MAU22C00 Lecture 20

John Stalker

Trinity College Dublin

Announcements

- I posted Assignment 5 to the module webpage on Monday. I *think* I posted it to Blackboard at the same time.
- I found out yesterday afternoon it wasn't there and (re?)posted it.
- I'm giving people an extra three days, so it's due on Monday now.
- I'm not the only one whose files Blackboard eats. We've had a few assignments eaten as well. Screenshot your upload confirmation!
- If you don't see materials on Blackboard which should be there then check the website, `https://www.maths.tcd.ie/~stalker/22C00/`, (and vice versa) and let me know.
- The lecture on Tuesday (7 November) will be pre-recorded. Normal service will resume, hopefully, on Thursday.

Degrees

With each vertex in a graph we can associate an in-degree, the number of edges for which it's the final vertex, and an out-degree, the number for which it's the initial vertex.

For undirected graphs these numbers are the same, and just called the degree.

Isomorphisms take always take a vertex to one of the same (in/out) degree.

Each edge contributes one to the in-degree of some vertex and one to the out-degree of some vertex, so the sum of the in-degrees is the number of edges and the sum of the out-degrees is the number of edges.

These sums are the same.

For an undirected graph with no self-loops the sum of the degrees is even. The number of number of vertices of odd degree is even.

Undirected graphs with no self-loops are called regular if all vertices have the same degree.

Complete graphs are regular. So is the Wumpus graph, but not $K_{4,3}$.

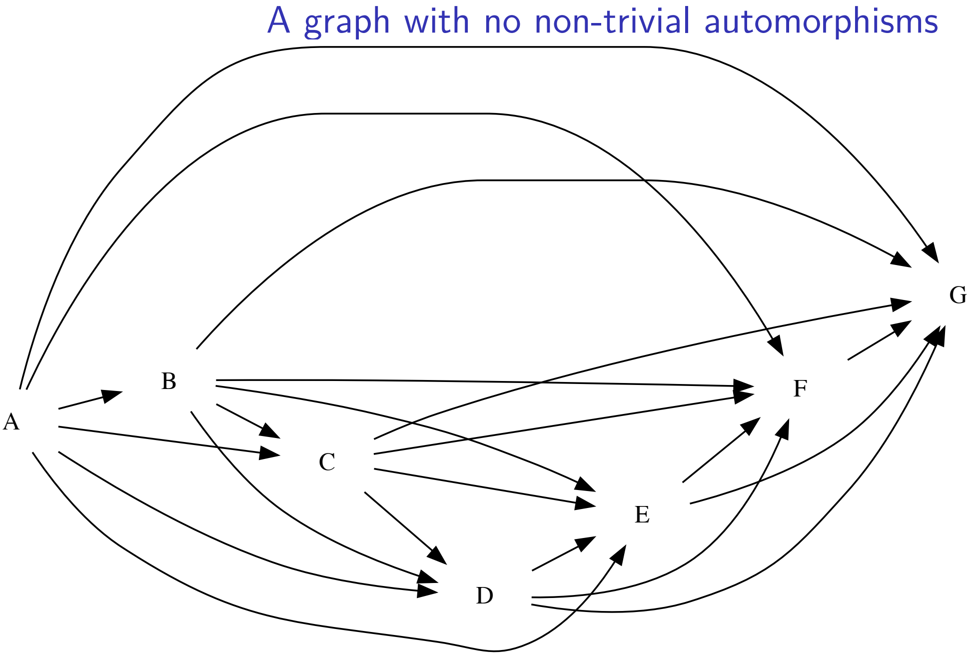

Figure 1: A directed graph

Definitions

- A walk is a list of edges, the initial vertex of each being the final vertex of the preceding one.
- A trail is a walk with no repeated edges.
- A path is a walk with no repeated vertices.
- A path is closed if the initial vertex (of the initial edge) is the same as the final vertex (of the final edge).
- Closed trails are called circuits.
- Every path is a trail.
- Not every trail is a path. For example, circuits are never paths.
- A graph is connected if for every pair of points there is a walk from the first to the second.

A disconnected graph

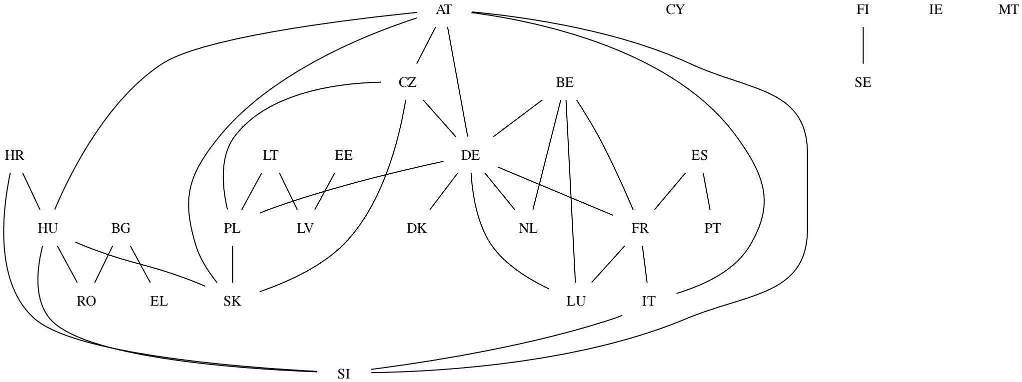

Figure 2: A disconnected graph

A path

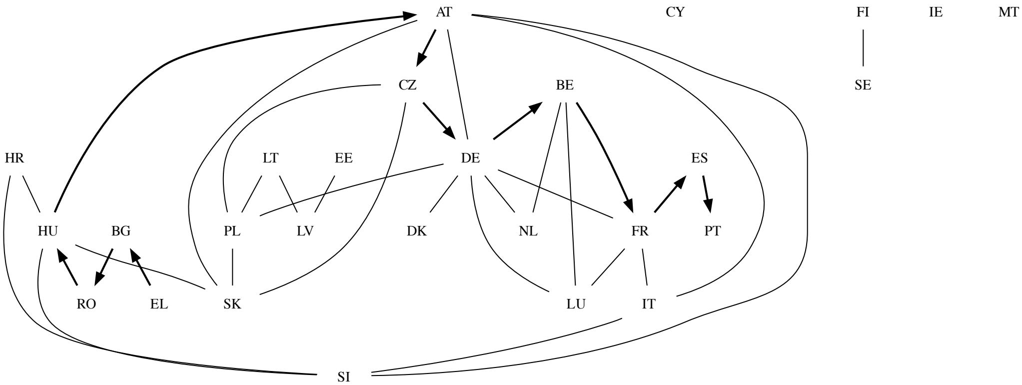

Figure 3: A path in the preceding graph

A circuit

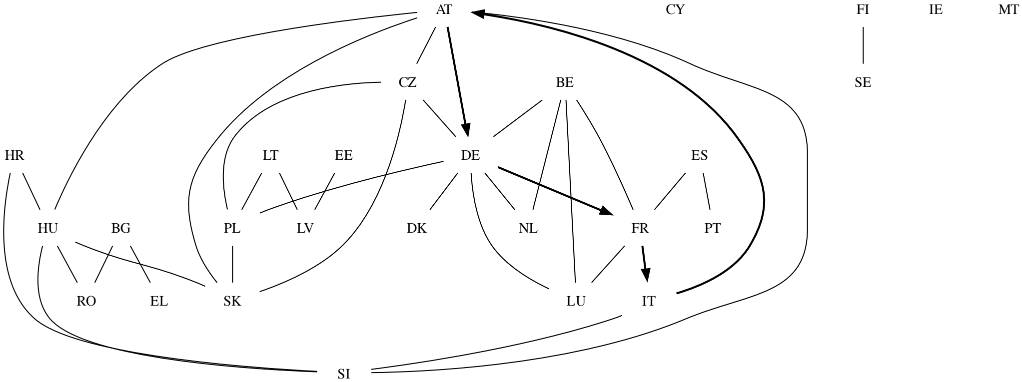

Figure 4: A circuit in the EU border graph

Touring the EU

Is it possible to visit all EU countries exactly once, without leaving the EU, leaving the ground, or taking the Øresund/Öresund Bridge?

No. If you don't start in Ireland you can never get here, and if you do start here then you can never leave.

Can you visit all the countries in the large connected component?

No. Denmark, Estonia, Greece and Portugal each border only one EU country. None of them can appear in the middle of your tour, only at the beginning or end.

A path in a graph is called Hamiltonian if each vertex occurs exactly once. The EU border graph has no Hamiltonian path.

A necessary, but not sufficient, condition for the existence of a Hamiltonian path, is connectedness.

There is no useful necessary and sufficient condition.

A Hamiltonian path

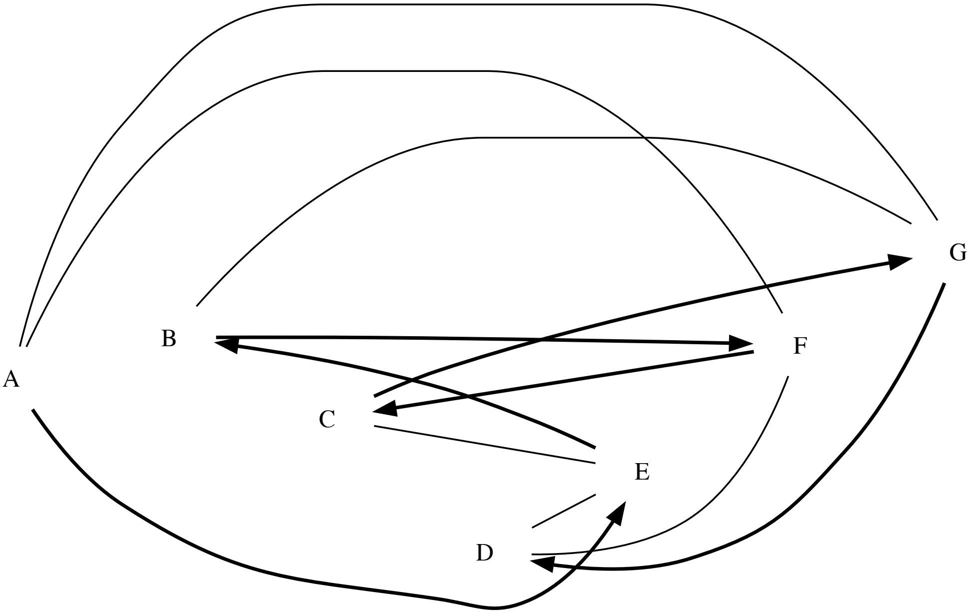

Figure 5: A Hamiltonian path

Eulerian trails

What if you want to cross every EU *border* exactly once?

A trail which traverses each edge (really one from each pair) in an undirected graph without self-loops exactly once is called an Eulerian trail.

This is also impossible. Either you start in Finland or Sweden and can cross only that border or you start elsewhere and can never cross that border.

Necessary condition for an Eulerian trail: the graph is connected-ish, i.e. at most one connected component has more than one vertex.

Can we find an Eulerian trail for the main component of the EU border graph?

No. Denmark, Estonia, Greece and Portugal are again a problem. Each has a single edge and that edge would have to be either the first or last we traverse.

Necessary condition for an Eulerian trail: there are at most two vertices of degree one.

We can strengthen this to: there are at most two vertices of odd degree. Why?

Eulerian trails, continued

Consider the directed graph consisting of those edges in the Eulerian trail.

Vertices appearing in the middle of the trail have an incoming edges, contributing one to the in-degree and an outgoing edge, contributing one to the out-degree.

The first vertex has only an outgoing edge, contributing one to its out-degree. The last vertex has only an incoming edge, contributing one to its in-degree.

These vertices could be the same, if the trail is a circuit.

If the trail is a circuit then the in-degrees and out-degrees are equal everywhere, and their sum for each vertex is even.

Otherwise the in-degree and out-degree are equal except at the initial and final vertices, where they differ by one, so the sum is odd at those vertices and even elsewhere.

These are the in-degrees and out-degrees for the directed graph. The degree of a vertex in the original, undirected, graph is their sum.

Eulerian trails, continued

If an undirected graph, without self loops, has an Eulerian trail then the number of vertices of odd degree is zero or two.

If it's zero then the trail, and all Eulerian trails, is a circuit.

If it's two then the trail is not a circuit, and neither is any other Eulerian trail.

We have a pair of necessary conditions for the existence of an Eulerian trail:

- At most one connected component has more than a single vertex
- At most two vertices have odd order

What about a single vertex of odd order?

Can't happen! The number of vertices of odd degree is even.

Finding an Eulerian circuit

Suppose at most one connected component has more than a single vertex and all vertices have even order. Is there an Eulerian circuit?

In other words, are our necessary conditions also sufficient?

We can ignore connected components with only one vertex since they have no edges.

So we can assume our graph is connected and non-empty.

First idea: Start anywhere and wander around (nearly) aimlessly.

Don't repeat any edges, but otherwise choose edges at random until none are available.