

5 Flows and cuts in digraphs

Recall that a **digraph** or **network** is a pair $\vec{G} = (V, \vec{E})$ where V is a set and \vec{E} is a multiset of ordered pairs of elements of V , which we refer to as **arcs**. Note that two vertices can be joined by many arcs in either direction. In a digraph $\vec{G} = (V, \vec{E})$, let $\Gamma^+(v)$ and $\Gamma^-(v)$ denote the sets of vertices adjacent **from** v and **to** v , respectively. These are the **out-neighborhood** of v and the **in-neighborhood** of v respectively. Thus

$$\Gamma^+(v) = \{u : (v, u) \in \vec{E}\} \quad \Gamma^-(v) = \{u : (u, v) \in \vec{E}\}.$$

For example, in the digraph drawn below, we have $\Gamma^+(x) = \{u, v, w\}$, $\Gamma^-(x) = \{v\}$.

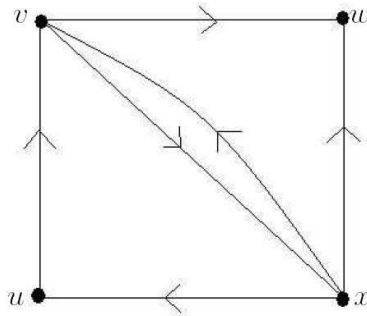


Figure 1 : A digraph

5.1 Flows

Let $\vec{G} = (V, \vec{E})$ be a digraph and $s, t \in V$. We shall refer to s as the source vertex and t as the sink vertex in what follows. Let $\mathbb{R}_{\geq 0}$ denote the non-negative real numbers. We want to define what it means for the network to have a flow from s to t . A **static st -flow** is a function $f : \vec{E} \rightarrow \mathbb{R}_{\geq 0}$ such that for all $x \in V \setminus \{s, t\}$:

$$\sum_{y \in \Gamma^+(x)} f(x, y) = \sum_{y \in \Gamma^-(x)} f(y, x)$$

This last requirement is known as **Kirchoff's Law**, in words it states that the flow into a vertex must be equal to the flow out of a vertex, and that this must be true for all vertices apart from s and t . The flow in an arc $e \in \vec{E}$ is denoted $f(e)$. The **value** of the flow f from s to t is defined by

$$v(f) = \sum_{y \in \Gamma^+(s)} f(s, y) - \sum_{y \in \Gamma^-(s)} f(y, s).$$

This is the net amount of flow leaving the source s . An example of all these concepts is given below. The flows f through the arcs in the networks equals the number of arrows on each arc. So $f(s, v) = 2 = f(v, w) = f(u, x)$, $f(s, u) = 3$, $f(w, t) = 4$, $f(u, w) = f(x, w) = f(x, t) = 1$, $f(v, u) = 0$. The value of the flow is

$$v(f) = f(s, v) + f(s, u) = 5.$$

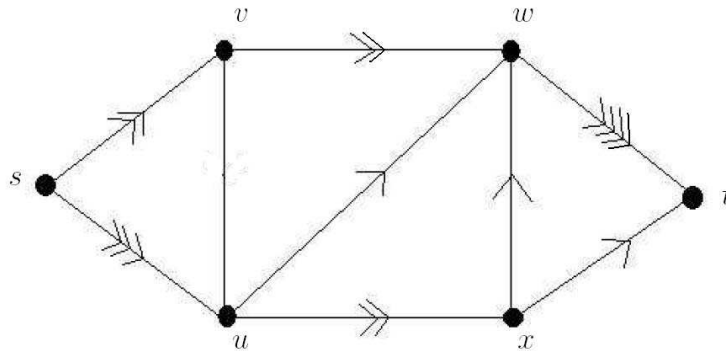


Figure 2 : Flow

5.2 Capacities

The **capacity** of an arc $e \in \vec{E}$ is a non-negative real number and denoted $c(e)$, and is the maximum possible amount of flow in the arc. An example of a network with capacities is shown below. Here $c(s, u) = c(v, w) = c(x, w) = c(v, u) = 1$ and $c(s, v) = c(u, w) = 4$ and $c(w, t) = 3$ and $c(u, x) = c(x, u) = c(x, t) = 2$.

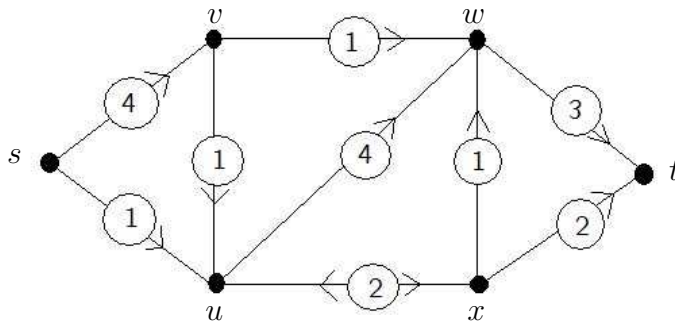


Figure 3 : Capacities

We require that if f is a flow in the network, then $f(e) \leq c(e)$ for every arc $e \in \vec{E}$. The flow in Figure 2 is impossible in Figure 3, since $f(s, u) = 3 > c(s, u) = 1$. However a valid flow would be to define $f(s, v) = f(u, x) = f(v, w) = f(w, t) = 2$, $f(s, u) = 3$,

$f(v, u) = 0$, $f(u, w) = f(x, w) = f(x, t) = 1$ and the value of this flow is 5. A **maximum flow** in a network with capacities is a flow f^* such that for every flow f , $v(f^*) \geq v(f)$ – it is a flow with largest value. It is not hard to see that the flow f above of value three in the above network is a maximum flow, since from $\{s, v\}$ to the rest of the network the total capacity is three.

5.3 Cuts

If $S \subset V$ and $s \in S$ and $t \notin S$, then the **cut induced by S** is the set of arcs from S to $V \setminus S$ – this is the set of arcs leaving S . This set of arcs is denoted (S, \bar{S}) , and is called an **st -cut** or simply a cut. For example, $\{(s, u), (v, u), (v, w)\}$ in Figure 3 is the cut induced by $S = \{s, v\}$. The **capacity** of a cut (S, \bar{S}) is defined by

$$c(S, \bar{S}) = \sum_{e \in (S, \bar{S})} c(e).$$

It is clear that if f is a flow in the network, then $v(f) \leq c(S, \bar{S})$ for any cut (S, \bar{S}) . In Figure 3, we found a flow with $v(f) = 3$, and also a cut (S, \bar{S}) with $c(S, \bar{S}) = 3$ – namely when $S = \{s, v\}$. A cut (S, \bar{S}) **minimum cut** is a cut (S, \bar{S}) such that $c(T, \bar{T}) \geq c(S, \bar{S})$ for every cut (T, \bar{T}) – so it is a cut with a minimum value of $c(S, \bar{S})$. Our main theorem says that the minimum capacity of a cut equals the maximum value of a flow – as already verified for the specific network in Figure 3. The proof will give us a way of finding a maximum flow. First we need a lemma:

Lemma 1* *For any flow f and any cut (S, \bar{S}) ,*

$$v(f) = \sum_{x \in S} \sum_{y \notin S} f(x, y) - \sum_{x \in S} \sum_{y \notin S} f(y, x).$$

Proof \triangleright By Kirchoff's Law, for all $x \notin \{s, t\}$,

$$\sum_{y \in \Gamma^+(x)} f(x, y) = \sum_{y \in \Gamma^-(x)} f(y, x)$$

Summing over $x \in S \setminus \{s\}$ we get

$$\sum_{x \in S \setminus \{s\}} \sum_{y \in \Gamma^+(x)} f(x, y) = \sum_{x \in S \setminus \{s\}} \sum_{y \in \Gamma^-(x)} f(y, x).$$

All those $f(x, y)$ with $x, y \in S$ cancel out: they are counted once in the left sum, and again once in the right sum. So we get

$$\sum_{x \in S \setminus \{s\}} \sum_{y \notin S} f(x, y) = \sum_{x \in S \setminus \{s\}} \sum_{y \in \Gamma^-(x)} f(y, x).$$

By definition of $v(f)$, this means

$$\sum_{x \in S} \sum_{y \notin S} f(x, y) - \sum_{x \in S} \sum_{y \notin S} f(y, x) - v(f) = 0$$

and this is the required result. ■

The definition of the value of the flow is for the specific cut (S, \bar{S}) with $S = \{s\}$, but this lemma says we can measure the value of a flow in any network by just looking at the net flow across a cut in the network. The main theorem we prove is the following:

Theorem 2 (Max-flow min-cut theorem for digraphs)

In any network with capacities, the maximum value of a flow equals the minimum value of a cut.

Proof \triangleright Let f be a maximum flow¹. Then $v(f) \leq c(S, \bar{S})$ for every cut (S, \bar{S}) so $v(f) \leq \min c(S, \bar{S})$. To prove the theorem, we define a set $S \subset V$ with $c(S, \bar{S}) = v(f)$. First put s , the source, into S . Then for every arc (x, y) such that $x \in S$ and $c(x, y) > f(x, y)$, put $y \in S$, and for every arc (y, x) with $x \in S$ and $f(y, x) > 0$, put $y \in S$. We claim $t \notin S$ and $c(S, \bar{S}) = v(f)$. Suppose that $t \in S$. Then there exists a path $x_1 x_2 \dots x_r$ where $x_i \in S$ for all i and $x_1 = s$ and $x_r = t$ and, by definition of S ,

$$c(x_i, x_{i+1}) - f(x_i, x_{i+1}) > 0 \quad \text{or} \quad f(x_{i+1}, x_i) > 0$$

for each i . Let ε be the smallest of all these positive numbers. Define a new flow g by taking $g(x_i, x_{i+1}) = f(x_i, x_{i+1}) + \varepsilon$ if $c(x_i, x_{i+1}) - f(x_i, x_{i+1}) > 0$, and taking $g(x_{i+1}, x_i) = f(x_{i+1}, x_i) - \varepsilon$ if $f(x_{i+1}, x_i) > 0$. Then $v(g) = v(f) + \varepsilon$, contradicting the maximality of f . We conclude that $t \notin S$. Finally, since $f(y, x) = 0$ for every $x \in S$ and $y \notin S$, by Lemma 1 we have

$$v(f) = \sum_{x \in S} \sum_{y \notin S} f(x, y) - \sum_{x \in S} \sum_{y \notin S} f(y, x) = \sum_{(x, y) \in (S, \bar{S})} c(x, y) = c(S, \bar{S}).$$

This completes the proof. ■

5.4 Variation of Max-Flow Min-Cut Theorem*

As a first variation, we can also prove a version of the Max-Flow Min-Cut Theorem for undirected graphs. If $G = (V, E)$ is an undirected graph, then a flow f in G is a function

¹Why does a maximum flow even exist? Prove that a maximum flow exists.

which assigns a non-negative real number to each edge of G and a direction to each edge of G . In other words, although the edges have no direction, the flow in an edge has direction. A flow again must satisfy Kirchoff's Law: the flow into a vertex must equal the flow out of that vertex. If $c : E \rightarrow \mathbb{R}_{\geq 0}$ defines the capacities of the edges of G , we insist that $f(e) \leq c(e)$, regardless of the direction of the flow. The capacity of an st -cut (S, \bar{S}) is the sum of the capacities of the edges between S and \bar{S} . Then the following theorem holds:

Theorem 3 (Max-flow min-cut theorem for graphs)

Let G be a graph with capacities on the edges. Then the maximum value of an st -flow equals the minimum capacity of an st -cut.

The proof is the same as the Max-Flow Min-Cut Theorem, so we leave it as an exercise. The next variation we discuss is the vertex-form of the Max-Flow Min-Cut Theorem. In the vertex form, there are no capacities on the edges, but there are capacities on the vertices. The capacity of a vertex is the amount of flow that is allowed to pass through that vertex. Let \vec{G} be a digraph and let s and t be a source and a sink in \vec{G} , respectively. Let $c : V \rightarrow \mathbb{R}_{\geq 0}$ be the set of capacities. Given a set $S \subset V$ containing s and not t , define

$$c(S) = \sum_{x \in S} c(x).$$

The vertex form of the Max-Flow Min-Cut Theorem is as follows:

Theorem 4 (Vertex-Form of Max-Flow Min-Cut Theorem)

In a network \vec{G} with vertex capacities, the maximum value of an st -flow equals the minimum capacity of an st -cut.

Proof \triangleright In fact this theorem follows from the Max-Flow Min-Cut Theorem by suitably altering \vec{G} . We can create a new digraph $\vec{H} = (W, \vec{F})$ from $\vec{G} = (V, \vec{E})$ as follows. First, replace each $x \in V$ with two vertices x^- and x^+ . Then add the arc (x^-, x^+) and give it capacity $c(x)$. Finally add an arc (x^+, y^-) whenever $(x, y) \in \vec{E}$ and assign all such arcs infinite capacity. Then we have a network \vec{H} with capacities on the arcs. This is shown below:

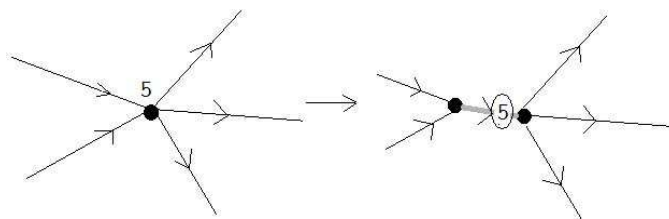


Figure 4 : Alteration

For each flow f in \vec{H} , we define a flow f^* in \vec{G} as follows: put $f^*(x, y) = f(x^+, y^-)$ for all $(x, y) \in \vec{E}$ – we leave it as an exercise to make sure this is actually a flow. We also leave the exercise to show that the minimum capacity of a cut in \vec{H} equals the smallest value of $c(S)$ in \vec{G} : that is

$$\min_{S \subset W} c(S, \bar{S}) = \min_{S \subset V} c(S).$$

So the maximum value of a flow f in \vec{G} is $\min c(S)$ by the Max-Flow Min-Cut Theorem. But $v(f) = v(f^*)$ and so we are done. ■

There are many other variants of the Max-Flow Min-Cut Theorem which we do not discuss here.

5.5 Algorithm

As we stated, the proof of the Max-Flow Min-Cut Theorem gives an algorithm for finding a maximum flow as well as a minimum cut. To construct a maximum flow f^* and a minimum cut (S^*, \bar{S}^*) , proceed as follows: start by letting f be the zero flow and $S = \{s\}$ where s is the source. Construct a set S as in the theorem: whenever there is an arc (x, y) such that $f(x, y) < c(x, y)$ and $x \in S$ and $y \notin S$, or an arc (y, x) such that $f(y, x) > 0$ and $x \in S$ and $y \notin S$, add y to S . If at the end of this procedure, $t \notin S$, then let $S^* = S$ to get a minimum cut and the current flow is a maximum flow. If at the end of this procedure $t \in S$, then there must be a path $x_0x_1x_2, \dots, x_r$ where $s = x_0$ and $t = x_r$, along which f can be augmented by some value $\varepsilon > 0$. The value of ε is given in the proof above: it is

$$\min\{c(x_i, x_{i+1}) - f(x_i, x_{i+1}), f(x_{i+1}, x_i) \mid 0 \leq i < r\}.$$

Now restart with the augmented flow which is $f(x_i, x_{i+1}) + \varepsilon$ and $c(x_i, x_{i+1}) > f(x_i, x_{i+1})$ and $f(x_{i+1}, x_i) - \varepsilon$ if $f(x_{i+1}, x_i) > 0$, for each $i : 0 \leq i < r$. Now we start again with $S = \{s\}$ and the new flow a input.

Example. Consider the network with capacities in Figure 3. According to the algorithm, start by letting f be the zero flow and $S = \{s\}$:

arc	flow	capacity
(s, u)	0	1
(s, v)	0	4
(v, w)	0	1
(x, u)	0	2
(u, x)	0	2
(u, w)	0	4
(x, t)	0	2
(w, t)	0	3

Since $f(s, v) = 0 < c(s, v) = 4$, we put v into S . Since $c(v, w) > f(v, w)$ we put w into S . Then put t into S since $c(w, t) > f(w, t)$. We stop since we have placed t in S . By the algorithm, there is a way to augment f : we consider the path $svwt$. We have the smallest difference between capacities and flows in the arcs of this path equal to 1. So we augment f to $f(s, v) = f(v, w) = f(w, t) = 1$. Now we start again with $S = \{s\}$ and the new flow.

arc	flow	capacity
(s, u)	0	1
(s, v)	1	4
(v, w)	1	1
(v, u)	0	1
(x, u)	0	2
(u, x)	0	2
(u, w)	0	4
(x, t)	0	2
(w, t)	1	3

Since $c(s, u) = 1$ and $f(s, u) = 0$, we add $u \in S$. Since $c(u, x) = 2$ and $f(u, x) = 0$, add $x \in S$. Since $c(x, t) = 2$ and $f(x, t) = 0$, add $t \in S$. So $S = \{s, u, x, t\}$ and since $t \in S$, we stop and we augment f by $\min\{1, 2, 2\} = 1$ along the path $suxt$ to get

arc	flow	capacity
(s, u)	1	1
(s, v)	1	4
(v, w)	1	1
(v, u)	0	1
(x, u)	0	2
(u, x)	1	2
(u, w)	0	4
(x, t)	1	2
(w, t)	1	3

Let $S = \{s\}$. Since $c(s, v) = 4$ and $f(s, v) = 1$, we can put $v \in S$. We cannot put $w \in S$ since $c(v, w) = 1 = f(v, w)$. But we can put $u \in S$ since $c(v, u) = 1$ and $f(v, u) = 0$. Then we can put $x \in S$ since $c(u, x) = 2$ and $f(u, x) = 1$. Finally we put $t \in S$ since $f(x, t) = 1$ and $c(x, t) = 2$. Since $t \in S$, we stop and augment f by $\min\{3, 1, 1, 1\} = 1$ along the path $svuxt$ to get

arc	flow	capacity
(s, u)	1	1
(s, v)	2	4
(v, w)	1	1
(v, u)	1	1
(x, u)	0	2
(u, x)	2	2
(u, w)	0	4
(x, t)	2	2
(w, t)	1	3

Let $S = \{s\}$. Since $c(s, v) = 4$ and $f(s, v) = 2$, we put $v \in S$. But now $c(v, w) = f(v, w) = 1$, $c(v, u) = f(v, u) = 1$ and $c(s, u) = f(s, u) = 1$. So $S = \{s, v\}$ and this induces a minimum cut (S, \bar{S}) . The flow we have just defined is a maximum flow, with value three, and notice $c(S, \bar{S}) = 3$, as expected.

5.6 Menger's Theorems

We prove the edge-form of Menger's Theorem for graphs here. Recall from §4 the statement of the edge-form of Menger's Theorem for graphs:

Theorem 5 (Edge form of Menger’s Theorem for graphs)

Let G be a graph and let s and t be distinct vertices of G . Then the minimum size of an st -edge-separator equals the maximum number of edge-disjoint st -paths.

Proof \triangleright A key point in the proof of the Max-Flow Min-Cut Theorem for both digraphs and graphs with integer capacities on the arcs or edges respectively, and with a maximum flow of value k , is the production of k paths from the source to the sink, one for each unit of flow. Note that the value of the flow is an integer if the capacities are integers, since the value ε by which successive flows are augmented in the proof of the Max-Flow Min-Cut Theorem is an integer. To prove the theorem, assign every edge of G a capacity of 1. Then clearly the minimum size of an st -edge-separator is the minimum capacity of an st -cut – let this be k . By the Max-Flow Min-Cut Theorem for graphs, this equals the value of a maximum st -flow in G . Now the number of edge-disjoint directed st -paths is clearly at most k , otherwise we would find an st -flow of greater value by sending one unit of flow independently along each of those paths. On the other hand, as we remarked above, there must be k paths from s to t along which each unit of flow is sent, and since each arc has unit capacity, none of these paths share any arcs, so there are at least k edge-disjoint directed st -paths. We conclude there are exactly k edge-disjoint directed st -paths. \blacksquare

It is not unexpected that we can prove a version of Menger’s Theorem for digraphs. First we need some terminology to explain what is meant by connectivity in digraphs. A **directed path** in a digraph is a digraph \vec{P} with vertex set $\{v_1, v_2, \dots, v_r\}$ and arc set $\{(v_i, v_{i+1}) : 1 \leq i < r\}$. We call this a directed v_1v_r -path. An **st -edge-separator** is a set of arcs whose removal from \vec{G} gives a digraph with no directed st -path. With these terms in mind, the following theorem has the same proof as the above theorem, except we use the Max-Flow Min-Cut Theorem for digraphs:

Theorem 6 (Arc form of Menger’s Theorem for digraphs)

Let \vec{G} be a digraph and let s and t be distinct vertices of \vec{G} . Then the minimum size of an st -edge-separator equals the maximum number of arc-disjoint directed st -paths.

A digraph is **connected** if and only if every pair (s, t) of vertices is joined by a directed st -path. More generally, a digraph is **k -arc-connected** if the removal of at most $k - 1$ arcs does not disconnect the digraph. According to the last theorem, a digraph is k -arc-connected if and only if every order pair of vertices (s, t) is joined by k arc-disjoint st -paths. The vertex-form of Menger’s Theorem for graphs and for digraphs follow from the vertex-form of the Max-Flow Min-Cut Theorem. We leave the details as an exercise.

5.7 Hall's Theorem

Another application of the max-flow min-cut theorem is to maximum matchings. Suppose that there are m applicants for n jobs, and each applicant is able to do some subset of the n jobs. On the other hand, only one person is required to do each job. The aim is to place as many applicants as possible in jobs that they are able to do. This scenario can be represented as a [bipartite graph](#): we have a set A of applicants and a set B of jobs, and we join an applicant $a \in A$ with an edge to job $b \in B$ if a is able to do job b . Then we want to find as many disjoint edges in this graph as possible.

We represent a bipartite graph $G = (V, E)$ with parts A and B as $G = (A \cup B, E)$. A [matching](#) in a bipartite graph $G = (A \cup B, E)$ is a set of vertex-disjoint edges in G . For example, a cycle of even length $2k$ is a bipartite graph and it contains a matching of size k – take alternate edges around the cycle. Similarly, the complete bipartite graph $K_{s,t}$ has a matching of size $\min\{s, t\}$, but no larger. A [maximum matching](#) in a bipartite graph is a matching with as many edges as possible. A [perfect matching](#) in a bipartite graph covers all the vertices – in $G = (A \cup B, E)$ a perfect matching has size $|A| = |B|$. In this section we give a theorem which says when a bipartite graph has a perfect matching.

Theorem 7 (Hall's Theorem) *A bipartite graph $G = (A \cup B, E)$ has a perfect matching if and only if for every non-empty set $X \subseteq A$ and every set $X \subseteq B$,*

$$|\Gamma(X)| \geq |X|.$$

The inequality in the theorem is often referred to as [Hall's Condition](#). Note that in one direction, the proof is easy: if G has a perfect matching then $|\Gamma(X)| \geq |X|$ for every $X \subseteq A$ and $X \subseteq B$: if M is the graph of all edges in a perfect matching, then $|\Gamma(X)| = |X|$ in this graph and so $|\Gamma(X)| \geq |X|$ in G . There are many proofs of the theorem. We will give two proofs: one is by induction and the other is using the Max-Flow Min-Cut Theorem.

Proof \triangleright (First Proof - By Induction) It is enough to prove that there is a matching covering all vertices of A , for then there must also be one covering all vertices of B , and we get $|A| = |B|$ and G has a perfect matching. We prove the theorem by induction on $|A|$. If $|A| = 1$, then $|\Gamma(A)| \geq |A|$ implies there is an edge out of A , and that edge is a matching covering A . Suppose the theorem is true for all graphs $G = (A \cup B, E)$ with $1 \leq |A| < n$ and let $G = (A \cup B, E)$ be a graph with $|A| = n$.

Case 1. If $|\Gamma(X)| > |X|$ for all $X \subset A$ with $X \neq A$, then pick adjacent vertices $a \in A$ and $b \in B$, and consider the graph $H = G - a - b$. Then in this graph

$$|\Gamma_H(X)| \geq |\Gamma_G(X)| - 1 \geq (|X| + 1) - 1 = |X|.$$

By induction, H has a matching covering $A \setminus \{a\}$. Now add the edge $\{a, b\}$ to get a matching covering A in G .

Case 2. There exists a set $X \subset A$ with $X \neq A$ and with $|\Gamma(X)| = |X|$. In this case, consider two graphs: the graph H consisting of all edges of G between X and $\Gamma(X)$, and the graph I consisting of all edges of G between $A \setminus X$ and $B \setminus \Gamma(X)$. Since $|X| < |A|$ and $|A \setminus X| < |A|$, the induction applies to both H and I provided we can show that Hall's Condition holds in each graph. Then we find a matching covering X in H , and a matching covering $A \setminus X$ in I , and together those two matchings cover A . Definitely in H Hall's Condition holds since for $Y \subset X$, $|\Gamma_H(Y)| = |\Gamma_G(Y)|$. For $Y \subset A \setminus X$, we have by Hall's Condition in G ,

$$|\Gamma_H(X)| + |\Gamma_I(Y)| \geq |\Gamma_G(X \cup Y)| \geq |X \cup Y| = |X| + |Y|$$

and since $|\Gamma_H(X)| = |X|$ we get $|\Gamma_I(Y)| \geq |Y|$ as required. ■

Proof \triangleright (Second Proof - Via Max-Flow Min-Cut) Since $|\Gamma(A)| \geq |A|$ and $|\Gamma(B)| \geq |B|$, we have $|A| = |B|$. Add a vertex a joined to all vertices in A and a vertex b joined to all vertices in B . We claim the smallest size of an ab -separator in this graph is $|A|$. Let S be an ab -separator and let $X = A \setminus S$ and $Y = B \setminus S$. Then $|\Gamma(A \setminus S)| \geq |A| - |S \cap A|$ and $|\Gamma(B \setminus S)| \geq |B| - |S \cap B|$ by Hall's Condition. Since $\Gamma(A \setminus S) \subseteq B \cap S$ and $\Gamma(B \setminus S) \subseteq A \cap S$ – since S separates a from b – we have

$$|B \cap S| + |A \cap S| \geq |A| - |S \cap A| + |B| - |S \cap B|.$$

It follows that

$$2|B \cap S| + 2|A \cap S| = 2|S| \geq |A| + |B| = 2|A|$$

and so $|S| \geq |A|$, as required. By Menger's Theorem or the vertex form of Max-Flow Min-Cut, there exist $|A|$ internally disjoint ab -paths, and the part of these paths in G forms a perfect matching in G . ■

The maximum matching problem – the problem of finding the largest possible matching in a given bipartite graph – is a special case of the [assignment problem](#). Both can be solved algorithmically via the Max-Flow Min-Cut Theorem. For the assignment problem, the setup is still m applicants for n jobs, but now the applicants attach a weight to each job, reflecting their preference for doing that job. Using the Max-Flow Min-Cut Theorem, one can solve the assignment problem if the objective is a matching whose sum of edge weights is as high as possible. Another problem which can be solved is the [stable matching problem](#), where we want to find a matching where no improvement would be gained if two applicants switched the jobs they were assigned to under the matching.

5.8 Doubly Stochastic Matrices*

A neat corollary to Hall's Theorem is that regular bipartite graphs have perfect matchings – recall a graph is [regular](#) if all vertices have the same degree:

Corollary 8 *Let $G = (A \cup B, E)$ be a regular non-empty bipartite graph. Then G has a perfect matching.*

Proof ▷ Suppose G is k -regular. We check Hall's Condition. For $X \subseteq A$, there are $k|X|$ edges out of X . Similarly, there are $k|\Gamma(X)|$ edges out of $\Gamma(X)$. But all edges out of X go to $\Gamma(X)$, so

$$k|\Gamma(X)| \geq k|X|$$

and this is Hall's Condition. The same holds for $X \subseteq B$. ■

A natural application of this corollary is to [doubly stochastic matrices](#). A matrix A is doubly stochastic if all its entries are non-negative and all its columns and rows have the same sum, say k . A matrix is a [permutation matrix](#) if it has exactly one non-zero entry in each row and column. Examples of a doubly stochastic matrix A and a permutation matrix P are given below:

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix} \quad P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Corollary 9 *Let A be a doubly stochastic 0-1 matrix, where all columns and rows sum to k . Then A is a sum of k permutation matrices.*

Proof ▷ The matrix A can be considered as a bipartite graph: let r_1, r_2, \dots, r_n be the rows of the matrix and c_1, c_2, \dots, c_n the columns of the matrix. Form a bipartite graph G where one part is $\{r_1, r_2, \dots, r_n\}$, the other part is $\{c_1, c_2, \dots, c_n\}$, and r_i is adjacent to c_j if and only if $A_{ij} = 1$. Then since each row has k ones and each column has k ones, the bipartite graph is k -regular. In particular, this bipartite graph has a 1-factorization, say $G = F_1 \cup F_2 \cup \dots \cup F_k$. Now let P_m be the matrix whose ij th entry is 1 if and only if $\{i, j\} \in F_m$. Then P_m is a permutation matrix for $m \in \{1, 2, \dots, k\}$ and $A = P_1 + P_2 + \dots + P_k$, as required. ■

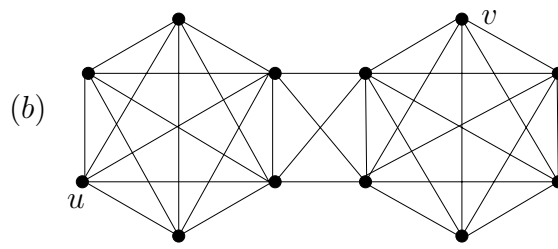
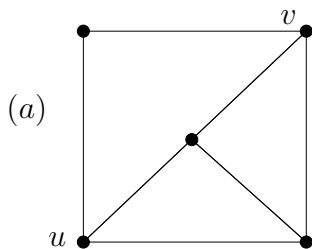
A more general fact is left as an exercise: if A is any doubly stochastic matrix, and the rows and columns sum to 1, then A is contained in the convex hull of the set of permutation matrices, namely

$$A = \lambda_1 P_1 + \dots + \lambda_N P_N$$

where P_i is a permutation matrix and $\lambda_i \geq 0$ for all i and $\sum \lambda_i = 1$.

5.9 Exercises

Question 1. Determine the minimum degree $\delta(G)$, edge connectivity $\lambda(G)$ and vertex connectivity $\kappa(G)$ for each of the following two graphs. You do not need to justify your answers. Also determine $\kappa(u, v)$ and $\lambda(u, v)$ for the given pair $u, v \in V(G)$.



Question 2. Find an example of a graph with $\delta(G) = d$, $\kappa(G) = k$ and $\lambda(G) = \ell$ where $k \leq \ell \leq d$ and k, ℓ, d are positive integers.

Question 3. Prove that if a graph has two edge-disjoint spanning trees, then that graph is 2-edge-connected.

Question 4. Show that every 3-connected graph has a cycle of even length.

Question 5. Let G be a graph with $n \geq k + 2$ vertices such that $\delta(G) \geq \frac{n+k-2}{2}$. Prove that G is k -connected.

Question 6. Let G be a graph of maximum degree at most three. Prove that $\lambda(G) = \kappa(G)$. [Hint: Menger's Theorem.]

Question 7* Let G be a 2-connected graph with m edges and n vertices. Prove that G contains at least $m - n + 2$ distinct paths connecting u and v for any pair of distinct vertices $u, v \in V(G)$, and at least $\binom{m-n+2}{2}$ distinct cycles. [Hint: prove this by induction on m .]

Question 8. Prove that every $2n \times 2n$ bipartite graph of minimum degree at least n has a perfect matching.

Question 9. Let $G(A \cup B, E)$ be a bipartite graph such that $|B| = k|A|$, and suppose $|\Gamma(X)| \geq k|X|$ for each set $X \subseteq A$. Prove that G has a spanning subgraph in which every vertex of A has degree k and every vertex of B has degree 1.

Question 10. Find a maximum st -flow in the network shown in Figure 5, starting with the given flow f consisting of unit flow in the st -path of length four at the top of the diagram. Also find a minimum cut in the network. The capacities of the arcs are denoted by numbers next to each arc.

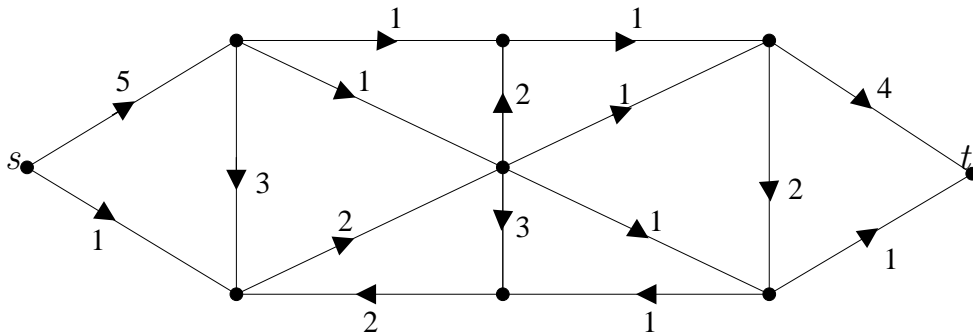


Figure 5

Question 11. Find a maximum st -flow in the network shown in Figure 6, starting with the zero flow. Also find a minimum cut in the network. The capacities of the arcs are shown as numbers next to each arc.

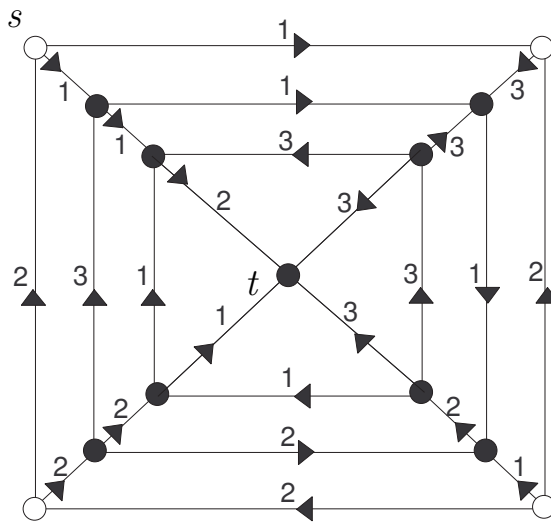


Figure 6

Question 12. Find a maximum uv -flow in the network shown in Figure 6, starting with the zero flow. Also find a minimum cut in the network. The capacities of the arcs are denoted by bars next to each arc.

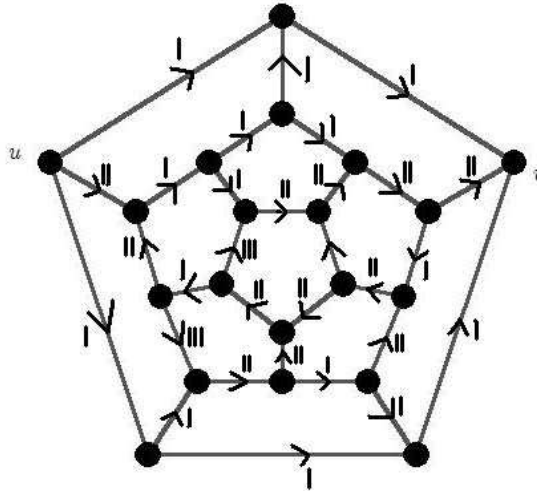


Figure 7

Question 13* Find the value of $\kappa(u, v)$ for the graph shown below. [Hint: use the proof of the vertex form of Menger's Theorem.]

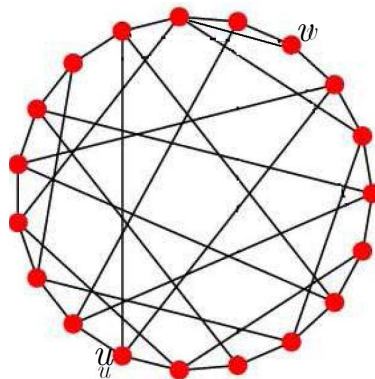


Figure 8

Question 14. In a network with a set Σ of sources and a set T of sinks, explain how you would find a maximum flow and minimum cut from Σ to T . [Hint: add a new source and a new sink.]

Question 15. Explain how to use the Max-Flow Min-Cut Theorem to find a maximum matching in a bipartite graph $G = (A \cup B, E)$. [Hint: proceed as in the proof of Hall's Theorem.]