Math 3322: Graph Theory[1] Mikhail Lavrov

Lecture 19: Directed graphs

October 21, 2021 *Kennesaw State University*

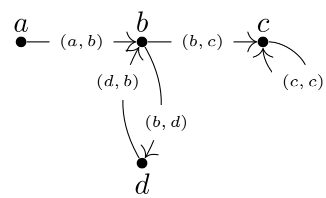
1 Introduction to directed graphs

A **directed graph** or **digraph** is not a graph. It is a generalization of a graph, meant to model asymmetric relationships. (Friendships on Facebook give us a graph; follows on Twitter give us a digraph.)

Formally, a directed graph has a set of vertices V and a set of **directed edges** or **arcs** E, which is a subset of $V \times V$. An arc is an ordered pair (v, w), which we might sometimes write "$v \to w$", and represents a relationship with a direction: from v to w.

We will allow loops: arcs of the form (v, v), which go from v back to v. We also allow the arcs (v, w) and (w, v) to exist in the same digraph.

In diagrams, we represent arcs by arrows:



1.1 Indegrees and outdegrees

In a directed graph, a vertex doesn't just have a degree. Instead, we count:

- The **indegree** $\deg^-(v)$ is the number of arcs oriented toward v. In the graph above, we have
$$\deg^-(a) = 0 \qquad \deg^-(b) = 2 \qquad \deg^-(c) = 2 \qquad \deg^-(d) = 1$$
Your textbook writes $\mathrm{id}(v)$ for the indegree, but this is unusual.
- The **outdegree** $\deg^+(v)$ is the number of arcs oriented away from v. In the graph above, we have
$$\deg^+(a) = 1 \qquad \deg^+(b) = 2 \qquad \deg^+(c) = 1 \qquad \deg^+(d) = 1$$
Your textbook writes $\mathrm{od}(v)$ for the outdegree, but this is unusual.

A version of the degree sum formula[2] holds for directed graphs: for any digraph D,
$$\sum_{v \in V(D)} \deg^-(v) = \sum_{v \in V(D)} \deg^+(v).$$

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

[2] If the degree sum formula is called the "handshake lemma", then this formula should be called the "handshake dilemma". The dilemma is this: how can you have a directed handshake?

1.2 Walks, paths, and cycles

We can define a $u - v$ walk in a directed graph D in almost the same way as we did for graphs. It's a sequence of vertices $(v_0, v_1, \dots, v_\ell)$ where $v_0 = u$, $v_\ell = v$, and for every i, (v_i, v_{i+1}) is an edge; but now, we ask for it to be a directed edge oriented from v_i to v_{i+1}. In other words, a walk in a directed graph must follow the arrows.

Paths and cycles are going to be very similar. However, we allow the cycle (v, v) of length 1 if there is a loop at v, and the cycle (v, w, v) of length 2 if there are edges in both directions between v and w. In the undirected case, we did not like (v, w, v) because it used the same edge twice; here, the cycle has to use two different edges.

We can still ask about the distance between u and v: the length of a shortest $u - v$ path. Your textbook writes this as $\vec{d}(u, v)$ rather than $d(u, v)$ to emphasize that this is asymmetric: the distance from v to u might be completely different (or might not even be defined).

1.3 Other properties

Subgraphs and isomorphisms of directed graphs are defined in the same way as for graphs.

Directed graphs can also have Hamiltonian cycles or Eulerian tours; we will look into some conditions for these things happening.

Menger's theorem holds, with $\kappa(s, t)$ counting the number of vertices we must delete to destroy all directed $s - t$ paths. In fact, if we investigate our proof carefully, it can be adapted to the directed case with almost no changes.

2 Weak and strong connectedness

The first problem we encounter is with defining a "connected" digraph. What are we supposed to do in situations where a path exists from v to w, but not from w back to v?

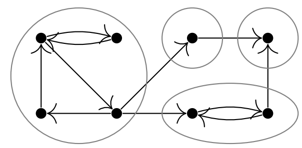
There are several solutions:

1. A **weakly connected** digraph is one that becomes a connected graph if we ignore the directions of the edges. This is sometimes a useful notion, but if that's all we do, why even bother having a digraph in the first place?
2. A **rooted** digraph has a "root vertex" u such that for all v, there is a $u - v$ (directed) path. This is not going to be very useful for us this semester, but comes up occasionally.
3. A **strongly connected** digraph has a directed $v - w$ path for any v and w (in either direction). This is asking for a lot, but it's very useful.

We defined connected components of graphs using equivalence relations. The relationship "there is a $v - w$ path" is no longer an equivalence relation: it is not symmetric. This complicates things.

However, the relationship "there is both a $v - w$ path and a $w - v$ path" *is* an equivalence relation. It lets us break up a directed graph into **strongly connected components**. Within each strongly connected component, every vertex can reach every other vertex. However, if v and w are in different connected components, either there is no $v - w$ path, or no $w - v$ path, or both.

Here is an example of a directed graph being separated into strongly connected components:



Note that this picture is more complicated than splitting up a graph into connected components, since some arcs go between different strongly connected components. In particular, the strongly connected components don't tell the whole story of which vertices have a path to which other vertices...

3 Eulerian digraphs

An Eulerian tour of a digraph D is a closed walk that uses every arc of D exactly once. We've already thought about Eulerian tours in the undirected case, so we just need to do the same things here and see what changes.

There are two necessary conditions analogous to the conditions we had for undirected graphs.

- A **degree condition**: if a digraph D has an Eulerian tour, then $\deg^-(v) = \deg^+(v)$ for every vertex v.

  This is necessary because the Eulerian tour enters v $\deg^+(v)$ times, and exits v $\deg^-(v)$ times.

- A **connectedness condition**: if a digraph D has an Eulerian tour then all non-isolated vertices[3] are in the same strongly connected component.

  If we have an Eulerian tour, then to get from a non-isolated vertex v to a non-isolated vertex w, just follow the Eulerian tour starting from v until it visits w.

To prove that these conditions are necessary in the undirected case, we did three things. First, we figured out how to find cycles in the graph. Then, we figured out how to find a cycle decomposition. Then, we put it together to get an Eulerian tour.

Lemma 3.1. *Let D be a digraph in which every vertex v has $\deg^+(v) \geq 1$. Then D contains a directed cycle.*

Proof. Let $(v_1, v_2, \ldots, v_k)$ be a path in D that's as long as possible (while still remaining a path). Because $\deg^+(v_k) \geq 1$, there is an arc out of vertex v_k; because the path cannot be extended, every such arc goes back to another vertex on the path. (That vertex could be v_k itself! This is fine.

Let (v_k, v_i) be one such arc. Then $(v_i, v_{i+1}, \ldots, v_k, v_i)$ is a cycle. □

[3] For now, let's say that a vertex v is isolated if $\deg^+(v) = \deg^-(v) = 0$. If there is a loop edge (v, v), then v is not isolated, even if there are no other arcs in or out of v.

Lemma 3.2. *Let D be a digraph in which every vertex v has $\deg^-(v) = \deg^+(v)$. Then D has a cycle decomposition.*

Proof. We induct on the number of arcs in D. If there are no arcs, then we don't have to do anything to get a cycle decomposition.

Suppose D has m arcs and the lemma holds for all digraphs D with fewer than m arcs. Let D' be the subgraph of D (subdigraph? nobody says that) without any of its isolated vertices. Then every vertex v of D' has $\deg^-(v) = \deg^+(v)$, but also $\deg^+(v), \deg^-(v)$ can't both be 0; therefore every vertex v of D' has $\deg^+(v) \geq 1$.

By Lemma 3.1, D' has a cycle. Deleting all arcs of the cycle from D' leaves a graph with fewer than m arcs in which the degree conditions still holds. That graph has a cycle decomposition, and adding on the cycle we found gives a cycle decomposition of D' — and of D. □

Theorem 3.3. *Let D be a digraph in which every vertex v has $\deg^-(v) = \deg^+(v)$, and which is at least weakly connected (except for isolated vertices). Then D has an Eulerian tour.*

Proof. This proof works in the same way as our proof for undirected graphs. The condition we use is "weakly connected" because if there are no unused cycles we can add to the partial tour, then the vertices visited by the partial tour and the vertices visited by the unused cycles cannot have any arcs between them, in either direction. This would contradict weak connectivity. □

The necessary condition we found was that the non-isolated vertices of D must form a strongly connected component, but the theorem's sufficient conditions only ask for a weakly connected component. This is not a mistake. Think about how to resolve this.

4 Directed acyclic graphs

Direced acyclic graphs, sometimes abbreviated **dag**s,[4] are exactly what they sound like: directed graphs that contain no cycles. In the directed case, there cannot even be closed walks (of positive length), because any such closed walk contains a cycle.

These are a sort of polar opposite of strongly connected digraphs. In a directed acyclic graph, there cannot be a $v-w$ path and a $w-v$ path at the same time, because that would form a closed walk. So there cannot be a strongly connected component containing more than one vertex. Instead, each vertex is its own strongly connected component.

If you have any directed graph that represents prerequisites (for classes you take, or for software libraries you install, or for steps in a recipe), it had better be acyclic. If there were a cycle of prerequisites between classes, then you wouldn't be able to take any of them, because each one of them would have another one you have to take first! This is one common way directed acyclic graphs show up in applications.

[4] It would probably make more sense to call them "acyclic directed graphs", but "adg" is harder to pronounce.

If you have a recipe, the steps will probably be listed in an order so that the prerequites for each step all come before it. (Sometimes there is flexibility, but you're going to have to follow the recipe in some order, anyway.) This is a general feature of directed acyclic graphs:

Theorem 4.1. *If D is a directed acyclic graph, then it has a* ***topological ordering****: an ordering $v_1, v_2, \ldots, v_n$ of the vertices of D such that every arc (v_i, v_j) has $i < j$.*

(In a diagram where vertices $v_1, v_2, \ldots, v_n$ are placed from left to right in that order, every arc will also point from left to right.)

Proof. This can be shown by induction on n, the number of vertices. If $n = 1$, then there is nothing to check.

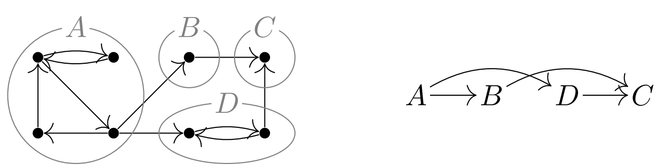
Assume that the theorem holds for $n-1$ vertices, and let D be an n-vertex directed acyclic graph. By Lemma 3.1, if $\deg^+(v) \geq 1$ for all vertices v, then D contains a cycle; since D does not have any cycles, there must be some vertex v with $\deg^+(v) = 0$.

Take a topological ordering $v_1, \ldots, v_{n-1}$ of $D - v$, and then add vertex v last, as v_n. There are no arcs out of v, and all arcs into v are (v_i, v_n) with $i < n$, so they obey this ordering. □

Here is one final application of directed acyclic graphs. Suppose we want to have a summary that says, for any two vertices v, w, if there is a $v - -w$ path. How should we give it?

Part of the story is, of course, the strongly connected components. If v, w are in the same strongly connected component, then there are paths betwen them in either direction. But what if they are in different strongly connected components?

For this, we can take every strongly connected component and condense it into a single vertex. In this new directed graph, we'll have an edge $A \to B$ if there are any edges from the strongly connected component A to the strongly connected component B:



This new graph is a directed acyclic graph (if there were a cycle in it, then those strongly connected components could be combined into one). To see if there is a $v - -w$ path in the original digraph, look up the strongly connected components of v and w, and see if there is a path between them in the directed acyclic graph.

For instance, there is a path (A, D, C) in the example above; therefore any vertex in strongly connected component A has a path to the vertex in strongly connected component C.