Algorithmic Graph Theory

Sommer Term 2015

Lecture #10

Testing Planarity

Prof. Rajasekhar Inkulu

IIT Guwahati
Planarity Test

**Theorem.** [Hopcroft & Tarjan, J. ACM 1974]
Let $G$ be a simple undirected graph with $n$ vertices. The planarity of $G$ can be tested in $O(n)$ time.

John Edward Hopcroft
*1939, Seattle, WA, U.S.A.

Robert Endre Tarjan
*1948 Pomona, CA, USA
Planarity Test

**Theorem.** [Hopcroft & Tarjan, J. ACM 1974]
Let $G$ be a simple undirected graph with $n$ vertices. The planarity of $G$ can be tested in $O(n)$ time.

Planarity Test

Theorem. [Hopcroft & Tarjan, J. ACM 1974]
Let $G$ be a simple undirected graph with $n$ vertices. The planarity of $G$ can be tested in $O(n)$ time.


We will treat a simpler, but slower algorithm.
Planarity Test II

**Theorem.** [Auslander & Parter 1961]
Let $G$ be a simple undirected graph with $n$ vertices. The planarity of $G$ can be tested in $O(n^3)$ time.
Planarity Test II

**Theorem.** [Auslander & Parter 1961]
Let $G$ be a simple undirected graph with $n$ vertices. The planarity of $G$ can be tested in $O(n^3)$ time.

**Observation.** $G$ planar $\iff$ each of its connected components is planar.
Planarity Test II

**Theorem.** [Auslander & Parter 1961]
Let \( G \) be a simple undirected graph with \( n \) vertices. The planarity of \( G \) can be tested in \( O(n^3) \) time.

\[ G \text{ planar } \iff \text{ each of its connected components is planar}. \]

\[ \Rightarrow \text{ It suffices to treat connected graphs.} \]
2-Connectivity

**Claim.** $G$ planar $\iff$ each of its 2-connected components (2CC) is planar.
2-Connectivity

**Claim.** \( G \) planar \( \iff \) each of its 2-connected components (2CC) is planar.
2-Connectivity

**Claim.** $G$ planar $\iff$ each of its 2-connected components (2CC) is planar.

Maximal vertex set $K \subseteq V$ such that $G[K]$ is 2-connected.

The 2CC are connected via *cut vertices*.

The adjacency graph of the 2CC is the so-called *2-block tree*. 
2-Connectivity

**Claim.** $G$ planar $\iff$ each of its 2-connected components (2CC) is planar.

The 2CC are connected via *cut vertices*.

The adjacency graph of the 2CC is the so-called **2-block tree**.

Maximal vertex set $K \subseteq V$ such that $G[K]$ is 2-connected.
2-Connectivity

Claim. \( G \) planar \( \iff \) each of its 2-connected components (2CC) is planar.

The 2CC are connected via cut vertices.

The adjacency graph of the 2CC is the so-called 2-block tree.

Maximal vertex set \( K \subseteq V \) such that \( G[K] \) is 2-connected.
2-Connectivity

Claim. \( G \) planar \( \iff \) each of its 2-connected components (2CC) is planar.

The 2CC are connected via cut vertices.

The adjacency graph of the 2CC is the so-called 2-block tree.

Maximal vertex set \( K \subseteq V \) such that \( G[K] \) is 2-connected.
2-Connectivity

Claim. $G$ planar $\iff$ each of its 2-connected components (2CC) is planar.

The 2CC are connected via cut vertices.
The adjacency graph of the 2CC is the so-called 2-block tree.

Maximal vertex set $K \subseteq V$ such that $G[K]$ is 2-connected.

$\Rightarrow$ suffices to consider 2-connected graphs!
Strategy

**Aim.** Planarity test for 2-connected graphs.
Strategy

Aim. Planarity test for 2-connected graphs.

Strategy.

- Compute separating cycle and partition graph in pieces.
- Test pieces recursively.
**Def.** Let $C$ be a cycle, and let $e, e' \not\in C$ be edges.
Def. Let $C$ be a cycle, and let $e, e' \not\in C$ be edges.
**Def.** Let $C$ be a cycle, and let $e, e' \notin C$ be edges. We say $e$ and $e'$ are *equivalent* (with respect to $C$), if they are connected by a path that does not touch $C$. 
**Def.** Let $C$ be a cycle, and let $e, e' \notin C$ be edges. We say $e$ and $e'$ are *equivalent* (with respect to $C$), if they are connected by a path that does not touch $C$. We call the resulting equivalence classes *pieces* (w.r.t. $C$).
**Pieces**

**Def.** Let $C$ be a cycle, and let $e, e' \not\in C$ be edges. We say $e$ and $e'$ are *equivalent* (with respect to $C$), if they are connected by a path that does not touch $C$. We call the resulting equivalence classes *pieces* (w.r.t. $C$).
Pieces

**Def.** Let $C$ be a cycle, and let $e, e' \notin C$ be edges. We say $e$ and $e'$ are equivalent (with respect to $C$), if they are connected by a path that does not touch $C$. We call the resulting equivalence classes pieces (w.r.t. $C$).

**Question:** How many pieces are there?
Piecess

**Def.** Let $C$ be a cycle, and let $e, e' \not\in C$ be edges. We say $e$ and $e'$ are *equivalent* (with respect to $C$), if they are connected by a path that does not touch $C$. We call the resulting equivalence classes *pieces* (w.r.t. $C$).
**Pieces**

**Def.** Let $C$ be a cycle, and let $e, e' \not\in C$ be edges. We say $e$ and $e'$ are *equivalent* (with respect to $C$), if they are connected by a path that does not touch $C$. We call the resulting equivalence classes *pieces* (w.r.t. $C$).

**Note.**

Every piece has connectors (on $C$).
**Pieces**

**Def.** Let $C$ be a cycle, and let $e, e' \not\in C$ be edges. We say $e$ and $e'$ are equivalent (with respect to $C$), if they are connected by a path that does not touch $C$. We call the resulting equivalence classes pieces (w.r.t. $C$).

**Note.**
Every piece has $\geq 2$ connectors (on $C$).
**Def.** A cycle $C$ is *separating* if it induces at least two pieces.
Def. A cycle $C$ is *separating* if it induces at least two pieces.

- $C_1$ is separating.
- $C_2$ is not separating.
Existence of a Separating Cycle

**Lem**$_1$. Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is not a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**
Existence of a Separating Cycle

**Lem**$_1$. Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is *not* a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**

Let $u,v$ be consecutive connectors of $P$ in the cyclic order on $C$. 
Existence of a Separating Cycle

**Lem**\textsubscript{1}. Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is *not* a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**

Let $u, v$ be consecutive connectors of $P$ in the cyclic order on $C$.

Consider $u$–$v$ path $\gamma$ on $C$ w/o connectors.
Existence of a Separating Cycle

Lem$_1$. Let $C$ be a non-separating cycle with piece $P$. If $P$ is not a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

Proof.
Let $u, v$ be consecutive connectors of $P$ in the cyclic order on $C$.

Consider $u$–$v$ path $\gamma$ on $C$ w/o connectors.

Let $\pi$ be a $u$–$v$ path in $P$. 
Existence of a Separating Cycle

**Lem** 1. Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is *not* a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**

Let $u, v$ be consecutive connectors of $P$ in the cyclic order on $C$.

Consider $u$–$v$ path $\gamma$ on $C$ w/o connectors.

Let $\pi$ be a $u$–$v$ path in $P$.

Consider cycle $C' := C + \pi - \gamma$. 
Existence of a Separating Cycle

**Lem$_1$.** Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is *not* a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**

Let $u, v$ be consecutive connectors of $P$ in the cyclic order on $C$.

Consider $u$–$v$ path $\gamma$ on $C$ w/o connectors.

Let $\pi$ be a $u$–$v$ path in $P$.

Consider cycle $C' := C + \pi - \gamma$. $\Rightarrow \gamma$ is piece w.r.t. $C'$. 
Existence of a Separating Cycle

**Lem** 1. Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is *not* a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**

Let $u, v$ be consecutive connectors of $P$ in the cyclic order on $C$.

Consider $u$–$v$ path $\gamma$ on $C$ w/o connectors.

Let $\pi$ be a $u$–$v$ path in $P$.

Consider cycle $C' := C + \pi - \gamma$. \(\Rightarrow\) $\gamma$ is piece w.r.t. $C'$.

If $P$ is not a path, there is an edge $e \in E(P) - E(\pi)$. 
Existence of a Separating Cycle

**Lem** Let $C$ be a *non-separating* cycle with piece $P$. If $P$ is *not* a path, $G$ contains a separating cycle $C'$, that consists of a piece of $C$ and a path in $P$ connecting two connectors.

**Proof.**

Let $u, v$ be consecutive connectors of $P$ in the cyclic order on $C$.

Consider $u$–$v$ path $\gamma$ on $C$ w/o connectors.

Let $\pi$ be a $u$–$v$ path in $P$.

Consider cycle $C' := C + \pi - \gamma$. $\Rightarrow \gamma$ is piece w.r.t. $C'$.

If $P$ is not a path, there is an edge $e \in E(P) - E(\pi)$.

Piece $\delta$ that contains $e$ is different from $\gamma$. 
Existence of a Separating Cycle

**Lem**\(^1\). Let \( C \) be a non-separating cycle with piece \( P \). If \( P \) is not a path, \( G \) contains a separating cycle \( C' \), that consists of a piece of \( C \) and a path in \( P \) connecting two connectors.

**Proof.**

Let \( u, v \) be consecutive connectors of \( P \) in the cyclic order on \( C \).

Consider \( u-v \) path \( \gamma \) on \( C \) w/o connectors.

Let \( \pi \) be a \( u-v \) path in \( P \).

Consider cycle \( C' := C + \pi - \gamma \). \( \Rightarrow \) \( \gamma \) is piece w.r.t. \( C' \).

If \( P \) is not a path, there is an edge \( e \in E(P) - E(\pi) \).

Piece \( \delta \) that contains \( e \) is different from \( \gamma \). \( \Rightarrow \) \( C' \) separating. \( \square \)
Conflicting Pieces

$G$ planar $\Rightarrow$ every piece must be embedded either completely inside or completely outside of $C$. 
Conflicting Pieces

$G$ planar $\Rightarrow$ every piece must be embedded either completely inside or completely outside of $C$.

**Obs.** Pieces $P \neq Q$ can be embedded on the same side of $C$. $\Leftrightarrow$ There exists a path $\gamma$ on $C$ such that $\gamma$ contains all connectors of $Q$ but no interior vertex of $\gamma$ is a connector for $P$. 
Conflicting Pieces

\( G \) planar \( \Rightarrow \) every piece must be embedded either completely inside or completely outside of \( C \).

**Obs.**  Pieces \( P \neq Q \) can be embedded on the same side of \( C \).
\[ \Leftrightarrow \] There exists a path \( \gamma \) on \( C \) such that \( \gamma \) contains all connectors of \( Q \) but no interior vertex of \( \gamma \) is a connector for \( P \).

**Def.**  Two pieces that cannot be embedded on the same side of \( C \) are *in conflict*. 
Conflict Graph

**Def.** The *conflict graph* $I$ (w.r.t. $C$) has a vertex for each piece and an edge whenever the two pieces are in conflict.
Conflict Graph

**Def.** The *conflict graph* $I$ (w.r.t. $C$) has a vertex for each piece and an edge whenever the two pieces are in conflict.
Bipartite Conflict Graph

**Lem_2.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and
(ii) the conflict graph $I$ is bipartite.
Bipartite Conflict Graph

**Lem$_2$.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. □
Bipartite Conflict Graph

**Lemma 2.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and
(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. □
Bipartite Conflict Graph

Lem2. Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and
(ii) the conflict graph $I$ is bipartite.

Proof. Exercise. ⊌

$I$ bipartite
⇒ $G$ planar
**Bipartite Conflict Graph**

**Lem$_2$.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. $\square$
**Lemma 2.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. $\square$
Bipartite Conflict Graph

**Lem$_2$.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. □
**Bipartite Conflict Graph**

**Lem$_2$.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. □
Bipartite Conflict Graph

**Lem2.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. □
Bipartite Conflict Graph

**Lemma 2.** Let $G$ be a graph with separating cycle $C$ and conflict graph $I$. Graph $G$ is planar if and only if

(i) for each piece $P$, the graph $C + P$ is planar and

(ii) the conflict graph $I$ is bipartite.

**Proof.** Exercise. □
Computation of the Conflict Graph

**Obs.** The neighbors of a piece $P$ in the conflict graph can be computed in $O(n)$ time if all pieces are known.
Computation of the Conflict Graph

**Obs.** The neighbors of a piece $P$ in the conflict graph can be computed in $O(n)$ time if all pieces are known.

Number vertices of $C$ with numbers $\{0, \ldots, 2k - 1\}$ as depicted ($k = \#$ connectors of $P$).
Computation of the Conflict Graph

**Obs.** The neighbors of a piece $P$ in the conflict graph can be computed in $O(n)$ time if all pieces are known.

Number vertices of $C$ with numbers $\{0, \ldots, 2k - 1\}$ as depicted ($k = \#$ connectors of $P$).

Piece $Q$ is *not* in conflict with $P$ if there exists $i$ such that all connectors of $Q$ lie in the interval $[2i, (2i + 2) \mod (2k + 2)]$. 

$P$, $Q$, $Q'$, and $C$ are depicted with vertex labels to illustrate the neighbors and conflict conditions.
**Computation of the Conflict Graph**

**Obs.** The neighbors of a piece $P$ in the conflict graph can be computed in $O(n)$ time if all pieces are known.

Number vertices of $C$ with numbers $\{0, \ldots, 2k - 1\}$ as depicted ($k = \# \text{ connectors of } P$).

Piece $Q$ is *not* in conflict with $P$ if $\exists i \text{ s.t. all connectors of } Q \text{ lie in the interval} [2i, (2i + 2) \mod (2k + 2)]$

**Cor.** The conflict graph can be constructed in $O(n^2)$ time.
Planarity Test

PlanarityTest(2-connected graph $G = (V, E)$, separ. cycle $C$)

compute pieces w.r.t. $C$

foreach piece $P$ which is not a path do

$G' := C + P$

$C' := C - \gamma + \pi$ as in Lemma 1

if PlanarityTest($G'$, $C'$) = false

return false

compute conflict graph $I$

if $I$ is bipartite

return true

else

return false
Planarity Test

PlanarityTest(2-connected graph $G = (V, E)$, separ. cycle $C$)

compute pieces w.r.t. $C$

foreach piece $P$ which is not a path do

$G' := C + P$

$C' := C - \gamma + \pi$ as in Lemma 1

if $I$ is bipartite then

return true

else

return false
Planarity Test

PlanarityTest(2-connected graph $G = (V, E)$, separ. cycle $C$)

compute pieces w.r.t. $C$

forall piece $P$ which is not a path do

$G' := C + P$

$C' := C - \gamma + \pi$ as in Lemma 1

if PlanarityTest($G'$, $C'$) = false then

return false
Planarity Test

PlanarityTest(2-connected graph \( G = (V, E) \), separ. cycle \( C \))

compute pieces w.r.t. \( C \)

\textbf{foreach} piece \( P \) which is not a path \textbf{do}

\[ G' := C + P \]
\[ C' := C - \gamma + \pi \text{ as in Lemma}_1 \]

\textbf{if} PlanarityTest(\( G' \), \( C' \)) = false \textbf{then}

\[ \text{return} \ false \]

compute conflict graph \( I \)
Planarity Test

PlanarityTest(2-connected graph $G = (V, E)$, separ. cycle $C$)

compute pieces w.r.t. $C$

$\textbf{foreach}$ piece $P$ which is not a path $\textbf{do}$

$G' := C + P$

$C' := C - \gamma + \pi$ as in Lemma 1

$\textbf{if}$ PlanarityTest($G'$, $C'$) $= \text{false}$ $\textbf{then}$

$\textbf{return}$ false

compute conflict graph $I$

$\textbf{if}$ $I$ is bipartite $\textbf{then}$

$\textbf{return}$ true

else

$\textbf{return}$ false
Planarity Test

PlanarityTest(2-connected graph $G = (V, E)$, separ. cycle $C$)

compute pieces w.r.t. $C$

foreach piece $P$ which is not a path do

$G' := C + P$

$C' := C - \gamma + \pi$ as in Lemma 1

if PlanarityTest($G'$, $C'$) = false then

return false

compute conflict graph $I$

if $I$ is bipartite then

return true

else

return false

Correctness?
Planarity Test

PlanarityTest(2-connected graph \( G = (V, E) \), separ. cycle \( C \))

compute pieces w.r.t. \( C \)

\textbf{foreach} piece \( P \) which is \textbf{not} a path \textbf{do}

\( G' := C + P \)

\( C' := C - \gamma + \pi \) as in Lemma 1

\textbf{if} PlanarityTest(\( G', C' \)) = false \textbf{then}

\hspace{1em} \textbf{return} false

\textbf{compute conflict graph } \( I \)

\textbf{if } \( I \) is bipartite \textbf{then}

\hspace{1em} \textbf{return} true

\textbf{else}

\hspace{1em} \textbf{return} false

Correctness?
Planarity Test

PlanarityTest(2-connected graph $G = (V, E)$, separ. cycle $C$)

compute pieces w.r.t. $C$

foreach piece $P$ which is not a path do

$G' := C + P$

$C' := C - \gamma + \pi$ as in Lemma 1

if PlanarityTest($G'$, $C'$) = false then

return false

compute conflict graph $I$

if $I$ is bipartite then

return true

else

return false

Correctness?

Exercise:
If $G$ has no separating cycle, $G$ is planar.

Exercise:
$G'$ is 2-connected!
Planarity Test

PlanarityTest(2-connected graph \( G = (V, E) \), separ. cycle \( C \))

compute pieces w.r.t. \( C \)

**foreach** piece \( P \) which is not a path **do**

\[
G' := C + P \quad \text{as in Lemma 1} \\
C' := C - \gamma + \pi 
\]

**if** PlanarityTest\( (G', C') = \text{false} \) **then**

\[ \text{return } \text{false} \]

compute conflict graph \( I \)

**if** \( I \) is bipartite **then**

\[ \text{return } \text{true} \]

**else**

\[ \text{return } \text{false} \]

**Correctness?** By induction on \(|E|\) using Lemma 2.
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar.
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar. So we can assume that $G$ has $O(n)$ edges.
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar.
So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces:
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar. So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces: in $O(n)$ total time by a modification of BFS (don’t explore vertices on $C$).
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar. So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces:
in $O(n)$ total time by a modification of BFS (don’t explore vertices on $C$).

Computing the conflict graph:
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar.
So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces:
in $O(n)$ total time by a modification of BFS
(don’t explore vertices on $C$).

Computing the conflict graph: $O(n^2)$ time
Running Time

If \( G \) has \( \geq 3n - 6 \) edges, then \( G \) not planar.
So we can assume that \( G \) has \( O(n) \) edges.

Computation of the pieces:
in \( O(n) \) total time by a modification of BFS
(don’t explore vertices on \( C \)).

Computing the conflict graph: \( O(n^2) \) time

\[ \Rightarrow \text{Every call (without recursion) takes } O(n^2) \text{ time.} \]
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar.
So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces:
in $O(n)$ total time by a modification of BFS
(don’t explore vertices on $C$).

Computing the conflict graph: $O(n^2)$ time

$\Rightarrow$ Every call (without recursion) takes $O(n^2)$ time.

Number of calls is:
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar. So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces:
in $O(n)$ total time by a modification of BFS (don’t explore vertices on $C$).

Computing the conflict graph: $O(n^2)$ time

$\Rightarrow$ Every call (without recursion) takes $O(n^2)$ time.

Number of calls is:
$O(n)$: associate with each call a unique edge $e \in C' - C$. 
Running Time

If $G$ has $\geq 3n - 6$ edges, then $G$ not planar.
So we can assume that $G$ has $O(n)$ edges.

Computation of the pieces:
in $O(n)$ total time by a modification of BFS (don’t explore vertices on $C$).

Computing the conflict graph: $O(n^2)$ time

$\Rightarrow$ Every call (without recursion) takes $O(n^2)$ time.

Number of calls is:
$O(n)$: associate with each call a unique edge $e \in C' - C$.

Total running time is: $O(n^3)$