Algorithms for graph visualization

Incremental algorithms. Orthogonal drawing.
Definition: Orthogonal Drawing

A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if its vertices are drawn as points and each edge is represented as a sequence of alternating horizontal and vertical segments.
Definition

Definition: Orthogonal Drawing

A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if its vertices are drawn as points and each edge is represented as a sequence of alternating horizontal and vertical segments.
Definition: Orthogonal Drawing

A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if its vertices are drawn as points and each edge is represented as a sequence of alternating horizontal and vertical segments.

Edges lie on the grid, i.e., **bends** lie on grid points.
Definition

Definition: Orthogonal Drawing

A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if its vertices are drawn as points and each edge is represented as a sequence of alternating horizontal and vertical segments.

- Edges lie on the grid, i.e., bends lie on grid points.
- Degree of each vertex has to be at most 4.
Applications

Er diagramm in OGDF

Organigram von HS Limburg

Circuit diagram by Jeff Atwood

UML Diagramm by Oracle

ER diagramm in OGDF

Organigramm von HS Limburg

Circuit diagram by Jeff Atwood

UML Diagramm by Oracle
Applications

Usefull Aesthetic Criteria?
\textbf{Definition: \textit{st}-ordering}

An \textit{st}-\textit{ordering} of a graph \( G = (V, E) \) is an ordering of the vertices \( \{v_1, v_2, \ldots, v_n\} \), such that for each \( j, 2 \leq j \leq n - 1 \), vertex \( v_j \) has at least one neighbour \( v_i \) with \( i < j \), and at least one neighbour \( v_k \) with \( k > j \).
An \textit{st-ordering} of a graph $G = (V, E)$ is an ordering of the vertices $\{v_1, v_2, \ldots, v_n\}$, such that for each $j$, $2 \leq j \leq n - 1$, vertex $v_j$ has at least one neighbour $v_i$ with $i < j$, and at least one neighbour $v_k$ with $k > j$. 

Example of an \textit{st-ordering}
**Definition: st-ordering**

An *st-ordering* of a graph $G = (V, E)$ is an ordering of the vertices $\{v_1, v_2, \ldots, v_n\}$, such that for each $j$, $2 \leq j \leq n - 1$, vertex $v_j$ has at least one neighbour $v_i$ with $i < j$, and at least one neighbour $v_k$ with $k > j$.

**Theorem [Lempel, Even, Cederbaum, 66]**

Let $G$ be a biconnected graph $G$ and let $s, t$ be vertices of $G$. $G$ has an *st-ordering* such that $s$ appears as the first and $t$ as the last vertex in this ordering.
Biedl & Kant Orthogonal Drawing Algorithm
Biedl & Kant Orthogonal Drawing Algorithm

first vertex
Biedl & Kant Orthogonal Drawing Algorithm

first vertex
Biedl & Kant Orthogonal Drawing Algorithm

first vertex
Biedl & Kant Orthogonal Drawing Algorithm

first vertex
indegree = 1
Biedl & Kant Orthogonal Drawing Algorithm

first vertex

indegree = 1
Biedl & Kant Orthogonal Drawing Algorithm

First vertex

Indegree = 1

Indegree = 2
Biedl & Kant Orthogonal Drawing Algorithm

first vertex

indegree = 1

indegree = 2
Biedl & Kant Orthogonal Drawing Algorithm

First vertex

Indegree = 1

Indegree = 2

Indegree = 3
Biedl & Kant Orthogonal Drawing Algorithm

1. first vertex
2. indegree = 1
3. indegree = 2
4. indegree = 3
Biedl & Kant Orthogonal Drawing Algorithm

![Graph Visualization Diagram]

- **first vertex**: 5
- **indegree = 1**: 5
- **indegree = 2**: 5
- **indegree = 3**: 5
- **indegree = 4**: 5

---

*Algorithmen zur Visualisierung von Graphen*

*Marcus Krug*

*Institut für Theoretische Informatik*

*Lehrstuhl Algorithmik I*
Biedl & Kant Orthogonal Drawing Algorithm

<table>
<thead>
<tr>
<th>Lemma (Area of Biedl &amp; Kant drawing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The width is $m - n + 1$ and the height at most $n + 1$.</td>
</tr>
</tbody>
</table>
Lemma (Area of Biedl & Kant drawing)

The width is $m - n + 1$ and the height at most $n + 1$.

Proof

- **Width**: At each step we increase the number of columns by $\text{outdeg}(v_i) - 1$, if $i > 1$ and $\text{outdeg}(v_1)$ for $v_1$. 
Lemma (Area of Biedl & Kant drawing)

The width is $m - n + 1$ and the height at most $n + 1$.

Proof

- **Width**: At each step we increase the number of columns by $\text{outdeg}(v_i) - 1$, if $i > 1$ and $\text{outdeg}(v_1)$ for $v_1$.

- **Height**: Every vertex except for $v_2$ is placed at a new row. Vertex $v_n$ uses one more row if $\text{indeg}(v_n) = 4$. 

Biedl & Kant Orthogonal Drawing Algorithm

Lemma (Area of Biedl & Kant drawing)

The width is \( m - n + 1 \) and the height at most \( n + 1 \).

Proof

- **Width**: At each step we increase the number of columns by \( \text{outdeg}(v_i) - 1 \), if \( i > 1 \) and \( \text{outdeg}(v_1) \) for \( v_1 \).

- **Height**: Every vertex except for \( v_2 \) is placed at a new row. Vertex \( v_n \) uses one more row if \( \text{indeg}(v_n) = 4 \).

Lemma (Number of bends in Biedl & Kant drawing)

There are at most \( 2m - 2n + 4 \) bends.
Biedl & Kant Orthogonal Drawing Algorithm

Lemma (Area of Biedl & Kant drawing)

The width is $m - n + 1$ and the height at most $n + 1$.

Proof

- **Width**: At each step we increase the number of columns by $\text{outdeg}(v_i) - 1$, if $i > 1$ and $\text{outdeg}(v_1)$ for $v_1$.

- **Height**: Every vertex except for $v_2$ is placed at a new row. Vertex $v_n$ uses one more row if $\text{indeg}(v_n) = 4$.

Lemma (Number of bends in Biedl & Kant drawing)

There are at most $2m - 2n + 4$ bends.

Proof

- Each vertex $v_i, i \neq 1, n$, introduces $\text{indeg}(v_i) - 1$ and $\text{outdeg}(v_i) - 1$ new bends.
Biedl & Kant Orthogonal Drawing Algorithm

Lemma (Number of bends per edge in Biedl & Kant drawing)
All edges but one bent at most twice. The exceptional edge bents at most three times.
Biedl & Kant Orthogonal Drawing Algorithm

Lemma (Number of bends per edge in Biedl & Kant drawing)

All edges but one bent at most twice. The exceptional edge bents at most three times.

Proof

Let \((v_i, v_j), i < j, i, j \neq 1, n\). Then \(\text{outdeg}(v_i), \text{indeg}(v_j) \leq 3\). I.e \((v_i, v_j)\) gets at most one bend after placement of \(v_i\) and at most one before placement of \(v_j\). Edges outgoing from \(v_1\) can me made 2-bend by using the column below \(v_1\) for the edge \((v_1, v_2)\).
Biedl & Kant Orthogonal Drawing Algorithm

Lemma (Number of bends per edge in Biedl & Kant drawing)
All edges but one bent at most twice. The exceptional edge bents at most three times.

Proof
- Let \((v_i, v_j), \ i < j, \ i, j \neq 1, n\). Then \(\text{outdeg}(v_i), \ \text{indeg}(v_j) \leq 3\). I.e \((v_i, v_j)\) gets at most one bend after placement of \(v_i\) and at most one before placement of \(v_j\). Edges outgoing from \(v_1\) can me made 2-bend by using the column below \(v_1\) for the edge \((v_1, v_2)\).

Lemma (planarity)
For planar embedded graphs, with \(v_1\) and \(v_n\) on the outer face, the algorithm produces a planar drawing.
Biedl & Kant Orthogonal Drawing Algorithm

Lemma (Number of bends per edge in Biedl & Kant drawing)

All edges but one bent at most twice. The exceptional edge bents at most three times.

Proof

- Let \((v_i, v_j), i < j, i, j \neq 1, n\). Then \(\text{outdeg}(v_i), \text{indeg}(v_j) \leq 3\). I.e \((v_i, v_j)\) gets at most one bend after placement of \(v_i\) and at most one before placement of \(v_j\). Edges outgoing from \(v_1\) can me made 2-bend by using the column below \(v_1\) for the edge \((v_1, v_2)\).

Lemma (planarity)

For planar embedded graphs, with \(v_1\) and \(v_n\) on the outer face, the algorithm produces a planar drawing.

Proof

- Consider a planar embedding of \(G\). Let \(v_1, \ldots, v_n\) be an \(st\)-ordering of \(G\). Let \(G_i\) be the graph induced by \(v_1, \ldots, v_i\). We now prove that if \(G\) is planar, vertex \(v_{i+1}\) lies on the outer face of \(G_i\).
Lemma (planarity)
For planar embedded graphs, with $v_1$ and $v_n$ on the outer face, the algorithm produces a planar drawing.

Proof (Continuation)
- Let $E_i$ be the edges outgoing from the vertices of $G_i$ in the order they appear in the embedded $G$. 

\[
\begin{array}{c}
\text{Proof (Continuation)} \\
\text{Let } E_i \text{ be the edges outgoing from the vertices of } G_i \text{ in the order they appear in the embedded } G.
\end{array}
\]
Lemma (planarity)

For planar embedded graphs, with \( v_1 \) and \( v_n \) on the outer face, the algorithm produces a planar drawing.

Proof (Continuation)

- Let \( E_i \) be the edges outgoing from the vertices of \( G_i \) in the order they appear in the embedded \( G \).
- During the execution of the algorithm we make sure that edges of \( E_i \) appear in the same order in the orthogonal drawing of \( G_i \).
Lemma (planarity)
For planar embedded graphs, with $v_1$ and $v_n$ on the outer face, the algorithm produces a planar drawing.

Proof (Continuation)
- Let $E_i$ be the edges outgoing from the vertices of $G_i$ in the order they appear in the embedded $G$.
- During the execution of the algorithm we make sure that edges of $E_i$ appear in the same order in the orthogonal drawing of $G_i$.
- Since $v_{i+1}$ is on the outer face of $G_i$, it can be placed without creating any crossing.
Lemma (planarity)
For planar embedded graphs, with $v_1$ and $v_n$ on the outer face, the algorithm produces a planar drawing.

Proof (Continuation)
- Let $E_i$ be the edges outgoing from the vertices of $G_i$ in the order they appear in the embedded $G$.
- During the execution of the algorithm we make sure that edges of $E_i$ appear in the same order in the orthogonal drawing of $G_i$.
- Since $v_{i+1}$ is on the outer face of $G_i$, it can be placed without creating any crossing.
Lemma (planarity)

For planar embedded graphs, with $v_1$ and $v_n$ on the outer face, the algorithm produces a planar drawing.

Proof (Continuation)

- Let $E_i$ be the edges outgoing from the vertices of $G_i$ in the order they appear in the embedded $G$.
- During the execution of the algorithm we make sure that edges of $E_i$ appear in the same order in the orthogonal drawing of $G_i$.
- Since $v_{i+1}$ is on the outer face of $G_i$, it can be placed without creating any crossing.
Biedl & Kant Orthogonal Drawing Algorithm

Theorem (Biedl & Kant 98)

A biconnected graph $G$ with vertex-degree at most 4 admits an orthogonal drawing such that:

- Area is $(m - n + 1) \times n + 1$
- Each edge (except maybe for one) has at most 2 bends
- The exceptional edge has at most 3 bends
- The total number of bends is at most $2m - 2n + 4$
- If $G$ is plane, the orthogonal drawing is planar
- Finally, provided an $st$-ordering such a drawing can be constructed in $O(n)$ time.
Theorem (Biedl & Kant 98)

A biconnected graph $G$ with vertex-degree at most 4 admits an orthogonal drawing such that:
- Area is $(m - n + 1) \times n + 1$
- Each edge (except maybe for one) has at most 2 bends
- The exceptional edge has at most 3 bends
- The total number if bends is at most $2m - 2n + 4$
- If $G$ is plane, the orthogonal drawing is planar
- Finally, provided an $st$-ordering such a drawing can be constructed in $O(n)$ time.

For the construction we have used an $st$-ordering of $G$!
Definition: st-digraph

Let $G$ be a directed graph. A vertex $s$ (resp. $t$) is called source (resp. sink) of $G$ if it has only outgoing (resp. incoming edges). A directed acyclic graph with one source and one sink is called \textit{st-digraph}.
**Definition: st-digraph**

Let $G$ be a directed graph. A vertex $s$ (resp. $t$) is called **source** (resp. **sink**) of $G$ if it has only outgoing (resp. incoming) edges. A directed acyclic graph with one source and one sink is called **st-digraph**.

**Definition: topological ordering**

A **topological ordering** of a directed graph $G$ (with $n$ vertices) is an assignment of numbers $\{1, \ldots, n\}$ to the vertices of $G$, such that for every edge $(u, v)$, $\text{number}(v) > \text{number}(u)$.
**Definition: st-digraph**

Let $G$ be a directed graph. A vertex $s$ (resp. $t$) is called **source** (resp. **sink**) of $G$ if it has only outgoing (resp. incoming) edges. A directed acyclic graph with one source and one sink is called **st-digraph**.

**Definition: topological ordering**

A **topological ordering** of a directed graph $G$ (with $n$ vertices) is an assignment of numbers $\{1, \ldots, n\}$ to the vertices of $G$, such that for every edge $(u, v)$, $\text{number}(v) > \text{number}(u)$.
**Definition: st-digraph**

Let $G$ be a directed graph. A vertex $s$ (resp. $t$) is called **source** (resp. **sink**) of $G$ if it has only outgoing (resp. incoming) edges. A directed acyclic graph with one source and one sink is called **st-digraph**.

**Definition: topological ordering**

A **topological ordering** of a directed graph $G$ (with $n$ vertices) is an assignment of numbers $\{1, \ldots, n\}$ to the vertices of $G$, such that for every edge $(u, v)$, $\text{number}(v) > \text{number}(u)$.
st-digraph, topological ordering

Definition: st-digraph
Let $G$ be a directed graph. A vertex $s$ (resp. $t$) is called source (resp. sink) of $G$ if it has only outgoing (resp. incoming) edges. A directed acyclic graph with one source and one sink is called st-digraph.

Definition: topological ordering
A topological ordering of a directed graph $G$ (with $n$ vertices) is an assignment of numbers $\{1, \ldots, n\}$ to the vertices of $G$, such that for every edge $(u, v)$, $\text{number}(v) > \text{number}(u)$.

How to construct a topological ordering?
Construction of an $st$-ordering:

$G$ is undirected biconnected graph
Construction of an \textit{st}-ordering:

\begin{itemize}
  \item[$G$ is undirected biconnected graph]
  \item[Orient edges of $G$]
\end{itemize}
Construction of an \textit{st}-ordering:

\textit{G} is undirected biconnected graph

Orient edges of \textit{G}

\textit{G}' is an \textit{st}-digraph
Construction of an $st$-ordering:

$G$ is undirected biconnected graph

$G'$ is an $st$-digraph

Let $v_1, \ldots, v_n$ be a topological ordering of $G'$

Orient edges of $G'$
Construction of an \textit{st}-ordering:

\(G\) is undirected biconnected graph \hspace{1cm} \text{Orient edges of} \hspace{1cm} \(G'\) is an \textit{st}-digraph

\hspace{1cm} \hspace{1cm}

Let \(v_1, \ldots, v_n\) be a topological ordering of \(G'\)

Since \(G'\) is an \textit{st}-digraph, for \(v_i, (i \neq 1, n) \exists (v_j, v_i)\) and \((v_i, v_k)\). By the property of topological ordering \(j < i\) and \(i < k\).
Construction of an \textit{st}-ordering:

$G$ is undirected biconnected graph

Orient edges of $G$

$G'$ is an \textit{st}-digraph

Let $v_1, \ldots, v_n$ be a topological ordering of $G'$

Since $G'$ is an \textit{st}-digraph, for $v_i$ ($i \neq 1, n$) there exist $(v_j, v_i)$ and $(v_i, v_k)$. By the property of topological ordering $j < i$ and $i < k$.

$v_1, \ldots, v_n$ is an \textit{st}-ordering of $G$
Construction of an $st$-ordering:

$G$ is an undirected biconnected graph

Orient edges of $G$

$G'$ is an $st$-digraph

Let $v_1, \ldots, v_n$ be a topological ordering of $G'$

Since $G'$ is an $st$-digraph, for $v_i (i \neq 1, n) \exists (v_j, v_i)$ and $(v_i, v_k)$. By the property of topological ordering $j < i$ and $i < k$.

$v_1, \ldots, v_n$ is an $st$-ordering of $G$

EXAMPLE
Construction of an $st$-ordering:

1. Let $G$ be an undirected biconnected graph.
2. Orient edges of $G$ to form an $st$-digraph $G'$.
3. Let $v_1, \ldots, v_n$ be a topological ordering of $G'$.

Since $G'$ is an $st$-digraph, for $v_i$ (i ≠ 1, n) ∃ (v_j, v_i) and (v_i, v_k). By the property of topological ordering $j < i$ and $i < k$.

$v_1, \ldots, v_n$ is an $st$-ordering of $G$.

**EXAMPLE**
**Definition: Ear decomposition**

An ear decomposition $D = (P_0, \ldots, P_r)$ of an undirected graph $G = (V, E)$ is a partition of $E$ into an ordered collection of edge disjoint paths $P_0, \ldots, P_r$, such that:

- $P_0$ is an edge
- $P_0 \cup P_1$ is a simple cycle
- both end-vertices of $P_i$ belong to $P_0 \cup \cdots \cup P_{i-1}$
- no internal vertex of $P_i$ belong to $P_0 \cup \cdots \cup P_{i-1}$

An ear decomposition of open if $P_0, \ldots, P_r$ are simple paths.
**Lemma (Ear decomposition)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.
Lemma (Ear decomposition)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.

Proof

- Let $P_0 = (s, t)$ and $P_1$ be path between $s$ and $t$, it exists since $G$ is biconnected.
Lemma (Ear decomposition)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.

Proof

- Let $P_0 = (s, t)$ and $P_1$ be path between $s$ and $t$, it exists since $G$ is biconnected.
- Induction hypothesis: $P_0, \ldots, P_i$ are ears.
Lemma (Ear decomposition)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.

Proof

- Let $P_0 = (s, t)$ and $P_1$ be path between $s$ and $t$, it exists since $G$ is biconnected.
- Induction hypothesis: $P_0, \ldots, P_i$ are ears.
- Let $(u, v)$ be an edge in $G$ such that $u \in P_0 \cup \cdots \cup P_i$ and $v \notin P_0 \cup \cdots \cup P_i$. Let $(u, u')$, such that $u' \in P_0 \cup \cdots \cup P_i$. Let $P$ be a path between $v$ and $u'$, not passing through $u$. $P$ exists since $G$ is biconnected.
Lemma (Ear decomposition)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.

Proof

- Let $P_0 = (s, t)$ and $P_1$ be path between $s$ and $t$, it exists since $G$ is biconnected.
- Induction hypothesis: $P_0, \ldots, P_i$ are ears.
- Let $(u, v)$ be an edge in $G$ such that $u \in P_0 \cup \cdots \cup P_i$ and $v \notin P_0 \cup \cdots \cup P_i$. Let $(u, u')$, such that $u' \in P_0 \cup \cdots \cup P_i$. Let $P$ be a path between $v$ and $u'$, not passing through $u$. $P$ exists since $G$ is biconnected.
**Lemma (Ear decomposition)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.

**Proof**

- Let $P_0 = (s, t)$ and $P_1$ be path between $s$ and $t$, it exists since $G$ is biconnected.
- Induction hypothesis: $P_0, \ldots, P_i$ are ears.
- Let $(u, v)$ be an edge in $G$ such that $u \in P_0 \cup \cdots \cup P_i$ and $v \notin P_0 \cup \cdots \cup P_i$. Let $(u, u')$, such that $u' \in P_0 \cup \cdots \cup P_i$. Let $P$ be a path between $v$ and $u'$, not passing through $u$. $P$ exists since $G$ is biconnected.
- Let $w$ be the first vertex of $P$ that is contained in $P_0 \cup \cdots \cup P_i$. Set $P_{i+1} = (u, v) \cup P(v \cdots w)$. 

\[ P_0 \cup \cdots \cup P_i \]
Lemma (Ear decomposition)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. $G$ has an open ear decomposition $(P_0, \ldots, P_r)$, where $P_0 = (s, t)$.

Proof

- Let $P_0 = (s, t)$ and $P_1$ be path between $s$ and $t$, it exists since $G$ is biconnected.
- Induction hypothesis: $P_0, \ldots, P_i$ are ears.
- Let $(u, v)$ be an edge in $G$ such that $u \in P_0 \cup \cdots \cup P_i$ and $v \notin P_0 \cup \cdots \cup P_i$. Let $(u, u')$, such that $u' \in P_0 \cup \cdots \cup P_i$. Let $P$ be a path between $v$ and $u'$, not passing through $u$. $P$ exists since $G$ is biconnected.
- Let $w$ be the first vertex of $P$ that is contained in $P_0 \cup \cdots \cup P_i$. Set $P_{i+1} = (u, v) \cup P(v - \cdots - w)$. 

\[ P_0 \cup \cdots \cup P_i \]

\[ P \]

\[ w \]

\[ u' \]

\[ u \]

\[ v \]

\[ P_{i+1} \]
Lemma ($st$-orientation)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$. 
**Lemma (st-orientation)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$.

**Proof**

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$. 


Lemma (st-orientation)

Let \( G = (V, E) \) be a biconnected graph and let \( (s, t) \in E \). There is an orientation \( G' \) of \( G \) which represents an st-digraph. \( G' \) is called st-orientation of \( G \).

Proof

- Let \( D = (P_0, \ldots, P_r) \) be an ear decomposition of \( G = (V, E) \). Notice that \( G = P_0 \cup \cdots \cup P_r \).

- Let \( G_i = P_0 \cup \cdots \cup P_i \). We prove that \( G_i \) has an st-orientation by induction on \( i \).
**Lemma (st-orientation)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$.

**Proof**

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.

- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an $st$-orientation by induction on $i$. 

```plaintext
s -- P_0 -- t
```
**Lemma (st-orientation)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$.

**Proof**

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.

- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an $st$-orientation by induction on $i$. 

![Diagram](image-url)
Lemma (st-orientation)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$.

Proof

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.
- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an $st$-orientation by induction on $i$. 

![Diagram](image-url)
**Lemma (st-orientation)**

Let \( G = (V, E) \) be a biconnected graph \( G \) and let \( (s, t) \in E \). There is an orientation \( G' \) of \( G \) which represents an \( st \)-digraph. \( G' \) is called \( st \)-orientation of \( G \).

**Proof**

- Let \( D = (P_0, \ldots, P_r) \) be an ear decomposition of \( G = (V, E) \). Notice that \( G = P_0 \cup \cdots \cup P_r \).
- Let \( G_i = P_0 \cup \cdots \cup P_i \). We prove that \( G_i \) has an \( st \)-orientation by induction on \( i \).
**Lemma (st-orientation)**

Let \( G = (V, E) \) be a biconnected graph \( G \) and let \( (s, t) \in E \). There is an orientation \( G' \) of \( G \) which represents an \( st \)-digraph. \( G' \) is called \( st \)-orientation of \( G \).

**Proof**

- Let \( D = (P_0, \ldots, P_r) \) be an ear decomposition of \( G = (V, E) \). Notice that \( G = P_0 \cup \cdots \cup P_r \).

- Let \( G_i = P_0 \cup \cdots \cup P_i \). We prove that \( G_i \) has an \( st \)-orientation by induction on \( i \).

- Distinguish two cases based on whether \( u \) and \( v \) are connected by a directed path or not.
**Lemma (st-orientation)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an st-digraph. $G'$ is called st-orientation of $G$.

**Proof**

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.

- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an st-orientation by induction on $i$.

- Distinguish two cases based on whether $u$ and $v$ are connected by a directed path or not.
**st-ordering**

**Lemma (st-orientation)**

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$.

**Proof**

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.

- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an $st$-orientation by induction on $i$.

- Distinguish two cases based on whether $u$ and $v$ are connected by a directed path or not.
Lemma (*st*-orientation)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an *st*-digraph. $G'$ is called *st*-orientation of $G$.

Proof

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.
- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an *st*-orientation by induction on $i$.

- Distinguish two cases based on whether $u$ and $v$ are connected by a directed path or not.
- Result is an *st*-digraph.
Lemma ($st$-orientation)

Let $G = (V, E)$ be a biconnected graph $G$ and let $(s, t) \in E$. There is an orientation $G'$ of $G$ which represents an $st$-digraph. $G'$ is called $st$-orientation of $G$.

Proof

- Let $D = (P_0, \ldots, P_r)$ be an ear decomposition of $G = (V, E)$. Notice that $G = P_0 \cup \cdots \cup P_r$.

- Let $G_i = P_0 \cup \cdots \cup P_i$. We prove that $G_i$ has an $st$-orientation by induction on $i$.

- Distinguish two cases based on whether $u$ and $v$ are connected by a directed path or not.

- Result is an $st$-digraph
Construction of an $st$-ordering:

$G$ is undirected biconnected graph

$G'$ is an $st$-digraph

Orient edges of $G$

Let $v_1, \ldots, v_n$ be a topological ordering of $G'$

Since $G'$ is an $st$-digraph, for $v_i$ ($i \neq 1, n$) $\exists (v_j, v_i)$ and $(v_i, v_k)$. By the property of topological ordering $j < i$ and $i < k$.

$v_1, \ldots, v_n$ is an $st$-ordering of $G$
Construction of an $st$-ordering:

1. Orient edges of $G$
2. Ear decomposition of $G$
3. $G'$ is an $st$-digraph
4. Let $v_1, \ldots, v_n$ be a topological ordering of $G'$
5. Since $G'$ is an $st$-digraph, for $v_i$ $(i \neq 1, n) \exists (v_j, v_i)$ and $(v_i, v_k)$. By the property of topological ordering $j < i$ and $i < k$.
6. $v_1, \ldots, v_n$ is an $st$-ordering of $G$
Construction of an $st$-ordering:

1. **$G$ is undirected biconnected graph**
2. **$G'$ is an $st$-digraph**
3. Let $v_1, \ldots, v_n$ be a topological ordering of $G'$

   Since $G'$ is an $st$-digraph, for $v_i$ ($i \neq 1, n$) there exist $(v_j, v_i)$ and $(v_i, v_k)$. By the property of topological ordering, $j < i$ and $i < k$.

   $v_1, \ldots, v_n$ is an $st$-ordering of $G$
Construction of an $st$-ordering:

$G$ is undirected biconnected graph

$G'$ is an $st$-digraph

Let $v_1, \ldots, v_n$ be a topological ordering of $G'$

Since $G'$ is an $st$-digraph, for $v_i$ ($i \neq 1, n$) $\exists (v_j, v_i)$ and $(v_i, v_k)$. By the property of topological ordering $j < i$ and $i < k$.

$v_1, \ldots, v_n$ is an $st$-ordering of $G$
Direct construction of $st$-ordering from ear decomposition
Direct construction of $st$-ordering from ear decomposition

We construct it incrementally, considering $G_i = P_0 \cup \cdots \cup P_i, i = 0, \ldots, r$. 
Direct construction of \( st \)-ordering from ear decomposition

- We construct it incrementally, considering \( G_i = P_0 \cup \cdots \cup P_i, i = 0, \ldots, r \).

- For \( G_1 \), let \( P_1 = \{u_1, \ldots, u_p\} \), here \( u_1 = s \) and \( u_p = t \). The sequence \( L = \{u_1, \ldots, u_p\} \) is an \( st \)-ordering of \( G_1 \).
Direct construction of \( st \)-ordering from ear decomposition

- We construct it incrementally, considering \( G_i = P_0 \cup \cdots \cup P_i, i = 0, \ldots, r \).

- For \( G_1 \), let \( P_1 = \{u_1, \ldots, u_p\} \), here \( u_1 = s \) and \( u_p = t \). The sequence \( L = \{u_1, \ldots, u_p\} \) is an \( st \)-ordering of \( G_1 \).

- Assume that \( L \) contains an \( st \)-ordering of \( G_i \) and let ear \( P_{i+1} = \{v_1, \ldots, v_q\} \). We insert vertices \( v_1, \ldots, v_q \) to \( L \) after vertex \( v_1 \).
**Direct construction of \( st \)-ordering from ear decomposition**

- We construct it incrementally, considering \( G_i = P_0 \cup \cdots \cup P_i, i = 0, \ldots, r \).
- For \( G_1 \), let \( P_1 = \{u_1, \ldots, u_p\} \), here \( u_1 = s \) and \( u_p = t \). The sequence \( L = \{u_1, \ldots, u_p\} \) is an \( st \)-ordering of \( G_1 \).
- Assume that \( L \) contains an \( st \)-ordering of \( G_i \) and let ear \( P_{i+1} = \{v_1, \ldots, v_q\} \). We insert vertices \( v_1, \ldots, v_q \) to \( L \) after vertex \( v_1 \).
Direct construction of \textit{st}-ordering from ear decomposition

- We construct it incrementally, considering $G_i = P_0 \cup \cdots \cup P_i$, $i = 0, \ldots, r$.

- For $G_1$, let $P_1 = \{u_1, \ldots, u_p\}$, here $u_1 = s$ and $u_p = t$. The sequence $L = \{u_1, \ldots, u_p\}$ is an \textit{st}-ordering of $G_1$.

- Assume that $L$ contains an \textit{st}-ordering of $G_i$ and let ear $P_{i+1} = \{v_1, \ldots, v_q\}$. We insert vertices $v_1, \ldots, v_q$ to $L$ after vertex $v_1$.

- **Why this is an \textit{st}-ordering?** Let $G_{i+1}'$ be an \textit{st}-orientation of $G_i$ as constructed in the previous proof. $L$ is a topological ordering of $G_{i+1}'$ and therefore an \textit{st}-ordering of $G_i$. 


Algorithm: $st$-ordering (example)
(Implementation details - Based on DFS)
$st$-ordering: implementation

Algorithm: $st$-ordering (example)
(Implementation details - Based on DFS)
Algorithm: \textit{st}-ordering (example)

(Implementation details - Based on DFS)
**st-ordering: implementation**

**Algorithm: st-ordering (example)**

(Implementation details - Based on DFS)
**Algorithm: st-ordering (example)**
(Implementation details - Based on DFS)
**st-ordering: implementation**

**Algorithm: st-ordering (example)**

(Implementation details - Based on DFS)

\[ s, b, f, g, t \]
Algorithm: \emph{st}-ordering (example)
(Implementation details - Based on DFS)

\begin{itemize}
\item \(s, b, f, g, t\)
\end{itemize}
$st$-ordering: implementation

Algorithm: $st$-ordering (example)

(Implementation details - Based on DFS)

$s, b, f, g, h, t$
Algorithm: $st$-ordering (example)

(Implementation details - Based on DFS)

$s, b, f, g, h, t$
$st$-ordering: implementation

Algorithm: $st$-ordering (example)
(Implementation details - Based on DFS)
Algorithm: \( st \)-ordering (example)
(Implementation details - Based on DFS)
Algorithm: \textit{st}-ordering (example)
(Implementation details - Based on DFS)
Algorithm: \( st \)-ordering (example)

(Implementation details - Based on DFS)

\[
s, e, b, f, g, h, t
\]
**st-ordering: implementation**

**Algorithm: st-ordering (example)**
(Implementation details - Based on DFS)

```
s, e, b, a, f, g, h, t  
```
st-ordering: implementation

Algorithm: st-ordering (example)
(Implementation details - Based on DFS)
Algorithm: \( st \)-ordering (example)

(Implementation details - Based on DFS)

Diagram:

\[ s, e, b, a, f, g, h, t \]
**st-ordering: implementation**

**Algorithm: st-ordering (example)**
(Implementation details - Based on DFS)

\[ s, e, b, a, f, c, d, g, h, t \]
**Algorithm $st$-ordering**

**Data:** Undirected biconnected graph $G = (V, E)$, edge $\{s, t\} \in E$

**Result:** List $L$ of nodes representing an $st$-ordering of $G$

**$dfs$** (vertex $v$) begin

- $i \leftarrow i + 1$; $DFS[v] \leftarrow i$
- **while** there exists non-enumerated $e = \{v, w\}$ **do**
  - $DFS[e] \leftarrow DFS[v]$;
  - **if** $w$ not enumerated **then**
    - $CHILDEDGE[v] \leftarrow e$; $PARENT[w] \leftarrow v$
    - $dfs(w)$;
  - **else**
    - $\{w, x\} \leftarrow CHILDEDGE[w]$; $D[\{w, x\}] \leftarrow D[\{w, x\}] \cup \{e\}$
    - **if** $x \in L$ **then** process_ears($w \rightarrow x$);

**begin**

- initialize $L$ as $\{s, t\}$
- $DFS[s] \leftarrow 1$; $i \leftarrow 1$; $DFS[\{s, t\}] \leftarrow 1$; $CHILDEDGE[s] \leftarrow \{s, t\}$
- $dfs(t)$;
$st$-ordering: implementation

Function $process$ $ears$

```
process_ears(tree edge $w \to x$) begin
    foreach $v \leftrightarrow w \in D[w \to x]$ do
        $u \leftarrow v$;
        while $u \notin L$ do $u \leftarrow PARENT[u]$;
    $P \leftarrow (u \to v \to w)$;
    if $w \to x$ is oriented from $w$ to $x$ (resp. from $x$ to $w$) then
        orient $P$ from $w$ to $u$ (resp. from $u$ to $w$);
        paste the inner nodes of $P$ to $L$
        before (resp. after) $u$;
    foreach tree edge $w' \to x'$ of $P$ do $process_ears(w' \to x')$;
    $D[\{w, x\}] \leftarrow \emptyset$;
```
The described algorithm produces an \( st \)-ordering of a given biconnected graph \( G = (V, E) \) in \( O(E) \) time.
Discussion

- Today: incremental algorithm for orthogonal drawings, with worst-case guarantees. $2n + 4$ bends in total, which is almost optimal. Lower bound: $2n - 2$.

- Algorithms is simple, linear-time, works for non-planar graphs

- For planar graphs produces planar drawing

- Uses st-ordering

- Construction of st-ordering using ear-decomposition

(lower bound)
Discussion

- Today: incremental algorithm for orthogonal drawings, with worst-case guarantees. \(2n + 4\) bends in total, which is almost optimal. Lower bound: \(2n - 2\).
- Algorithms is simple, linear-time, works for non-planar graphs
- For planar graphs produces planar drawing
- Uses st-ordering
- Construction of st-ordering using ear-decomposition
- Next: algorithm based on network flow, that achieves minimum number of bends

lower bound