15-484 Final Exam

Arthur O'Dwyer

May 7, 2005

1.

Is the Petersen graph planar? If it is, exhibit a plane graph isomorphic to it; otherwise, prove that it is not.

(Note that this is Exercise 9.8.1 in the book, except that they provide an algorithm that looks to me like overkill in this case.)

The Petersen graph is not planar because it is possible to condense sets of its vertices to produce a graph isomorphic to $K_{5,5}$, which is not planar itself. Condensing each pair of vertices on a "spoke" between the "inner" and "outer" cycles yields a graph isomorphic to $K_{5,5}$. In technical terms, I think we say that $K_{5,5}$ is a *minor* of the Petersen graph.

2.

Let the graph G be given. Prove that there exists an orientation D of G such that the indegree and outdegree of each vertex in D differ by at most 1.

This problem can be stated in terms of a lemma and a corollary, where the corollary is what we're asked to prove. But first, let us define this useful concept:

Definition. Let us call G an *even graph* if it is the case that $d_G(v)$ is even for every $v \in V(G)$.

Lemma. Any graph G can be constructed by taking an even graph G' and removing a set of vertex-disjoint edges from it. (I will call any such even graph G' an *even supergraph* of G.)

We prove the lemma as follows: Let G be given. It contains an even number of odd vertices. Pair them up in any way you like, and join each pair with an edge. Now each odd vertex of G has been converted to an even vertex, and each even vertex is still even; therefore, the resulting graph G' is an even graph, and removing the set of pairwise vertex-disjoint edges we just added will yield the graph G.

Corollary. Any graph G has an orientation D such that the indegree and outdegree of each vertex in D differ by at most 1.

We can orient any even graph G' simply by choosing an Euler circuit on each connected component of the graph; all connected even graphs have Euler circuits. The resulting orientation D' is such that the indegree and outdegree of each vertex are actually *equal*; this is important to what follows.

Once we know how to orient any even graph, we can orient a non-even graph by taking the orientation D' of one of its even supergraphs G', and removing the arcs corresponding to the edges in $E(G') \setminus E(G)$. Since those arcs are vertex-disjoint, the degree of each vertex decreases by at most 1. Therefore, the resulting orientation of G satisfies our criterion. *Q.E.D.*

3.

Let the plane graph G be given, and assume that it has no cut edges. Show that the faces of G can be colored with two colors so that adjacent faces have different colors, if and only if *$d_G(v)$ is even for every vertex $v \in V(G)$.*

(Note that this is Exercise 9.6.1 in the book, modulo Bondy and Murty's forgetting to deal with graphs which have cut edges, and also with graphs that are unconnected and therefore not strictly *Eulerian*.)

If G has no cut edges, then each edge of G separates two distinct faces of G. (If some edge did not separate two faces, then it would be possible to draw a continuous curve from one side of the edge to the other, not cutting any other edges, thus proving that the edge in question was a cut edge all along, since that closed curve separates the two endpoints of the given edge!)

So each edge separates two faces. Now we must prove that we can 2-color the faces of the graph iff it contains no odd vertices.

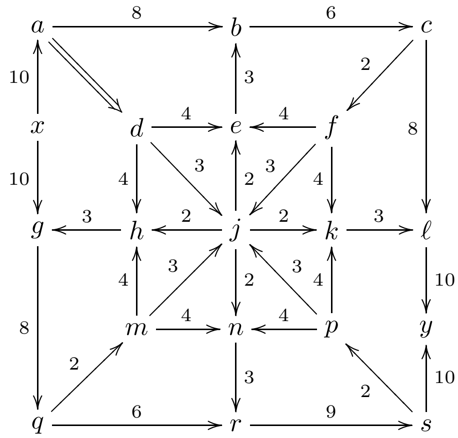
We cannot 2-color G's faces if G contains an odd vertex; this is equivalent to stating that we cannot 2-color the vertices of the dual graph G^* if G^* contains an odd face, and this is certainly true. (No bipartite graph contains an odd cycle, and each face of G^* is a cycle.)

We can 2-color G's faces if G contains no odd vertex; one algorithm for doing this is to start with a "scratch" graph $H := G$ and all G's faces colored white, and then iteratively pick one face f of H, toggle the colors of all the faces of G on the interior of f (white to black and black to white), and remove the edges of f from H. This ensures that each edge of f separates two distinct colors, and since G is composed of edge-disjoint cycles (being even), we can continue this coloring process until H is empty of edges. Then we've processed each edge of G, so each edge separates two colors on adjacent faces. And if we pick any two adjacent faces a and b, there exists an edge separating them (by definition), so a and b must have different colors. Therefore we have constructed a 2-face-coloring of the graph G.

We have now proved both directions of the equivalence; *Q.E.D.*

4.

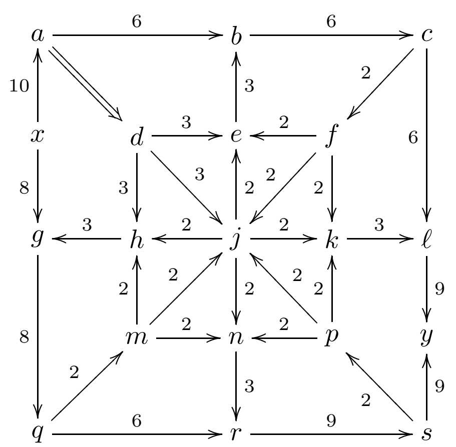
In a network, a bottleneck *is defined to be an arc such that an increase in its capacity would increase the maximum of the values of the network flows. In the depicted network, with source x and sink y, determine:*



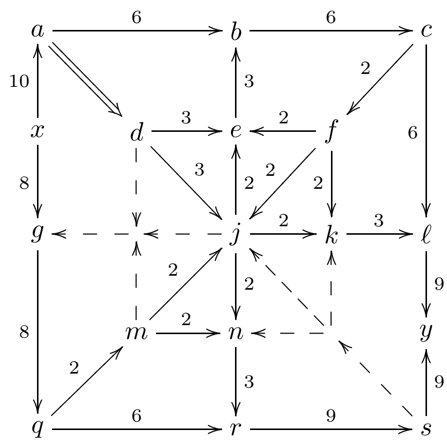
(a.) *The least capacity that the arc denoted by the heavy line must have to prevent its being a bottleneck*

The answer is 3 units. To show this is correct, we can show that the value of a maximum flow when $\text{cap}(ad) = 4$ is the same as the value of the maximum flow when $\text{cap}(ad) = 3$, and that the value of a max flow when $\text{cap}(ad) = 2$ is less. But that's just plug 'n' chug; I'm going to try a more ambitious approach.

First, we note that any edge wz cannot carry a flow greater than the total inflow at w or greater than the total outflow at z. So we modify the given network's weights accordingly:

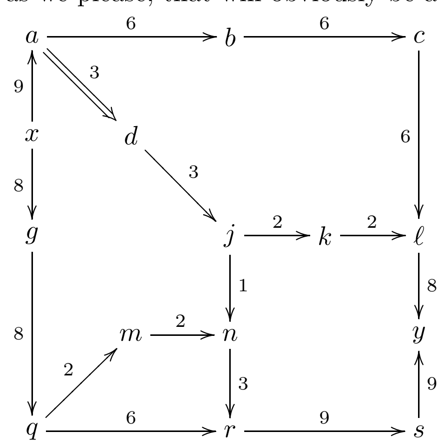


Next, we assert that since all the flow must originate at x, and everything flowing along xg must also flow along gq, we might as well set the effective capacity of hg to zero. Similar reasoning applies to sy, rs, and sp. We reapply the inflow and outflow rules: Since the effective outflow of h and inflow of p are both zero, those nodes do not participate in any max flow. So we can get rid of them:



Now, suppose our max flow carried 9 units on ℓy. Then $k\ell$ and $c\ell$ must be at max capacity. But if $c\ell$ is at max capacity, then bc carries 6 units and thus cf is empty. Then fk is empty, and therefore the inflow at k is less than 3, and $k\ell$ can't be carrying 3 units. Contradiction! Therefore, ℓy has an effective max capacity of at most 8 units.

Now for the finale. Can we produce a flow on this "reduced" network, letting cap(ad) be as big as we please, that will obviously be a maximum flow? Yes, we can.



Notice that cap$(xa) = 9$; this is because anything that goes through de must go on up eb, which doesn't gain us anything over routing it through ab to begin with. So de must be zero, and then cap(ad) must be 3 units.

This is a maximum flow on the original network as well, no matter how big we let cap(ad) get. So the minimum capacity ad can have before it becomes a bottleneck is its capacity in this maximum flow: 3 units. Increasing its capacity doesn't increase the value of the maximum flow after this point.

(b.) *A maximum flow and a minimum cut in the network when this arc is given the capacity determined in part (a).*

For the maximum flow, see the last diagram in part (a). One minimum cut (which must have capacity 17) in the given network with cap$(ad) = 3$ is the pair of sets $\{x, a, b, d, e, g, h\}$ and $\{c, f, j, k, \ell, m, n, p, q, r, s, y\}$. Notice that a and d are on the same side of the cut, proving (in a *third* way!) that ad is not a bottleneck when the max flow is 17 units; at least one of bc, dj, gq would have to be increased in capacity before the max flow could increase.