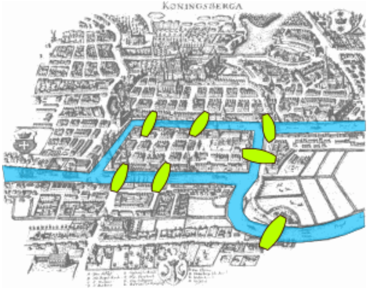
3-10 Traversability

(Part I: Eulerian Graphs)

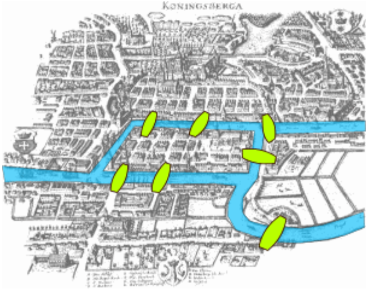
Hengfeng Wei

hfwei@nju.edu.cn

December 03, 2018



Leonhard Euler (1707 – 1783)



Leonhard Euler (1707 – 1783)

Graph Theory Topology


CCTV 14
CNTV
大风车


CCTV 14
CNTV
大风车


田


CCTV 14
CNTV
大风车


田

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873);

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G.

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G.

$$\text{Inductive Step}: m = k + 1$$

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G.

$$\text{Inductive Step}: m = k + 1$$

Let $C = u \sim v$ be any trail in G of the maximum length.

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G.

$$\text{Inductive Step} : m = k + 1$$

Let $C = u \sim v$ be any trail in G of the maximum length.

$$v = u$$

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian if and only if the degree of each vertex of G is even.

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G.

$$\text{Inductive Step}: m = k + 1$$

$$\text{Let } C = u \sim v \text{ be any trail in } G \text{ of the maximum length.}$$

$$v = u \implies C = u \sim u$$

Theorem (Leonhard Euler 1735)

A connected graph G is Eulerian *if and only if the degree of each vertex of G is even.*

"$\Longleftarrow$" (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G.

$$\text{Inductive Step} : m = k + 1$$

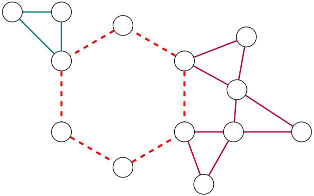
Let $C = u \sim v$ be any trail in G of the maximum length.

$$v = u \implies C = u \sim u$$

$$H = G - E(C) = \bigcup H_i$$

$\square$

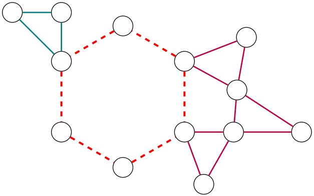
$$H = G - E(C) = \bigcup H_i$$



$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

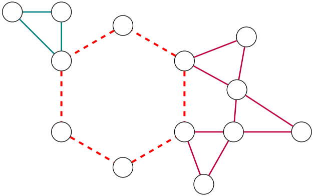
(II) $\forall i : \left|E(H_i)\right| < m$



$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

(II) $\forall i : \Big| E(H_i) \Big| < m$

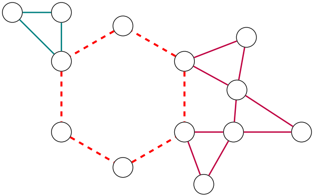


By I.H., each H_i has an Eulerian circuit C_i.

$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

(II) $\forall i : \left| E(H_i) \right| < m$



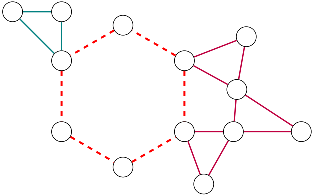
By I.H., each H_i has an Eulerian circuit C_i.

$$\forall i : V(H_i) \cap V(C) \neq \emptyset$$

$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

(II) $\forall i : \left| E(H_i) \right| < m$

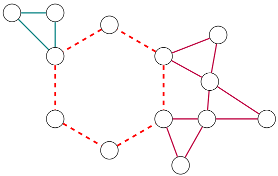


By I.H., each H_i has an Eulerian circuit C_i.

$$\forall i : V(H_i) \cap V(C) \neq \emptyset$$

Glue together each C_i with C to get an Eulerian circuit of G.

procedure HIERHOLZER(G)
  $u \in V(G)$
  $C \leftarrow$ any circuit $u \sim u$ in G
  while $\exists v \in C : \deg(v) > 0$ **do**
    $H \leftarrow G - E(C)$
    $v \leftarrow$ any vertex in $V(C)$ such that $\deg(v) > 0$
    $C' \leftarrow$ any circuit $v \sim v$ in H
    $C \leftarrow C \otimes C'$ ▷ Glue $C' = v \sim v$ with C via v
  return C



procedure $\textsc{Hierholzer}(G)$

  $u \in V(G)$

  $C \leftarrow$ any circuit $u \sim u$ in G

  while $\exists v \in C : \deg(v) > 0$ **do**

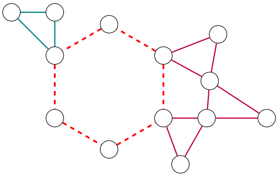
    $H \leftarrow G - E(C)$

    $v \leftarrow$ any vertex in $V(C)$ such that $\deg(v) > 0$

    $C' \leftarrow$ any circuit $v \sim v$ in H

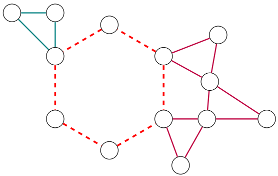
    $C \leftarrow C \otimes C'$ $\triangleright$ Glue $C' = v \sim v$ with C via v

  return C



Q : Time Complexity?

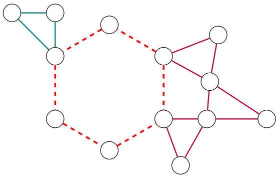
```
procedure HIERHOLZER(G)
    u ∈ V(G)
    C ← any circuit u ∼ u in G
    while ∃v ∈ C : deg(v) > 0 do
        H ← G − E(C)
        v ← any vertex in V(C) such that deg(v) > 0
        C′ ← any circuit v ∼ v in H
        C ← C ⊗ C′                    ▷ Glue C′ = v ∼ v with C via v
    return C
```



Q : Time Complexity?

Q : Data Structures?

```
procedure HIERHOLZER(G)
    u ∈ V(G)
    C ← any circuit u ∼ u in G
    while ∃v ∈ C : deg(v) > 0 do
        H ← G − E(C)
        v ← any vertex in V(C) such that deg(v) > 0
        C′ ← any circuit v ∼ v in H
        C ← C ⊗ C′                      ▷ Glue C′ = v ∼ v with C via v
    return C
```



Q : Time Complexity?

Q : Data Structures?

$O(m)$: Using doubly linked list

Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

(a) e_{i+1} is incident with v_i

(b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented

Fleury's Algorithm (1883)

(I) $v_0 \in V(G); \quad P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

(a) e_{i+1} is incident with v_i

(b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented

Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $\quad P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

(a) e_{i+1} is incident with v_i

(b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented

Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

(a) e_{i+1} is incident with v_i

(b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented

Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

(a) e_{i+1} is incident with v_i

(b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i *is incident with* ≤ 1 *bridge in* $E(G) - \{e_1, e_2, \cdots, e_i\}$.

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

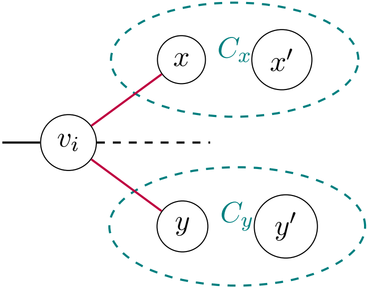
Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

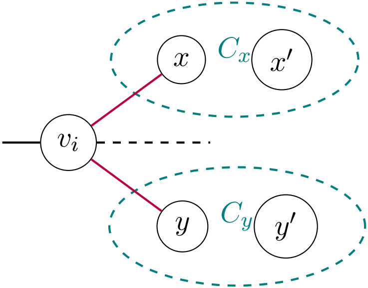


Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.



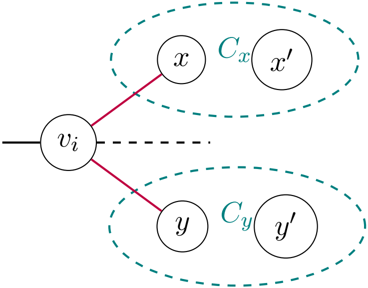
$\exists x' \in C_x : \deg(x)$ is odd

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.



$\exists x' \in C_x : \deg(x)$ is odd

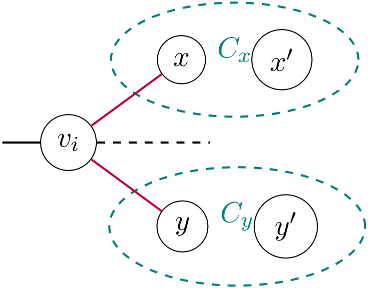
$\exists y' \in C_y : \deg(y)$ is odd

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.



$\exists x' \in C_x : \deg(x)$ is odd

$\exists y' \in C_y : \deg(y)$ is odd

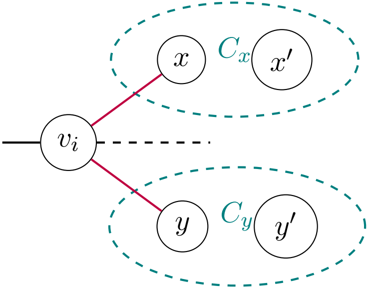
We have found 2 odd vertices.

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.



$\exists x' \in C_x : \deg(x)$ is odd

$\exists y' \in C_y : \deg(y)$ is odd

We have found 2 odd vertices.

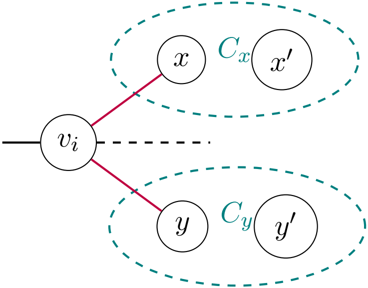
Q : What is the contradiction?

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \cdots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \cdots, e_i\}$.



$\exists x' \in C_x : \deg(x)$ is odd

$\exists y' \in C_y : \deg(y)$ is odd

We have found 2 odd vertices.

Q : What is the contradiction?

Is $\deg(v_i)$ odd or even?

procedure FLEURY(G)

$v_0 \in V(G)$ ▷ Choose any starting vertex

$C \leftarrow v_0$ ▷ Keep track of the circuit

$i \leftarrow 0,\ V_0 \leftarrow V(G),\ E_0 \leftarrow E(G)$

```
procedure FLEURY(G)
    v_0 ∈ V(G)                                 ▷ Choose any starting vertex
    C ← v_0                                    ▷ Keep track of the circuit
    i ← 0, V_0 ← V(G), E_0 ← E(G)

    while deg(v_i) > 0 in E_i do               ▷ Stop otherwise

    return C
```

procedure FLEURY(G)

$v_0 \in V(G)$ ▷ Choose any starting vertex

$C \leftarrow v_0$ ▷ Keep track of the circuit

$i \leftarrow 0,\ V_0 \leftarrow V(G),\ E_0 \leftarrow E(G)$

while $\deg(v_i) > 0$ in E_i **do** ▷ Stop otherwise

if $\deg(v_i) = 1$ in E_i **then** ▷ No alternative: go the bridge

$e_{i+1} \triangleq v_i v_{i+1}$

▷ Delete the isolated vertex v_i

else ▷ Have alternatives: don't go the bridge

Choose $e_{i+1} \triangleq v_i v_{i+1}$ that is not a bridge of (V_i, E_i)

▷ No isolated vertex produced

return C

procedure FLEURY(G)

  $v_0 \in V(G)$ ▷ Choose any starting vertex

  $C \leftarrow v_0$ ▷ Keep track of the circuit

  $i \leftarrow 0,\ V_0 \leftarrow V(G),\ E_0 \leftarrow E(G)$

  while $\deg(v_i) > 0$ in E_i **do** ▷ Stop otherwise

    if $\deg(v_i) = 1$ in E_i **then** ▷ No alternative: go the bridge

      $e_{i+1} \triangleq v_i v_{i+1}$

      ▷ Delete the isolated vertex v_i

    else ▷ Have alternatives: don't go the bridge

      Choose $e_{i+1} \triangleq v_i v_{i+1}$ that is not a bridge of (V_i, E_i)

      ▷ No isolated vertex produced

    $C \leftarrow C e_{i+1} v_{i+1}$

    $E_{i+1} \leftarrow E_i - \{e_{i+1}\}$

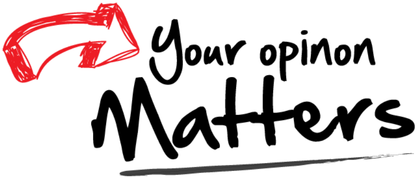
    $i \leftarrow i + 1$

  return C

```
procedure FLEURY(G)
    v_0 ∈ V(G)                                   ▷ Choose any starting vertex
    C ← v_0                                      ▷ Keep track of the circuit
    i ← 0, V_0 ← V(G), E_0 ← E(G)

    while deg(v_i) > 0 in E_i do                 ▷ Stop otherwise
        if deg(v_i) = 1 in E_i then              ▷ No alternative: go the bridge
            e_{i+1} ≜ v_i v_{i+1}
            V_{i+1} ← V_i − {v_i}                ▷ Delete the isolated vertex v_i
        else                                     ▷ Have alternatives: don't go the bridge
            Choose e_{i+1} ≜ v_i v_{i+1} that is not a bridge of (V_i, E_i)
            V_{i+1} ← V_i                        ▷ No isolated vertex produced

        C ← C e_{i+1} v_{i+1}
        E_{i+1} ← E_i − {e_{i+1}}
        i ← i + 1

    return C
```

complexity


THANK YOU



Office 302

Mailbox: H016

hfwei@nju.edu.cn