Chapter 2 : Paths and Cycles

Omar Saadi
omar.saadi@um6p.ma

UM6P - College of Computing

1st year of the engineering degree of UM6P-CS
Academic year 2024-2025

Outline

More details on this chapter can be found in Sect 1.2 and Sect 7.2 of D. B. West's book.

Outline

Recall

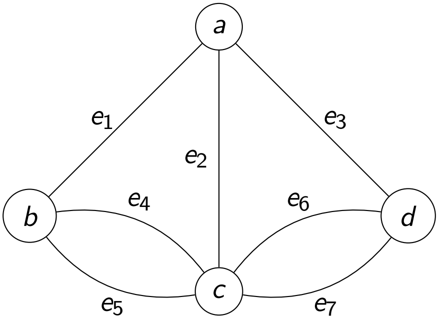
A **path** is a simple graph whose vertices can be ordered so that two vertices are adjacent if and only if they are consecutive in the list.
A **cycle** is a graph with an equal number of vertices and edges whose vertices can be placed around a circle so that two vertices are adjacent if and only if they appear consecutively along the circle.

Note that a path in a graph G is a subgraph of G that is a path (similarly for cycles).

Definition

A **walk** is a list $v_0, e_1, v_1, \ldots, e_k, v_k$ of vertices and edges such that, for $1 < i < k$, the edge e_i has endpoints v_{i-1} and v_i.
A **trail** is a walk with no repeated edge.
A u, v-**walk** or u, v-**trail** has first vertex u and last vertex v; these are its endpoints. A u, v-**path** is a path whose vertices of degree 1 (its endpoints) are u and v; the others are **internal vertices**.
The length of a walk, trail, path, or cycle is its number of edges. A walk or trail is **closed** if its endpoints are the same.

Example : In the Königsberg graph :



the list $b, e_4, c, e_2, a, e_1, b, e_5, c, e_4, b$ is a closed walk of length 5 ; it repeats edge e_4 and hence is not a trail. If we delete the last edge and vertex e_4, b, we obtain trail.

The list $a, e_1, b, e_4, c, e_7, d$ is a a, d-path, and $d, e_6, c, e_2, a, e_3, d$ is a cycle.

Remark : In a simple graph, a walk (or trail) is completely specified by its ordered list of vertices.

Lemma

Every u, v-walk contains a u, v-path.

Proof : by induction on the length l of a u, v-walk.

Definition

A graph G is **connected** if for all $u, v \in V(G)$, there exists u, v-path in G (otherwise, G is **disconnected**).
If G has a u, v-path, then u is connected to v in G. The connection relation on $V(G)$ consists of the ordered pairs (u, v) such that u is connected to v.

Remark

We recall the distinction between "adjacent" and "connected" vertices :

- Vertices u and v are adjacent if and only if $uv \in E(G)$.
- Vertices u and v are connected if and only if G contains a u, v-path.

Remark

Using the previous lemma, a way to prove that a graph is connected is by fixing a particular vertex v^* and showing that from each vertex $u \in V(G)$ there is a walk to that particular vertex v^*.

Proposition

The connection relation is an equivalence relation.

Proof :

Components of a graph

Definition

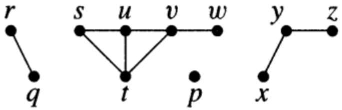
A **maximal connected subgraph** of G is a subgraph that is connected and that is not contained in any other connected subgraph of G.
The **components** of a graph G are its maximal connected subgraphs. A component (or graph) is **trivial** if it has no edges ; otherwise it is nontrivial.
An **isolated vertex** is a vertex of degree 0.

Remark

The equivalence classes of the connection relation on $V(G)$ are the vertex sets of the components of G.
An isolated vertex forms a trivial component, consisting of one vertex and no edge.

Example

In the following graph :



we have four components. The vertex sets of these components are $\{p\}, \{q, r\}, \{s, t, u, v, w\}$, and $\{x, y, z\}$; these are the equivalence classes of the connection relation.

Remark

Components are pairwise disjoint ; no two share a vertex.
Adding an edge with endpoints in distinct components combines them into one component. Thus, adding an edge decreases the number of components by 0 or 1,
Deleting an edge increases the number of components by 0 or 1.

Proposition

Every graph with n vertices and k edges has at least $n - k$ components.

Proof :

Remark

Deleting a vertex (and also all edges incident to it, to still have a graph) can increase the number of components by many. Consider the example of the $K_{1,m}$, where deleting the vertex corresponding to the partite set with one vertex increases the number of components from 1 to m.

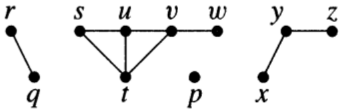
Cut-deges and cut-vertices

Definition

A **cut-edge** or **cut-vertex** of a graph is an edge or vertex whose deletion increases the number of components.
We write $G - e$ or $G - M$ for the subgraph of G obtained by deleting an edge e or set of edges M. Similarly, we write $G - v$ or $G - S$ for the subgraph obtained by deleting a vertex v or set of vertices S.
An **induced subgraph** is a subgraph obtained by deleting a set of vertices. We write $G[T]$ for $G - \bar{T}$, where $\bar{T} = V(G) - T$; this is the subgraph of G induced by T.

Note that when $T \subset V(G)$, the induced subgraph $G[T]$ consists of T and all edges whose endpoints are contained in T.

Example : We consider the same above graph :



It has cut-vertices v and y. Its cut-edges are qr, vw, xy, and yz.

This graph has C_4 [(t, s, u, v)] and P_5 [$\{t, s, u, v, w\}$] as subgraphs but not as induced subgraphs.

The graph P_4 is an induced subgraph ; it is the subgraph induced by $\{s, t, v, w\}$.

Characterization of cut-edges in terms of cycles

Theorem

An edge is a cut-edge if and only if it belongs to no cycle.

Proof :

Outline

Definition : A walk is **odd** (resp. **even**) if its length is odd (resp. even). A closed walk W **contains** a cycle C if the vertices and edges of C occur as a sublist of W, in cyclic order but not necessarily consecutive.

Lemma

Every closed odd walk contains an odd cycle.

Proof : By induction on the length l of a closed odd walk.

Theorem (König [1936])

A graph is bipartite if and only if it has no odd cycle.

Proof :

Remark

- The theorem implies that we can prove that a graph G is not bipartite by finding an odd cycle in G ; this is much easier than examining all possible bipartitions to prove that none work.
- When we want to prove that G is bipartite, we define a bipartition and prove that the two sets are independent ; this is easier than examining all cycles.

Outline

Definition

A graph is **Eulerian** if it has a closed trail containing all edges.
We call a closed trail a **circuit** when we do not specify the first vertex but keep the list in cyclic order.
An **Eulerian circuit** or **Eulerian trail** in a graph is a circuit or trail containing all the edges.
An **even graph** is a graph with vertex degrees all even. A vertex is **odd** (resp. **even**) when its degree is odd (resp. even).

A **maximal path** in a graph G is a path P in G that is not contained in a longer path. When a graph is finite, no path can extend forever, so maximal (non-extendible) paths exist.

Lemma

If every vertex of a graph G has degree at least 2, then G contains a cycle.

Proof : by considering a maximal path P in G, and an endpoint of P.

Theorem

A graph G is Eulerian if and only if it has at most one nontrivial component and its vertices all have even degree.

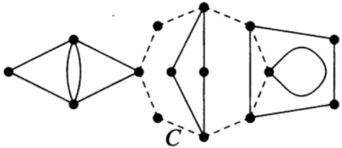
Proof of the theorem

Necessity. Suppose that G has an Eulerian circuit C. Each passage of C through a vertex uses two incident edges, and the first edge is paired with the last at the first vertex. Hence every vertex has even degree. Also, two edges can be in the same trail only when they lie in the same component, so there is at most one nontrivial component.

Proof (Cont.)

Sufficiency. Assuming that the condition holds, we obtain an Eulerian circuit using induction on the number of edges, m.
Basis step : $m = 0$. A closed trail consisting of one vertex suffices.
Induction step : $m > 0$. With even degrees, each vertex in the nontrivial component of G has degree at least 2. By the previous Lemma, the nontrivial component has a cycle C. Let G' be the graph obtained from G by deleting $E(C)$. Since C has 0 or 2 edges at each vertex, each component of G' is also an even graph. Since each component also is connected and has less than m edges, we can apply the induction hypothesis to conclude that each component of G' has an Eulerian circuit. To combine these into an Eulerian circuit of G, we traverse C, but when a component of G' is entered for the first time we detour along an Eulerian circuit of that component. This circuit ends at the vertex where we began the detour. When we complete the traversal of C, we have completed an Eulerian circuit of G.

Illustration of the Eulerian circuit constructed :



Proposition

Every even graph decomposes into cycles.

Proof : by induction on the number of edges.

Proposition

If G is a simple graph in which every vertex has degree at least k, then G contains a path of length at least k. If $k \geq 2$, then G also contains a cycle of length at least $k+1$.

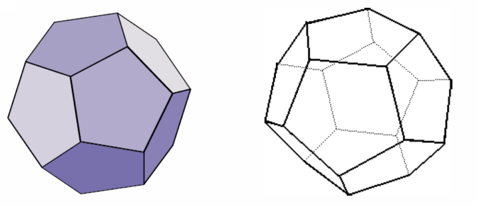
Proof : Consider an endpoint of a maximal path P.

Outline

Definition

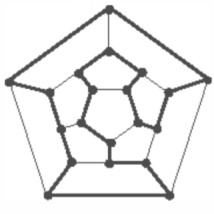
A graph G is **Hamiltonian** if it contains a spanning cycle, which is a cycle that goes through each vertex of G exactly once. Such cycle is also called a **Hamiltonian cycle**.

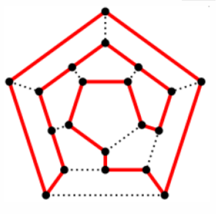
Example : The notion of a Hamiltonian cycle has its origins in a game invented in 1859 by Hamilton : The game consists of a regular wooden dodecahedron (polyhedron with 12 faces and 20 vertices), as shown in the following figure



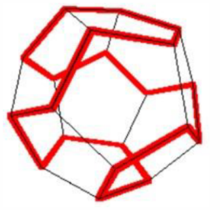
The player must go through the twenty vertices exactly once following the edges of the dodecahedron and return to his starting point.

Some solutions in 2D :





In 3D :



Sufficient conditions

Theorem (Dirac [1952])

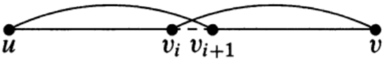
If G is a simple graph with $n \geq 3$ vertices and $\forall v \in V(G), \deg(v) \geq n/2$, then G is Hamiltonian.

Proof

The proof uses contradiction and extremality. If there is a non-Hamiltonian graph satisfying the hypotheses, then adding edges cannot reduce the minimum degree. Thus we may restrict our attention to maximal non-Hamiltonian graphs with minimum degree at least $n/2$, where "maximal" means that adding any edge joining nonadjacent vertices creates a spanning cycle.
When $uv \notin E(G)$, the maximality of G implies that $G + uv$ contains a Hamiltonian cycle and that cycle contains the edge uv. This implies that G has a spanning path $v_1, \cdots, v_n$ from $u = v_1$ to $v = v_n$. To prove the theorem, it suffices to make a small change in this cycle to avoid using the edge uv; this will build a spanning cycle in G.

Proof (Cont.)

If a neighbor of u directly follows a neighbor of v on the path, such as $uv_{i+1} \in E(G)$ and $vv_i \in E(G)$, then $(u, v_{i+1}, v_{i+2}, \cdots, v, v_i, v_{i-1}, \cdots, u)$ is a spanning cycle in G.



To prove that such a cycle exists, we show that there is a common index in the sets S and T defined by $S = \{i : uv_{i+1} \in E(G), 1 \leq i \leq n-2\}$ and $T = \{i : vv_i \in E(G), 2 \leq i \leq n-1\}$. Summing the sizes of these sets yields

$$|S \cup T| + |S \cap T| = |S| + |T| = \deg(u) + \deg(v) \geq n/2 + n/2 = n.$$

We have $S \cup T \subset [\![1, n-1]\!]$, then $|S \cup T| \leq n-1$, hence $|S \cap T| \geq 1$. This means that $S \cap T \neq \emptyset$.
So, we have established a contradiction by finding a spanning cycle in G; hence there is no (maximal) non-Hamiltonian graph satisfying the hypotheses.

Remark

Ore observed that the argument uses $\forall v \in V(G), \deg(v) \geq n/2$ only to show that $\deg(u) + \deg(v) \geq n$. Therefore, we can weaken the requirement of minimum degree $n/2$ to require only that $\deg(u) + \deg(v) \geq n$ whenever $uv \notin E(G)$.

Theorem (Ore [1960])

If G is a simple graph with $n \geq 3$ vertices and $\deg(u) + \deg(v) \geq n$ for every pair of distinct non-adjacent vertices u and v of G, then G is Hamiltonian.

Proof : The same as the above proof of Dirac's theorem using the above remark.