our main theorem (proven in the next section) is the following.

Max-flow min-cut theorem. *The maximum value of a flow in a network is equal to the minimum capacity of a cut in that network.*

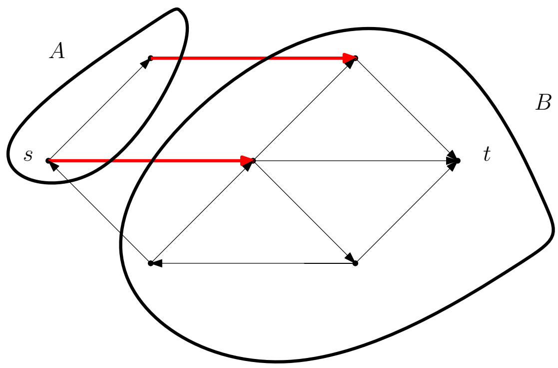


Figure 2.1: A cut $S(A, B)$ in a network. (The edges of the cut are in red.)

2 Proof of the Max-flow min-cut theorem

We now need some terminology and notation. First, for a network (G, s, t, c), a flow f in that network, and a set of edges $R \subseteq E(G)$, we write

- $c(R) = \sum_{e \in R} c(e)$;
- $f(R) = \sum_{e \in R} f(e)$.

Next, for a directed graph G and disjoint sets $A, B \subseteq V(G)$, we set

$$S(A, B) = \{(a, b) \in E \mid a \in A, b \in B\}.$$

Thus, $S(A, B)$ is the set of all arcs from A to B (see Figure 2.1 for an example).[3]

For a network (G, s, t, c), disjoint sets $A, B \subseteq V(G)$, and a flow f, we write

- $c(A, B) = c(S(A, B))$;[4]
- $f(A, B) = f(S(A, B))$.

Proposition 2.1. *Let (G, s, t, c) be a network, and let (A, B) be a partition of $V(G)$ such that $s \in A$ and $t \in B$. Then $S(A, B)$ is a cut in (G, s, t, c).*

Proof. Let $P = p_0, p_1, \ldots, p_\ell$, with $p_0 = s$ and $p_\ell = t$, be a directed path in G. By hypothesis, $p_0 = s \in A$ and $p_\ell = t \in B$; let $i \in \{0, \ldots, \ell - 1\}$ be maximum with the property that $p_i \in A$. Then $p_{i+1} \in B$, and see that

[3] $S(A, B)$ does not contain arcs from B to A!

[4] According to our notation, $c(S(A, B)) = \sum_{e \in S(A,B)} c(e)$, i.e. $c(A, B)$ is the sum of capacities of all the edges from A to B.

$(p_i, p_{i+1}) \in S(A, B)$, i.e. the directed path P uses an edge of $S(A, B)$. Since the path P was chosen arbitrarily, it follows $G \setminus R$ contains no directed paths from s to t, and so $S(A, B)$ is indeed a cut of (G, s, t, c). □

Proposition 2.2. *Let (G, s, t, c) be a network, and let R be a cut in this network. Then there exists a partition (A, B) of $V(G)$ such that $s \in A$, $t \in B$, and $S(A, B) \subseteq R$.*[5]

Proof. Let A be the set of all vertices $v \in V(G)$ such that $G \setminus R$ contains a directed path from s to v, and set $B = V(G) \setminus A$. Clearly, $s \in A$ and $t \in B$.[6] We now claim that $S(A, B) \subseteq R$. Suppose otherwise, and fix an edge $(x, y) \in S(A, B) \setminus R$. (In particular, $y \in B$.) Let $P = p_0, \dots, p_\ell$, with $p_0 = s$ and $p_\ell = x$, be a directed path in $G \setminus R$. Since $(x, y) \notin R$, we then have that $p_0, \dots, p_\ell, y$ is a directed path from s to y in $G \setminus R$, and so by construction, we have that $y \in A$, contrary to the fact that $y \in B$. □

Lemma 2.3. *Let f be a flow in a network (G, s, t, c), and let (A, B) be a partition of $V(G)$ such that $s \in A$ and $t \in B$. Then $val(f) = f(A, B) - f(B, A)$. In particular,*[7] *we have that* $val(f) = \Big(\sum\limits_{(x,t) \in E(G)} f(x, t) \Big) - \Big(\sum\limits_{(t,x) \in E(G)} f(t, x) \Big)$.

Proof. By the definition of a flow, for all vertices $v \in A \setminus \{s\}$, we have that

$$\Big(\sum_{(v,x) \in E(G)} f(v, x) \Big) - \Big(\sum_{(x,v) \in E(G)} f(x, v) \Big) = 0,$$

and consequently,

$$\sum_{v \in A \setminus \{s\}} \left(\Big(\sum_{(v,x) \in E(G)} f(v, x) \Big) - \Big(\sum_{(x,v) \in E(G)} f(x, v) \Big) \right) = 0,$$

On the other hand, for the source s, we have that

$$\Big(\sum_{(s,x) \in E(G)} f(s, x) \Big) - \Big(\sum_{(x,s) \in E(G)} f(x, s) \Big) = val(f).$$

By adding the last two equalities, we get

$$\sum_{v \in A} \left(\Big(\sum_{(v,x) \in E(G)} f(v, x) \Big) - \Big(\sum_{(x,v) \in E(G)} f(x, v) \Big) \right) = val(f).$$

[5] Note that this implies that $c(A, B) \leq c(R)$. Thus, our proof of the Max-flow min-cut theorem, it will be enough to consider cuts of the form $S(A, B)$, where (A, B) is a partition of $V(G)$, with $s \in A$ and $t \in B$; cuts of this form are sometimes called *elementary cuts.*

[6] The fact that $t \notin A$ follows from the fact that R is a cut in (G, s, t, c), and so there are no directed paths from s to t in $G \setminus R$; so, $t \in B$.

[7] This happens if we take $A = V(G) \setminus \{t\}$ and $B = \{t\}$.

Note that for each edge $(u_1, u_2) \in E(G)$ such that $u_1, u_2 \in A$, the term $f(u_1, u_2)$ appears exactly twice in the sum above: once with the $+$ sign,[8] and one with the $-$ sign.[9] After we cancel out such terms, what remains is precisely $f(A, B) - f(B, A) = val(f)$, which is what we needed to show. $\square$

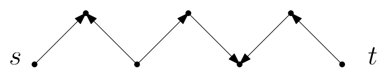
Corollary 2.4. *Let f be a flow in a network (G, s, t, c), and let R be a cut. Then $val(f) \leq c(R)$.*

Proof. By Proposition 2.2, there exists a partition (A, B) of $V(G)$ such that $s \in A$, $t \in B$, and $S(A, B) \subseteq R$. Then

$$\begin{aligned}
val(f) &= f(A,B) - f(B,A) && \text{by Lemma 2.3} \\
&\leq f(A,B) && \text{because } f(e) \geq 0 \text{ for all } e \in E(G) \\
&\leq c(A,B) && \text{because } f(e) \leq c(e) \text{ for all } e \in E(G) \\
&\leq c(R) && \text{because } S(A,B) \subseteq R \text{ and} \\
& && \text{and } c(e) \geq 0 \text{ for all } e \in E(G)
\end{aligned}$$

which is what we needed to show. $\square$

We now introduce a key new concept: that of an "augmenting path." First, an *(s, t)-path* in a network (G, s, t, c) is a sequence $v_0, v_1, \dots, v_\ell$ of vertices of G such that $v_0 = s$, $v_\ell = t$, and for all $i \in \{0, \dots, \ell - 1\}$, we have that one of (v_i, v_{i+1}) and (v_{i+1}, v_i) belongs to $E(G)$. Note that an (s, t)-path may, but need not be, a directed (s, t)-path (see the figure below for an example).



Now, given a flow f in the network (G, s, t, c), an (s, t)-path $v_0, v_1, \dots, v_\ell$ in (G, s, t, c) is said to be an *f-augmenting path* if the following two conditions are satisfied (see Figure 2.2 for an example):

- for all $i \in \{1, \dots, \ell - 1\}$ such that $(v_i, v_{i+1}) \in E(G)$, we have that $f(v_i, v_{i+1}) < c(v_i, v_{i+1})$;
- for all $i \in \{1, \dots, \ell - 1\}$ such that $(v_{i+1}, v_i) \in E(G)$, we have that $f(v_{i+1}, v_i) > 0$.

[8]For this, we take $v = u_1$, $x = u_2$, and $(v, x) \in E(G)$ to add $f(u_1, u_2)$ (via the first sum)

[9]For this, we take $v = u_2$, $x = u_1$, and $(x, v) \in E(G)$ to subtract $f(u_1, u_2)$ (via the second sum).

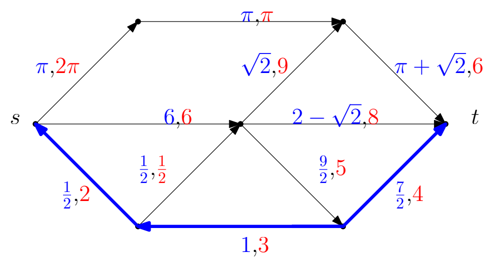


Figure 2.2: An f-augmenting path (edges in blue) in a network (G, s, t, c). (Flow is in blue and capacities are in red.)

Lemma 2.5. *Let f be a flow in a network (G, s, t, c). Then f is a maximum flow if and only if there does not exist an f-augmenting path in (G, s, t, c). Furthermore, if f is a maximum flow, then there exists a cut R in (G, s, t, c) such that $val(f) = c(R)$.*

Proof. It suffices the prove the following two statements:

(a) if there exists an f-augmenting path in (G, s, t, c), then f is not a maximum flow in (G, s, t, c);

(b) if there does not exist an f-augmenting path in (G, s, t, c), then f is a maximum flow in (G, s, t, c), and furthermore, there exists a cut R in (G, s, t, c) such that $val(f) = c(R)$.

We first prove (a). Suppose that $v_0, \dots, v_\ell$ (with $v_0 = s$ and $v_\ell = t$) an f-augmenting path in (G, s, t, c). Now, set

- $\varepsilon_1 = \min\Big(\{c(v_i, v_{i+1}) - f(v_i, v_{i+1}) \mid 0 \leq i \leq \ell - 1, (v_i, v_{i+1}) \in E(G)\} \cup \{\infty\}\Big)$;
- $\varepsilon_2 = \min\Big(\{f(v_{i+1}, v_i) \mid 0 \leq i \leq \ell - 1, (v_{i+1}, v_i) \in E(G)\} \cup \{\infty\}\Big)$;
- $\varepsilon = \min\{\varepsilon_1, \varepsilon_2\}$.[10]

Since $v_0, \dots, v_\ell$ is an f-augmenting path, we have that $\varepsilon_1, \varepsilon_2 > 0$, and consequently, $\varepsilon > 0$. We now define a new flow f' as follows:

- $f'(v_i, v_{i+1}) = f(v_i, v_{i+1}) + \varepsilon$ for all $i \in \{0, \dots, \ell - 1\}$ such that $(v_i, v_{i+1}) \in E(G)$;[11]

[10] The reason we have ∞ in the definition of ε_1 and ε_2 is because our f-augmenting path may have only "with-the-flow" or only "against-the-flow" edges, and we cannot take the minimum of an empty set. Note, however, that at least one of ε_1 and ε_2 is a real number (and not ∞), and consequently, ε is a real number.

[11] So, for edges on our augmenting path directed with the flow, we increase the flow by ε.

- $f'(v_{i+1}, v_i) = f(v_{i+1}, v_i) - \varepsilon$ for all $i \in \{0, \dots, \ell-1\}$ such that $(v_{i+1}, v_i) \in E(G)$;[12]

- $f'(e) = f(e)$ for all other edges e.

It is easy to verify that f' is indeed a feasible flow.[13] Furthermore, by construction, $val(f') = val(f) + \varepsilon$, and so (since $\varepsilon > 0$) we have that $val(f') > val(f)$, and so f is not a maximum flow in (G, s, t, c).

It remains to prove (b). For this, we suppose that (G, s, t, c) does not admit an f-augmenting path, and we show that f is a maximum flow. Let A be the set of all vertices $v \in V(G)$ such that there exists a path $v_0, \dots, v_\ell$ with $v_0 = s$ and $v_\ell = v$, and satisfying the following properties:[14]

- for all $i \in \{1, \dots, \ell-1\}$ such that $(v_i, v_{i+1}) \in E(G)$, we have that $f(v_i, v_{i+1}) < c(v_i, v_{i+1})$;

- for all $i \in \{1, \dots, \ell-1\}$ such that $(v_{i+1}, v_i) \in E(G)$, we have that $f(v_{i+1}, v_i) > 0$.

Set $B = V(G) \setminus A$. Clearly, $s \in A$ and $t \in B$.[15] Further, for all $x \in A$ and $y \in B$,

- if $(x, y) \in E(G)$, then $f(x, y) = c(x, y)$, and

- if $(y, x) \in E(G)$, then $f(y, x) = 0$.[16]

Note that this implies that $f(A, B) = c(A, B)$ and $f(B, A) = 0$. But now we have that

$$\begin{array}{rcll} val(f) & = & f(A,B) - f(B,A) & \text{by Lemma 2.3} \\ & = & c(A,B) & \text{because } f(A,B) = c(A,B) \\ & & & \text{and } f(B,A) = 0 \end{array}$$

It now follows from Corollary 2.4 that f is a maximum flow in (G, s, t, c).[17] Furthermore, by Proposition 2.1, we know that $R := S(A, B)$ is a cut, and by what we just showed, $val(f) = c(A, B) = c(R)$. □

[12] So, for edges on our augmenting path directed against the flow, we decrease the flow by ε.

[13] Check this!

[14] Essentially, but somewhat informally, we are choosing A to be the set of all vertices $v \in V(G)$ such that there exists an f-augmenting path from s to v.

[15] If we had $t \in A$, then by the construction of A, there would be an f-augmenting path in (G, s, t, c).

[16] Otherwise, there would be an f-augmenting path from s to y, contrary to the fact that $y \notin A$.

[17] Indeed, suppose f' is any flow in (G, s, t, c). Then by Corollary 2.4, we have that $val(f') \leq cap(A, B)$, and so by what we just showed, $val(f') \leq val(f)$.

We are now ready to prove the Max-flow min-cut theorem, restated below.

Max-flow min-cut theorem. *The maximum value of a flow in a network is equal to the minimum capacity of a cut in that network.*

Proof. Let (G, s, t, c) be a network, and let f be a maximum flow in it (the existence of such a flow is guaranteed by Theorem 1.1). By Lemma 2.5, there exists a cut R in (G, s, t, c) such that $val(f) = c(R)$. Furthermore, for any cut R' in (G, s, t, c), Corollary 2.4 guarantees that $val(f) \leq c(R')$, and consequently, $c(R) \leq c(R')$; thus, R is a cut of minimum capacity in (G, s, t, c). $\square$

3 The Ford-Fulkerson algorithm

The proof of Lemma 2.5 can easily be converted into an algorithm that finds a maximum flow and a minimum capacity of a cut in an input network. The idea is to repeatedly find augmenting paths and update the flow (increasing its value). When no augmenting path exists, we instead find a cut whose capacity is equal to the value of our flow, which (by Corollary 2.4) guarantees that this cut is of minimum capacity.

Suppose that f is a flow in a network (G, s, t, c). We now either find an f-augmenting path in (G, s, t, c), or we find a cut whose capacity is $val(f)$, as follows:

1. Set $A := \{s\}$.
2. While $t \notin A$:
   (a) Either find vertices $x \in A$ and $y \in V(G) \setminus A$ such that
      - $(x, y) \in E(G)$ and $f(x, y) < c(x, y)$, or
      - $(y, x) \in E(G)$ and $f(y, x) > 0$,

      or determine that such x and y do not exist.

   (b) If we found x and y, then we set backpoint$(y) = x$, and we update $A := A \cup \{y\}$.
   (c) Otherwise, we stop and return the cut $S(A, V(G) \setminus A)$.[18]
3. Construct an f-augmenting path by following backpoints starting from t, and return this path.

Example 3.1. *Consider the flow f in the network (G, s, t, c) in Figure 3.1. Either find an f-augmenting path, or find a cut whose capacity is $val(f)$.*

[18] In this case, an argument analogous to the proof of Lemma 2.5 guarantees that $c(A, V(G) \setminus A) = val(f)$.

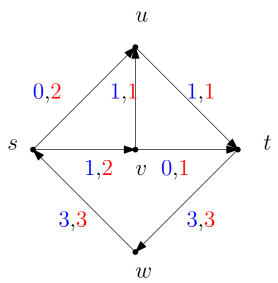


Figure 3.1: The network and flow from Example 3.1. Flows are in blue and capacities in red.

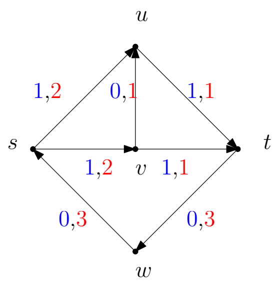


Figure 3.2: The network and flow from Example 3.2. Flows are in blue and capacities in red.

Solution. We begin with $A = \{s\}$. We now iterate several times.

1. We select $s \in A$ and $u \in V(G) \setminus A$, and we set $A := \{s, u\}$ and $\text{backpoint}(u) = s$.

2. We select $s \in A$ and $w \in V(G) \setminus A$, and we set $A := \{s, u, w\}$ and $\text{backpoint}(w) = s$.

3. We select $u \in A$ and $v \in V(G) \setminus A$, and we set $A := \{s, u, w, v\}$ and $\text{backpoint}(v) = u$.

4. We select $v \in A$ and $t \in V(G) \setminus A$, and we set $A := \{s, u, w, v, t\}$ and $\text{backpoint}(t) = v$.

We now reconstruct our f-augmenting path: s, u, v, t. (It is easy to see that this really is an f-augmenting path.) $\square$

Example 3.2. *Consider the flow f in the network (G, s, t, c) in Figure 3.2. Either find an f-augmenting path, or find a cut whose capacity is $val(f)$.*

Solution. We begin with $A = \{s\}$. We now iterate several times.

1. We select $s \in A$ and $u \in V(G) \setminus A$, and we set $A := \{s, u\}$ and $\text{backpoint}(u) = s$.
2. We select $s \in A$ and $v \in V(G) \setminus A$, and we set $A := \{s, u, v\}$ and $\text{backpoint}(v) = s$.

There are now no further vertices that we can select, and $t \notin A$. We now see that $S(A, V(G) \setminus A) = \{(u, t), (v, t)\}$ is a cut whose capacity is 2, which is precisely equal to $val(f)$. $\square$

We now describe the Ford-Fulkerson algorithm, which finds a maximum flow in a network (G, s, t, c). Its steps are as follows:

1. Set $f(e) := 0$ for all $e \in E(G)$.
2. While there exists an f-augmenting path in the network:
   (a) Find an f-augmenting path $v_0, \dots, v_\ell$ (with $v_0 = s$ and $v_\ell = t$).
   (b) Set
   - $\varepsilon_1 = \min\Big(\{c(v_i, v_{i+1}) - f(v_i, v_{i+1}) \mid 0 \le i \le \ell - 1, (v_i, v_{i+1}) \in E(G)\} \cup \{\infty\}\Big)$;
   - $\varepsilon_2 = \min\Big(\{f(v_{i+1}, v_i) \mid 0 \le i \le \ell - 1, (v_{i+1}, v_i) \in E(G)\} \cup \{\infty\}\Big)$;
   - $\varepsilon = \min\{\varepsilon_1, \varepsilon_2\}$.

   (c) Update f as follows:
   - $f(v_i, v_{i+1}) := f(v_i, v_{i+1}) + \varepsilon$ for all $i \in \{0, \dots, \ell - 1\}$ such that $(v_i, v_{i+1}) \in E(G)$;[19]
   - $f(v_{i+1}, v_i) := f(v_{i+1}, v_i) - \varepsilon$ for all $i \in \{0, \dots, \ell - 1\}$ such that $(v_{i+1}, v_i) \in E(G)$.[20]
3. Return f.

Example 3.3. *Find a maximum flow and an a cut of minimum capacity in the network represented in Figure 3.3.*

Solution. We first set $f(e) = 0$ for all $e \in E(G)$ (see the figure below, with flow in blue and capacities in red).

[19] So, for edges on our augmenting path directed with the flow, we increase the flow by ε.
[20] So, for edges on our augmenting path directed against the flow, we decrease the flow by ε.

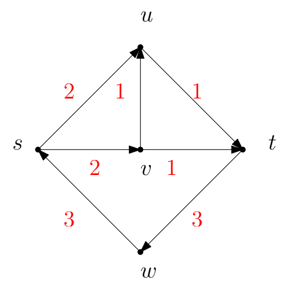
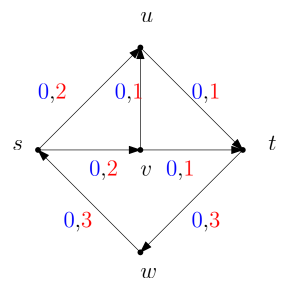
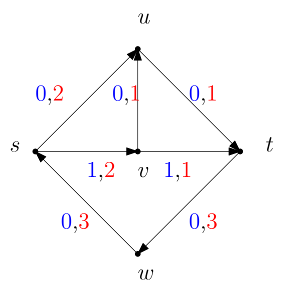


Figure 3.3: The network from Example 3.3.

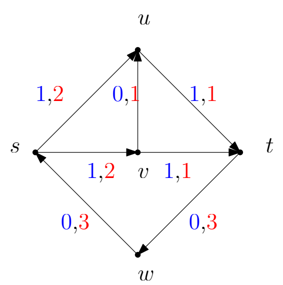


We now iterate several times.

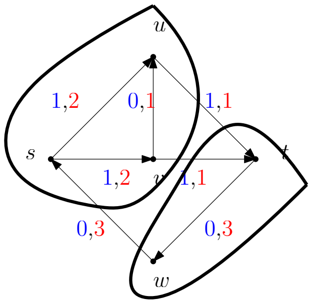
1. We find an augmenting path s, v, t, we get $\varepsilon = 1$, and we update f as in the picture below (flow in blue and capacities in red).



2. We find an augmenting path s, u, t, we get $\varepsilon = 1$, and we update f as in the picture below (flow in blue and capacities in red).



3. We find a cut $S(\{s,u,v\},\{w,t\}) = \{(u,t),(v,t)\}$ of capacity is 2, which is precisely equal to $val(f)$.



The flow f is a maximum flow, and the cut $S(\{s,u,v\},\{w,t\}) = \{(u,t),(v,t)\}$ is a minimum capacity cut. $\square$