Math 3322: Graph Theory[1] Mikhail Lavrov

Lecture 15: Connectivity

October 7, 2021 *Kennesaw State University*

1 New definitions

So far, we've talked about bridges (removing one edge to disconnect a graph) and cut vertices (removing one vertex to disconnect a graph). Now, we will generalize this.

1.1 Vertex connectivity

A **vertex cut** in a graph G is a subset $U \subseteq V(G)$ such that $G - U$ (that's the subgraph of G with the vertices in U, and all their incident edges, deleted) is no longer connected. If G is not connected, then the empty set is a vertex cut; we will not think about details like "increasing the number of connected components" here.

We would like to measure the connectivity of a graph by the number of vertices in the smallest vertex cut. There's a bit of a sticking point here: complete graphs don't have any vertex cuts! There is no way to turn K_n into a disconnected graph by deleting vertices.

So we have a bit of a funny definition. The **(vertex) connectivity** of a graph G, denoted $\kappa(G)$ is defined to be:

- The size of the smallest vertex cut of G, if one exists: if G is not a complete graph.
- $\kappa(K_n)$ is "artificially" set to $n-1$. We will see later why $n-1$ is the "right" value to choose.

In particular, $\kappa(G) = 0$ if G is not connected.

We say that a graph is k**-connected** if $\kappa(G)$ is *at least* k. This definition exists because we often want to say "Such-and-such result applies if our graph is connected *enough* for it to work".

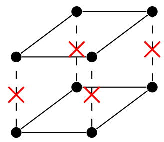
For example, all our properties of nonseparable graphs on $n \geq 3$ vertices are also properties of 2-connected graphs. The two definitions are almost the same, except that they treat K_2 (the graph with two vertices and one edge) differently. We'd like K_2 to be nonseparable, because it lets us decompose graphs into blocks. However, it does not share most properties of 2-connected graphs, and it makes more sense to say $\kappa(K_2) = 1$.

1.2 Edge connectivity

An **edge cut** in a graph G is a subset $X \subseteq E(G)$ such that $G - X$ (that's the subgraph of G with the edges in X deleted, and with all the same vertices) is no longer connected. If G is not connected, then the empty set is an edge cut; we will not think about details like "increasing the number of connected components" here.

[1] This document comes from the Math 3322 course webpage: http://facultyweb.kennesaw.edu/mlavrov/courses/3322-fall-2021.php

If you delete some edges to make G disconnected, there is going to be some set of vertices S such that you've deleted all the edges between S and $V(G) - S$ (disconnecting S from the rest of the graph). For example, one way to disconnect the cube graph is to delete the edges between the four "top" vertices and the four "bottom" vertices:



Conversely, deleting all the edges between S and $V(G) - S$ is enough to disconnect G (provided neither S nor $V(G) - S$ is empty); we don't need to do anything else. In fact, some sources reserve the word "edge cut" just for sets of edges of this form.

The **edge connectivity** $\kappa'(G)$ is the smallest number of edges in an edge cut of G. Here, there's no funny business: $\kappa'(K_n) = n - 1$ without any need to make it a special case. (We'll see why later.) We still say $\kappa'(G) = 0$ if G is not connected.

Finally, just as in the previous section, we say that a graph G is **k-edge-connected** if $\kappa'(G)$ is at least k.

1.3 A note on the Greek letters

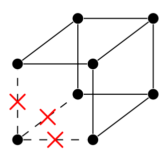
In both of these, κ is the Greek letter "kappa".

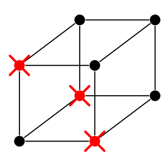
We use matching notation—$\kappa(G)$ for vertex connectivity and $\kappa'(G)$ for edge connectivity—as part of a general rule: we add a prime (the $'$ next to the kappa) to indicate "the edge version of this parameter". This will come up again in later units, where we have other paired parameters like this.

Your textbook uses $\lambda(G)$ for $\kappa'(G)$; this is less common and worse notation.

2 Some starting bounds on connectivity

The four-edge cut in the cube graph is not the best edge cut. We can disconnect the cube graph by deleting only three edges, if they're all edges out of the same vertex:





Similarly, by deleting the three neighbors of a vertex, we see that the cube graph has a vertex cut of size 3.

This strategy works in general. We can always disconnect a graph by deleting all the edges out of a vertex (and it makes sense to choose a minimum degree vertex). Graphs that are not complete can be disconnected by deleting all vertices adjacent to a minimum degree vertex; this gives us the bounds

$$\kappa(G) \le \delta(G) \qquad \kappa'(G) \le \delta(G)$$

that, separately, work out for complete graphs as well.

There is a stronger result:

Theorem 2.1. *For any graph G, $\kappa(G) \le \kappa'(G) \le \delta(G)$.*

Proof. The only inequality here that we haven't shown yet is $\kappa(G) \le \kappa'(G)$. Intuitively, this should hold because deleting vertices is more efficient at disconnecting a graph than deleting edge: deleting a vertex destroys all edges out of that vertex.

Let $X \subseteq E(G)$ be an edge cut of size $\kappa'(G)$: because X is as small as possible, we may assume that it consists of all edges between a set $S \subseteq V(G)$ and $V(G) - S$. We will use X to find a vertex cut $U \subseteq V(G)$ with $|U| \le |X|$.

Here's a first attempt at a proof: for each edge of X, add one of its endpoints to U. This guarantees that $|U| \le |X|$ (maybe $|U| < |X|$ if we add the same vertex multiple times) and it *almost* guarantees that deleting all vertices in U separates S from $V(G) - S$. There is one problem: what if we end up deleting all of S, or all of $V(G) - S$?

Here's a second attempt at a proof: pick a vertex $s \in S$ and a vertex $t \in V(G) - S$, and do the above, but more carefully. If s or t is one of the endpoints of an edge in X, we add the other endpoint to U. Now $G - U$ separates s from t in *almost* all cases... but what do we do if our edge cut contains edge st?

Here's a final attempt at a proof: if not *every* edge between S and $V(G) - S$ is present, then pick vertices s and t such that st is not an edge, and do the above.

What if we are, in fact, deleting all edges between S and $V(G) - S$? Then if $|V(G)| = n$ and $|S| = k$, our edge cut contains $k(n-k)$ edges, which is always at least $n-1$. But we already know $\kappa(G) \le \delta(G) \le n-1$, so we can still conclude that $\kappa(G) \le \kappa'(G)$. □

3 Local connectivity and disjoint paths

3.1 Vertex $s - t$ cuts

Now let's suppose we have a graph G and we pick two vertices s and t. An $s - t$ **cut** in G is a vertex cut U that separates s from t: $G - U$ still contains both s and t, but they're in different components. When it is clear which graph G we are talking about, we write

$$\kappa(s,t) = \min\{|U| : U \text{ is an } s - t \text{ cut}\}.$$

This is a more "local" measure of connectivity: instead of trying to separate the graph anywhere, we specifically want to separate s from t.

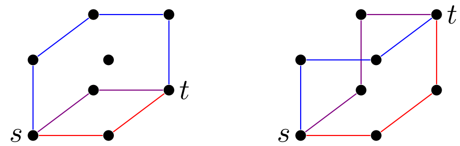
If s and t are adjacent, then no $s-t$ cut can exist: you can't destroy the edge st by deleting vertices that aren't s or t. In that case, we say that $\kappa(s,t) = \infty$.

Sometimes we specifically care about finding $\kappa(s,t)$, but other times, we use it as an indirect path toward finding $\kappa(G)$. Here is an example:

Claim 3.1. *If G is the cube graph, and s, t are any two non-adjacent vertices of G, then $\kappa(s,t) \geq 3$. As a result, $\kappa(G) = 3$.*

Proof. It's enough to consider two cases: taking $s = (0,0,0)$ and $t = (0,1,1)$, or taking $s = (0,0,0)$ and $t = (1,1,1)$. That's because the cube graph has many automorphisms. In particular, it has an automorphism taking any pair of non-adjacent vertices to one of these two pairs.

In each of the two cases, we can draw three **internally disjoint** $s-t$ paths: three $s-t$ paths that don't share any vertices other than their endpoints. Here's one way to do this:



If there are three such paths, then we can't possibly separate s from t by deleting fewer than three vertices. Each vertex we delete can destroy at most one of the paths.

This means that the cube graph can't have any vertex cut of size 2. Such a vertex cut would separate *some* pair of vertices s and t, and then we'd get $\kappa(s,t) = 2$ as well, which we just proved cannot happen. As a result, $\kappa(G) \geq 3$.

Since $\kappa(G) \leq \delta(G) = 3$, we conclude that $\kappa(G)$ must be equal to 3. □

The ideas here generalize to other graphs:

- If s, t are two non-adjacent vertices in a graph G, we can prove that $\kappa(s,t) \geq k$ by finding k internally disjoint $s-t$ paths.

  In the next lecture, we will prove Menger's theorem, which promises that this strategy can always find the exact value of $\kappa(s,t)$. That is, if s and t are two non-adjacent vertices in a graph G, there are always $\kappa(s,t)$ internally disjoint $s-t$ paths in G.

- If we prove that $\kappa(s,t) \geq k$ for *every* pair of vertices in G, then $\kappa(G) \geq k$ as well.

3.2 Edge $s-t$ cuts

Similarly to the above, an $s-t$ **edge cut** in G is an edge cut X such that s and t are in different components of $G - X$. We write

$$\kappa'(s,t) = \min\{|U| : U \text{ is an } s-t \text{ edge cut}\}.$$

Basically everything we said for $\kappa(s,t)$ generalizes to $\kappa'(s,t)$, with suitable modifications. For example, to prove lower bounds on $\kappa'(s,t)$, it is enough for our collection of $s-t$ paths to be edge-disjoint: it's okay if they share vertices, as long as they share no edges.

For example, if s, t are any two vertices of K_n, then $\kappa'(s,t) = n-1$. To prove the lower bound, consider the length-1 path (s,t) and the $n-2$ length-2 paths (s,v,t) where v is any other vertex. To prove the upper bound, delete all edges out of s.