Graph Theory

Chapter 3. Connected Graphs

3.3. Euler Tours—Proofs of Theorems

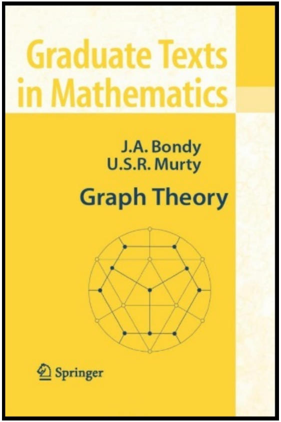


Table of contents

Lemma 3.3.A

Lemma 3.3.A. If G is an Eulerian graph then G is even.

Proof. Let the walk $W = v_0e_1v_1e_2v_2 \cdots v_{m-1}e_mv_0$ be an Euler tour of G. The internal vertices of W, namely $v_1, v_2, \ldots, v_{m-1}$, are incident with edges in W 2 at a time; for $i = 1, 2, \ldots, m-1$ we have v_i is incident with edges e_i and e_{i+1}. Since the edges are distinct in an Euler tour then, each internal vertex in an Euler tour (i.e., each internal vertex of W) is of even degree.

Lemma 3.3.A

Lemma 3.3.A. If G is an Eulerian graph then G is even.

Proof. Let the walk $W = v_0 e_1 v_1 e_2 v_2 \cdots v_{m-1} e_m v_0$ be an Euler tour of G. The internal vertices of W, namely $v_1, v_2, \ldots, v_{m-1}$, are incident with edges in W 2 at a time; for $i = 1, 2, \ldots, m-1$ we have v_i is incident with edges e_i and e_{i+1}. Since the edges are distinct in an Euler tour then, each internal vertex in an Euler tour (i.e., each internal vertex of W) is of even degree. Now v_0 may also appear in the set of vertices $\{v_1, v_2, \ldots, v_{m-1}\}$ and it will be incident to an even number of edges in the counting process used for these vertices. But v_0 is also incident to edges e_1 and e_m, so its total degree is even as well. That is (since G is connected, by the definition of "tour"), each vertex of G is of even degree and G is an even graph, as claimed. □

Lemma 3.3.A

Lemma 3.3.A. If G is an Eulerian graph then G is even.

Proof. Let the walk $W = v_0 e_1 v_1 e_2 v_2 \cdots v_{m-1} e_m v_0$ be an Euler tour of G. The internal vertices of W, namely $v_1, v_2, \ldots, v_{m-1}$, are incident with edges in W 2 at a time; for $i = 1, 2, \ldots, m-1$ we have v_i is incident with edges e_i and e_{i+1}. Since the edges are distinct in an Euler tour then, each internal vertex in an Euler tour (i.e., each internal vertex of W) is of even degree. Now v_0 may also appear in the set of vertices $\{v_1, v_2, \ldots, v_{m-1}\}$ and it will be incident to an even number of edges in the counting process used for these vertices. But v_0 is also incident to edges e_1 and e_m, so its total degree is even as well. That is (since G is connected, by the definition of "tour"), each vertex of G is of even degree and G is an even graph, as claimed. □

Theorem 3.4

Theorem 3.4. If G is a connected even graph, then the walk W returned by Fleury's Algorithm is an Euler tour of G.

Proof. Since the algorithm chooses an edge to add to the walk W under construction and then deletes that edge (when replacing F by $F \setminus e$) from those which may be chosen in subsequent steps, then the edges of walk W must be distinct and so the walk is a trail throughout the procedure.

Theorem 3.4

Theorem 3.4. If G is a connected even graph, then the walk W returned by Fleury's Algorithm is an Euler tour of G.

Proof. Since the algorithm chooses an edge to add to the walk W under construction and then deletes that edge (when replacing F by $F \setminus e$) from those which may be chosen in subsequent steps, then the edges of walk W must be distinct and so the walk is a trail throughout the procedure. The algorithm starts with initial vertex u of W, so in graph F we have $d_F(u) = d_G(u) - 1$ initially and remains so unless the walk returns to u and continues on, so that $d_F(u)$ drops by 2 and $d_F(u)$ remains odd. Since G is an even graph by hypothesis, the algorithm cannot terminate at some $x \neq u$ since such vertex x is of even degree in G and when W has x as its terminal vertex we then have $d_F(x)$ odd so that $\partial_F(x) \neq \varnothing$ and the algorithm does not end. So the algorithm and the walk W produced can only terminate at vertex u. Hence the algorithm produces a closed trail of G with vertex u as its initial and terminal vertex.

Theorem 3.4

Theorem 3.4. If G is a connected even graph, then the walk W returned by Fleury's Algorithm is an Euler tour of G.

Proof. Since the algorithm chooses an edge to add to the walk W under construction and then deletes that edge (when replacing F by $F \setminus e$) from those which may be chosen in subsequent steps, then the edges of walk W must be distinct and so the walk is a trail throughout the procedure. The algorithm starts with initial vertex u of W, so in graph F we have $d_F(u) = d_G(u) - 1$ initially and remains so unless the walk returns to u and continues on, so that $d_F(u)$ drops by 2 and $d_F(u)$ remains odd. Since G is an even graph by hypothesis, the algorithm cannot terminate at some $x \neq u$ since such vertex x is of even degree in G and when W has x as its terminal vertex we then have $d_F(x)$ odd so that $\partial_F(x) \neq \varnothing$ and the algorithm does not end. So the algorithm and the walk W produced can only terminate at vertex u. Hence the algorithm produces a closed trail of G with vertex u as its initial and terminal vertex. We now need to confirm that W includes all edges of G. We do so with a proof by contradiction.

Theorem 3.4

Theorem 3.4. If G is a connected even graph, then the walk W returned by Fleury's Algorithm is an Euler tour of G.

Proof. Since the algorithm chooses an edge to add to the walk W under construction and then deletes that edge (when replacing F by $F \setminus e$) from those which may be chosen in subsequent steps, then the edges of walk W must be distinct and so the walk is a trail throughout the procedure. The algorithm starts with initial vertex u of W, so in graph F we have $d_F(u) = d_G(u) - 1$ initially and remains so unless the walk returns to u and continues on, so that $d_F(u)$ drops by 2 and $d_F(u)$ remains odd. Since G is an even graph by hypothesis, the algorithm cannot terminate at some $x \neq u$ since such vertex x is of even degree in G and when W has x as its terminal vertex we then have $d_F(x)$ odd so that $\partial_F(x) \neq \varnothing$ and the algorithm does not end. So the algorithm and the walk W produced can only terminate at vertex u. Hence the algorithm produces a closed trail of G with vertex u as its initial and terminal vertex. We now need to confirm that W includes all edges of G. We do so with a proof by contradiction.

Theorem 3.4 (continued 1)

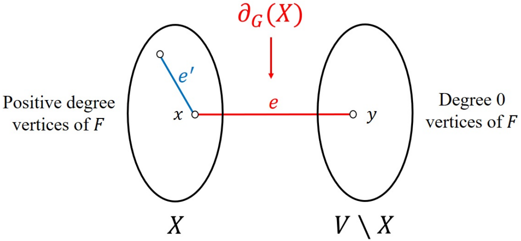
Proof (continued). ASSUME that W, the walk produced by Fleury's Algorithm, is not an Euler tour of G. Let X be the set of vertices of positive degree in subgraph F at the stage when the algorithm terminates. Then $X \neq \varnothing$ (since W is assumed to omit some edge(s) of G), G is even by hypothesis, and W determines an even induced subgraph of G, so the induced subgraph $F[X]$ is an even subgraph of G. As described above, the algorithm must terminate at vertex u so $d_F(u) = 0$, $u \notin X$, and so $u \in V \setminus X$ so that $V \setminus X \neq \varnothing$. Now $\partial_G(X) \neq \varnothing$, or else X and $V \setminus X$ would form a "separation" of G, but G is hypothesized to be connected. But $\partial_F(X) = \varnothing$ since each vertex of $V \setminus X$ has degree 0 in F. Since the algorithm selects edges for inclusion in W (and then deletes those edges in the creation of F), all of the edges of $\partial_G(X)$ must have been chosen for W since, when the algorithm ended, we had $\partial_F(X) = \varnothing$. Let $e = xy$ be the last edge of $\partial_G(X)$ chosen for inclusion in W, where $x \in X$ and $y \in V \setminus X$.

Theorem 3.4 (continued 1)

Proof (continued). ASSUME that W, the walk produced by Fleury's Algorithm, is not an Euler tour of G. Let X be the set of vertices of positive degree in subgraph F at the stage when the algorithm terminates. Then $X \neq \varnothing$ (since W is assumed to omit some edge(s) of G), G is even by hypothesis, and W determines an even induced subgraph of G, so the induced subgraph $F[X]$ is an even subgraph of G. As described above, the algorithm must terminate at vertex u so $d_F(u) = 0$, $u \notin X$, and so $u \in V \setminus X$ so that $V \setminus X \neq \varnothing$. Now $\partial_G(X) \neq \varnothing$, or else X and $V \setminus X$ would form a "separation" of G, but G is hypothesized to be connected. But $\partial_F(X) = \varnothing$ since each vertex of $V \setminus X$ has degree 0 in F. Since the algorithm selects edges for inclusion in W (and then deletes those edges in the creation of F), all of the edges of $\partial_G(X)$ must have been chosen for W since, when the algorithm ended, we had $\partial_F(X) = \varnothing$. Let $e = xy$ be the last edge of $\partial_G(X)$ chosen for inclusion in W, where $x \in X$ and $y \in V \setminus X$.

Theorem 3.4 (continued 2)

Proof (continued). At the step when e was chosen, graph F must have included edge e and edge e must be a cut edge of F (since this is the step at which the last edge of $\partial_G(X)$ was added to W so that $c(F \setminus e) = c(F) + 1$ because $F \setminus e$ has a "new" component which is a subset of $V \setminus X$ and includes y; see the figure below).



But when the algorithm ended, the degree of x in the final graph F was positive so that there was another choice of an edge e' to add to W when cut edge e was chosen, since $d_F(x) > 0$ when the algorithm ends.

Theorem 3.4 (continued 3)

Theorem 3.4. If G is a connected even graph, then the walk W returned by Fleury's Algorithm is an Euler tour of G.

Proof (continued). As argued above, all vertices of X are of even positive degree in $F[X]$ when the algorithm ends. So by Exercise 3.2.3(a), the connected component of $F \setminus e$ containing vertex x has no cut edges so that e' is not a cut edge of $F \setminus e$ (and so not of F itself before edge e was chosen). But this violates the algorithm since it will not add cut edge e to W since non-cut edge e' is available for inclusion in W, a CONTRADICTION. So the assumption that the walk produced by the algorithm is not an Euler tour is false. Hence, an Euler tour of G is produced by the algorithm, as claimed. □

Theorem 3.4 (continued 3)

Theorem 3.4. If G is a connected even graph, then the walk W returned by Fleury's Algorithm is an Euler tour of G.

Proof (continued). As argued above, all vertices of X are of even positive degree in $F[X]$ when the algorithm ends. So by Exercise 3.2.3(a), the connected component of $F \setminus e$ containing vertex x has no cut edges so that e' is not a cut edge of $F \setminus e$ (and so not of F itself before edge e was chosen). But this violates the algorithm since it will not add cut edge e to W since non-cut edge e' is available for inclusion in W, a CONTRADICTION. So the assumption that the walk produced by the algorithm is not an Euler tour is false. Hence, an Euler tour of G is produced by the algorithm, as claimed. □