Graph Theory: Week 3

Connectivity

John Quinn

4th September, 2007

Refs for this week

- We are again loosely following Bondy and Murty, “Graph Theory with Applications”, Chapters 2 & 3.
- For revision, look at sections 2.3 (cut vertices), 2.4 (Cayley’s formula), and sections 3.1–3.3 on connectivity.
- Local copies will be kept on `blackboard.mak.ac.ug`.

Recap: trees and spanning trees

- What are the conditions for a graph to be a tree?
- What is the relationship between the number of edges ε and number of vertices ν in a tree?
- How can you tell quickly if a simple, connected graph contains any cycles?
- Which edges in a tree are *cut edges*?
- What is a spanning tree of a graph?

Finding the number of possible spanning trees

- Denote the number of spanning trees of a graph G by $\tau(G)$.
- Cayley's formula can be used to find $\tau(G)$ recursively.

Cut vertices

- A vertex v is a cut vertex of a graph if the set of edges E can be partitioned into two sets $\{E_1, E_2\}$ which only have v in common.
- Intuitively, removing a cut vertex disconnects a graph.

Cut vertices (cont.)

Where are the cut vertices in the graph on the board?

Blocks

- A block is a connected graph which has no cut vertices.
- A block of a graph is a maximal subgraph with no cut vertices.

Block-finding algorithm

Exercise: design an algorithm for finding blocks in a graph.

Some confusing terminology

- An *edge cut* is a subset of the edges of a graph, such that removing them disconnects the graph.
- A *vertex cut* is a subset of the vertices of a graph, such that removing them disconnects the graph.
- Don't get confused with cut edges and cut vertices!
- Cut edges are sometimes referred to as *bridges*.

Connectivity

- The connectivity $\kappa(G)$ is the smallest k for which there is a k-vertex cut in G.
- In a communications network, for example, this would tell us how resilient the network is: k nodes have to be removed before there is any disconnection.

Edge connectivity

- The edge connectivity $\kappa'(G)$ is a similar concept for the smallest number of edges which can be removed to disconnect a graph.
- We say a graph is *k*-edge connected if it takes *k* edges to be removed from the graph to disconnect it.
- So this is another measure of how strongly connected the graph is.

Designing good networks

- Last week we talked about designing networks (e.g. of roads or computers) which most efficiently connect all points.
- What if we also want to make these networks reliable?
- We might want to add extra links to improve reliablility, but where is the best place to put them?

Extension of Kruskal's algorithm

- Kruskal's algorithm finds a spanning graph which is 1-edge connected.
- This can be extended to find k-edge connected subgraphs if we assume that each edge has unit weight.

Exercise: designing a resilient network

A network of 6 computers is to be constructed, such that any 3 computers can fail but all the others remain connected. Find the graph of an efficient solution.