Visualization of Graphs

Lecture 5:
Orthogonal Layouts

Part I:
Topology – Shape – Metric

Jonathan Klawitter
Orthogonal Layout – Applications

ER diagram in OGDF
Orthogonal Layout – Applications

ER diagram in OGDF

UML diagram by Oracle
Orthogonal Layout – Applications

Organigram of HS Limburg

ER diagram in OGDF

UML diagram by Oracle
Orthogonal Layout – Applications

ER diagram in OGDF

Organigram of HS Limburg

Circuit diagram by Jeff Atwood

UML diagram by Oracle
Definition. A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if

- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.
Orthogonal Layout – Definition

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

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called \textbf{orthogonal} if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.

- Edges lie on grid $\Rightarrow$ bends lie on grid points
- Max degree of each vertex is at most 4
Orthogonal Layout – Definition

**Observations.**

- Edges lie on grid $\Rightarrow$ **bends** lie on grid points

**Definition.**

A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if

- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.

- Edges lie on grid $\Rightarrow$ **bends** lie on grid points
- Max degree of each vertex is at most 4
Orthogonal Layout – Definition

Observations.

- Edges lie on grid ⇒ bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.
Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid ⇒ bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise
Orthogonal Layout – Definition

Definition.
A drawing \( \Gamma \) of a graph \( G = (V, E) \) is called orthogonal if:
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid \( \Rightarrow \) bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise
**Orthogonal Layout – Definition**

**Observations.**
- Edges lie on grid ⇒ **bends** lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

**Definition.**
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

**Planarization.**
- Fix embedding
- Crossings become vertices
**Orthogonal Layout – Definition**

**Definition.**
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

**Observations.**
- Edges lie on grid $\Rightarrow$ **bends** lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

**Planarization.**
- Fix embedding

![Diagram of an orthogonal layout with examples of bends and vertex degrees]
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid $\Rightarrow$ **bends** lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

Planarization.
- Fix embedding
- Crossings become vertices
Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid $\Rightarrow$ **bends** lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

Planarization.
- Fix embedding
- Crossings become vertices
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid $\Rightarrow$ bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

Planarization.
- Fix embedding
- Crossings become vertices
Orthogonal Layout – Definition

**Observations.**
- Edges lie on grid $\Rightarrow$ bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

**Definition.**
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

**Planarization.**
- Fix embedding
- Crossings become vertices

**Aesthetic criteria.**
- Number of bends
- Length of edges
- Width, height, area
- Monotonicity of edges
- ...
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid $\Rightarrow$ bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

Planarization.
- Fix embedding
- Crossings become vertices

Aesthetic criteria.
- Number of bends
Orthogonal Layout – Definition

**Definition.**
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

**Observations.**
- Edges lie on grid $\Rightarrow$ **bends** lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

**Planarization.**
- Fix embedding
- Crossings become vertices

**Aesthetic criteria.**
- Number of bends
- Length of edges
Orthogonal Layout – Definition

Definition.
A drawing $\Gamma$ of a graph $G = (V, E)$ is called orthogonal if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.
- Edges lie on grid $\Rightarrow$ bends lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

Planarization.
- Fix embedding
- Crossings become vertices

Aesthetic criteria.
- Number of bends
- Length of edges
- Width, height, area
Orthogonal Layout – Definition

**Observations.**
- Edges lie on grid ⇒ **bends** lie on grid points
- Max degree of each vertex is at most 4
- Otherwise

**Definition.**
A drawing $\Gamma$ of a graph $G = (V, E)$ is called **orthogonal** if
- vertices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

**Planarization.**
- Fix embedding
- Crossings become vertices

**Aesthetic criteria.**
- Number of bends
- Length of edges
- Width, height, area
- Monotonicity of edges
- ...

---

**Diagram:**
A diagram showing an example of an orthogonal layout, with vertices and edges represented on a grid, illustrating the concept of bends and the orthogonality of edges.
Topology – Shape – Metrics

Three-step approach: [Tamassia 1987]
Topology – Shape – Metrics

Three-step approach: [Tamassia 1987]

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]
Topology – Shape – Metrics

Three-step approach:

\[ V = \{ v_1, v_2, v_3, v_4 \} \]
\[ E = \{ v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4 \} \]

[Tamassia 1987]
Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

combinatorial embedding/planarization

reduce crossings

[[Tamassia 1987]]
Topology – Shape – Metrics

Three-step approach:  

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

![Graph](image)

- **Topology**
- **Shape**
- **Metrics**

[Tamassia 1987]
Topology – Shape – Metrics

Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

- Combinatorial embedding/planarization
- Orthogonal representation
- Bend minimization

[Tamassia 1987]
Topology – Shape – Metrics

Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

- Combinatorial embedding/planarization
- Orthogonal representation
- Planar orthogonal drawing
- Bend minimization
- Reduce crossings

[Tamassia 1987]
Topology – Shape – Metrics

Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

[Tamassia 1987]
Topology – Shape – Metrics

Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

[Tamassia 1987]
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorically.
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$. 
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorically.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorically.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.
- Let $e$ be an edge with the face $f$ to the right.
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An **edge description** of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where

  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
  - $\alpha$ is angle $\in \{\pi/2, \pi, 3\pi/2, 2\pi\}$ between $e$ and next edge $e'$

\[ (e, 100, \pi) \]
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorically.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ (0 = right bend, 1 = left bend)
  - $\alpha$ is angle $\in \{\pi/2, \pi, 3\pi/2, 2\pi\}$ between $e$ and next edge $e'$
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ (0 = right bend, 1 = left bend)
  - $\alpha$ is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between $e$ and next edge $e'$.
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An **edge description** of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.
- Let $e$ be an edge with the face $f$ to the right.
  An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 = \text{right bend, } 1 = \text{left bend}$)
  - $\alpha$ is angle $\in \{\pi/2, \pi, 3\pi/2, 2\pi\}$ between $e$ and next edge $e'$

Idea.
Describe orthogonal drawing combinatorically.
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An **edge description** of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
    - $\delta$ is a sequence of $\{0, 1\}^*$ (0 = right bend, 1 = left bend)
    - $\alpha$ is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between $e$ and next edge $e'$
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.
- Let $e$ be an edge with the face $f$ to the right.
  An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ (0 = right bend, 1 = left bend)
  - $\alpha$ is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between $e$ and next edge $e'$
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorically.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right. An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
  - $\alpha$ is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between $e$ and next edge $e'$

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$. 

- Let $e$ be an edge with the face $f$ to the right. An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
  - $\alpha$ is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between $e$ and next edge $e'$
Orthogonal Representation

Idea.
Describe orthogonal drawing combinatorially.

Definitions.
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
  - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
  - $\alpha$ is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between $e$ and next edge $e'$
- A face representation $H(f)$ of $f$ is a clockwise ordered sequence of edge descriptions $(e, \delta, \alpha)$. 
Orthogonal Representation

**Idea.**
Describe orthogonal drawing combinatorially.

**Definitions.**
Let $G = (V, E)$ be a plane graph with faces $F$ and outer face $f_0$.

- Let $e$ be an edge with the face $f$ to the right.
  - An edge description of $e$ wrt $f$ is a triple $(e, \delta, \alpha)$ where
    - $\delta$ is a sequence of $\{0, 1\}^*$ ($0 =$ right bend, $1 =$ left bend)
    - $\alpha$ is angle $\in \{\pi/2, \pi, 3\pi/2, 2\pi\}$ between $e$ and next edge $e'$

- A face representation $H(f)$ of $f$ is a clockwise ordered sequence of edge descriptions $(e, \delta, \alpha)$.

- An orthogonal representation $H(G)$ of $G$ is defined as

\[ H(G) = \{H(f) \mid f \in F\}. \]
Orthogonal Representation – Example
Orthogonal Representation – Example
Orthogonal Representation – Example

\[ f_0, f_1, f_2, e_1, e_2, e_3, e_4, e_5, e_6 \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]

Combinatorial “drawing” of \( H(G) \)?
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \varnothing, \pi), (e_3, \varnothing, \pi), (e_2, \varnothing, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \varnothing, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \varnothing, \pi), (e_4, \varnothing, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[
H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2}))
\]

\[
H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi))
\]

\[
H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2}))
\]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[
H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2}))
\]

\[
H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi))
\]

\[
H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2}))
\]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]
\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]
\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[ H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2})) \]

\[ H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi)) \]

\[ H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2})) \]
Orthogonal Representation – Example

\[
H(f_0) = ((e_1, 11, \frac{\pi}{2}), (e_5, 111, \frac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \frac{\pi}{2}))
\]

\[
H(f_1) = ((e_1, 00, \frac{3\pi}{2}), (e_2, \emptyset, \frac{\pi}{2}), (e_6, 00, \pi))
\]

\[
H(f_2) = ((e_5, 000, \frac{\pi}{2}), (e_6, 11, \frac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \frac{\pi}{2}))
\]

Concrete coordinates are not fixed yet!
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u,v\}$ shared by faces $f$ and $g$ with $((u,v),\delta_1,\alpha_1) \in H(f)$ and $((v,u),\delta_2,\alpha_2) \in H(g)$, the sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta_0|$ (resp. $|\delta_1|$) be the number of zeros (resp. ones) in $\delta$ and $r = (e,\delta,\alpha)$. Let $C(r) := |\delta_0| - |\delta_1| + 2 - \alpha \cdot 2/\pi$.

For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

\[ f_0 \]
\[ f_1 \]
\[ f_2 \]
\[ e_1 \]
\[ e_2 \]
\[ e_3 \]
\[ e_4 \]
\[ e_5 \]
\[ e_6 \]
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.

Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot \frac{2}{\pi}$.

For each face $f$ it holds that:

$\sum_{r \in \text{H}(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise} \end{cases}$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

![Diagram](image-url)
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ the sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta_1|$ (resp. $|\delta_2|$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.

Let $C(r) := |\delta_1| - |\delta_2| + 2 - \alpha \cdot 2\pi/\pi$. For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise} \end{cases}.$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

$\begin{array}{c}
\text{Diagram}\n\end{array}$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$. 

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.

For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

\begin{itemize}
  \item $H(G)$ corresponds to $F$, $f_0$.
  \item For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.
  \item Let $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
  \item For each face $f$ it holds that:
    $$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$
  \item For each vertex $v$ the sum of incident angles is $2\pi$.
\end{itemize}
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. 
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C'(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$. 
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.
Let $C'(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u,v\}$ shared by faces $f$ and $g$ with $((u,v), \delta_1, \alpha_1) \in H(f)$ and $((v,u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.

For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}$$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.

For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge ${u, v}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.
Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2 / \pi$.
For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

$C(e_3) = 0 - 0 + 2 - =$

$C(e_4) = - + 2 - =$

$C(e_5) = - + 2 - =$

$C(e_6) = - + 2 - =$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.

For each face $f$, it holds that:

$$
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
$$

$C(e_3) = 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = $ 

$C(e_4) = - + 2 - = $ 

$C(e_5) = - + 2 - = $ 

$C(e_6) = - + 2 - = $
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.

Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.

For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

$$C(e_3) = 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = 0$$
$$C(e_4) = - + 2 - =$$
$$C(e_5) = - + 2 - =$$
$$C(e_6) = - + 2 - =$$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.
Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

\[
\begin{align*}
C(e_3) &= 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = 0 \\
C(e_4) &= 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = 1 \\
C(e_5) &= - + 2 - \\
C(e_6) &= - + 2 -
\end{align*}
\]
Correctness of an Orthogonal Representation

(H1) $H(G')$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.
Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
For each face $f$ it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
\]
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u,v\}$ shared by faces $f$ and $g$ with $((u,v),\delta_1,\alpha_1) \in H(f)$ and $((v,u),\delta_2,\alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$. For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise} \end{cases}$$

$$C(e_3) = 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = 0$$

$$C(e_4) = 0 - 0 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = 1$$

$$C(e_5) = 3 - 0 + 2 - = $$. 

$$C(e_6) = - + 2 - = $$
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u,v\}$ shared by faces $f$ and $g$ with $((u,v),\delta_1,\alpha_1) \in H(f)$ and $((v,u),\delta_2,\alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.
Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
For each face $f$ it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
\]
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u,v\}$ shared by faces $f$ and $g$ with
$((u,v),\delta_1,\alpha_1) \in H(f)$ and $((v,u),\delta_2,\alpha_2) \in H(g)$
sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros
(resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.  
Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
For each face $f$ it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
\]
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.

For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

\[C(e_3) = 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = 0\]
\[C(e_4) = 0 - 0 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = 1\]
\[C(e_5) = 3 - 0 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = 4\]
\[C(e_6) = 0 - 2 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = -1\]
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$. Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$. For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

\[
C(e_3) = 0 - 0 + 2 - \pi \cdot \frac{2}{\pi} = 0 \\
C(e_4) = 0 - 0 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = 1 \\
C(e_5) = 3 - 0 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = 4 \\
C(e_6) = 0 - 2 + 2 - \frac{\pi}{2} \cdot \frac{2}{\pi} = -1
\]
Correctness of an Orthogonal Representation

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$ with $((u, v), \delta_1, \alpha_1) \in H(f)$ and $((v, u), \delta_2, \alpha_2) \in H(g)$ sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) Let $|\delta|_0$ (resp. $|\delta|_1$) be the number of zeros (resp. ones) in $\delta$ and $r = (e, \delta, \alpha)$.
Let $C(r) := |\delta|_0 - |\delta|_1 + 2 - \alpha \cdot 2/\pi$.
For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.
Visualization of Graphs

Lecture 5:
Orthogonal Layouts

Part III:
Bend Minimization

Jonathan Klawitter
Reminder: $s$-$t$-Flow Networks

Flow network $(G = (V, E); S, T; u)$ with
- directed graph $G = (V, E)$
- sources $S \subseteq V$, sinks $T \subseteq V$
- edge capacity $u: E \rightarrow \mathbb{R}_0^+$

A function $X: E \rightarrow \mathbb{R}_0^+$ is called $S$-$T$-flow, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i,j)\in E} X(i,j) - \sum_{(j,i)\in E} X(j,i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A maximum $S$-$T$-flow is an $S$-$T$-flow where $\sum_{(i,j)\in E,i\in S} X(i,j)$ is maximized.
Reminder: \(s-t\)-Flow Networks

**Flow network** \((G = (V, E); s, t, u)\) with

- directed graph \(G = (V, E)\)
- source \(s \in V\), sink \(t \in V\)
- edge capacity \(u: E \to \mathbb{R}_0^+\)

A function \(X: E \to \mathbb{R}_0^+\) is called **S-T-flow**, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A **maximum** S-T-flow is an S-T-flow where \(\sum_{(i, j) \in E, i \in S} X(i, j)\) is maximized.
Reminder: \textit{s-t-Flow Networks}

**Flow network** \((G = (V, E); s, t; u)\) with
- directed graph \(G = (V, E)\)
- source \(s \in V\), sink \(t \in V\)
- edge capacity \(u: E \to \mathbb{R}_0^+\)

A function \(X: E \to \mathbb{R}_0^+\) is called \textit{s-t-flow}, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i,j) \in E} X(i, j) - \sum_{(j,i) \in E} X(j,i) = 0 \quad \forall i \in V \setminus \{s, t\}
\]

A \textbf{maximum} \textit{S-T-flow} is an \textit{S-T-flow} where \(\sum_{(i,j) \in E, i \in S} X(i, j)\) is maximized.
Reminder: $s$-$t$-Flow Networks

**Flow network** $(G = (V, E); s, t; u)$ with
- directed graph $G = (V, E)$
- source $s \in V$, sink $t \in V$
- edge capacity $u : E \to \mathbb{R}_0^+$

A function $X : E \to \mathbb{R}_0^+$ is called $s$-$t$-flow, if:

$$0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E$$

$$\sum_{(i,j) \in E} X(i, j) - \sum_{(j,i) \in E} X(j,i) = 0 \quad \forall i \in V \setminus \{s,t\}$$

A maximum $s$-$t$-flow is an $s$-$t$-flow where $\sum_{(s,j) \in E} X(s, j)$ is maximized.
Reminder: \( s-t \)-Flow Networks

**Flow network** \((G = (V, E); s, t; u)\) with
- directed graph \( G = (V, E) \)
- **source** \( s \in V \), **sink** \( t \in V \)
- edge **capacity** \( u: E \rightarrow \mathbb{R}_0^+ \)

A function \( X: E \rightarrow \mathbb{R}_0^+ \) is called **\( s-t \)-flow**, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]
\[
\sum_{(i,j) \in E} X(i, j) - \sum_{(j,i) \in E} X(j,i) = 0 \quad \forall i \in V \setminus \{s, t\}
\]

A maximum **\( s-t \)-flow** is an \( s-t \)-flow where \( \sum_{(s,j) \in E} X(s, j) \) is maximized.
Reminder: \(s-t\)-Flow Networks

**Flow network** \((G = (V, E); s, t, u)\) with
- directed graph \(G = (V, E)\)
- source \(s \in V\), sink \(t \in V\)
- edge capacity \(u: E \to \mathbb{R}^+_0\)

A function \(X: E \to \mathbb{R}^+_0\) is called \(s-t\)-flow, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]
\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus \{s, t\}
\]

A maximum \(s-t\)-flow is an \(s-t\)-flow where \(\sum_{(s, j) \in E} X(s, j)\) is maximized.
Reminder: \( s-t \)-Flow Networks

Flow network \((G = (V, E); s, t; u)\) with
- directed graph \(G = (V, E)\)
- source \(s \in V\), sink \(t \in V\)
- edge capacity \(u: E \to \mathbb{R}_0^+\)

A function \(X: E \to \mathbb{R}_0^+\) is called \(s-t\)-flow, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus \{s, t\}
\]

A maximum \(s-t\)-flow is an \(s-t\)-flow where \(\sum_{(s, j) \in E} X(s, j)\) is maximized.
Reminder: $s$-$t$-Flow Networks

**Flow network** $(G = (V, E); s, t; u)$ with
- directed graph $G = (V, E)$
- source $s \in V$, sink $t \in V$
- edge capacity $u: E \to \mathbb{R}_0^+$

A function $X: E \to \mathbb{R}_0^+$ is called *$s$-$t$-flow*, if:

$$0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E$$

$$\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus \{s, t\}$$

A **maximum** $s$-$t$-flow is an $s$-$t$-flow where $\sum_{(s, j) \in E} X(s, j)$ is maximized.
Reminder: $s$-$t$-Flow Networks

A flow network $(G = (V, E); s, t; u)$ with
- directed graph $G = (V, E)$
- source $s \in V$, sink $t \in V$
- edge capacity $u : E \rightarrow \mathbb{R}_0^+$

A function $X : E \rightarrow \mathbb{R}_0^+$ is called $s$-$t$-flow, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]
\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus \{s, t\}
\]

A maximum $s$-$t$-flow is an $s$-$t$-flow where $\sum_{(s, j) \in E} X(s, j)$ is maximized.
General Flow Network

**Flow network** \((G = (V, E); S, T; u)\) with
- directed graph \(G = (V, E)\)
- sources \(S \subseteq V\), sinks \(T \subseteq V\)
- edge capacity \(u: E \rightarrow \mathbb{R}^+_0\)

A function \(X: E \rightarrow \mathbb{R}^+_0\) is called **S-T-flow**, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]
\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A **maximum** S-T-flow is an S-T-flow where \(\sum_{(i, j) \in E, i \in S} X(i, j)\) is maximized.
General Flow Network

Flow network \((G = (V, E); S, T; \ell; u)\) with
- directed graph \(G = (V, E)\)
- sources \(S \subseteq V\), sinks \(T \subseteq V\)
- edge capacity \(u: E \rightarrow \mathbb{R}_0^+\)

A function \(X: E \rightarrow \mathbb{R}_0^+\) is called \(S-T\)-flow, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A maximum \(S-T\)-flow is an \(S-T\)-flow where \(\sum_{(i,j) \in E, i \in S} X(i, j)\) is maximized.
General Flow Network

**Flow network** \((G = (V, E); S, T; ℓ; u)\) with

- directed graph \(G = (V, E)\)
- sources \(S \subseteq V\), sinks \(T \subseteq V\)
- edge *lower bound* \(ℓ : E \to \mathbb{R}_0^+\)
- edge *capacity* \(u : E \to \mathbb{R}_0^+\)

A function \(X : E \to \mathbb{R}_0^+\) is called **S-T-flow**, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A **maximum** S-T-flow is an S-T-flow where \(\sum_{(i, j) \in E, i \in S} X(i, j)\) is maximized.
A function $X : E \to \mathbb{R}^+_0$ is called a $S-T$-flow, if:

\[
0 \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A maximum $S-T$-flow is an $S-T$-flow where $\sum_{(i, j) \in E, i \in S} X(i, j)$ is maximized.
A function $X : E \to \mathbb{R}_0^+$ is called $S$-$T$-flow, if:

$$
\ell(i, j) \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
$$

$$
\sum_{(i,j) \in E} X(i,j) - \sum_{(j,i) \in E} X(j,i) = 0 \quad \forall i \in V \setminus (S \cup T)
$$

A maximum $S$-$T$-flow is an $S$-$T$-flow where $\sum_{(i,j) \in E, i \in S} X(i,j)$ is maximized.
**General Flow Network**

**Flow network** \((G = (V, E); b; \ell; u)\) with
- directed graph \(G = (V, E)\)
- node *production/consumption* \(b: V \to \mathbb{R}\) with \(\sum_{i \in V} b(i) = 0\)
- edge *lower bound* \(\ell: E \to \mathbb{R}^+_0\)
- edge *capacity* \(u: E \to \mathbb{R}^+_0\)

A function \(X: E \to \mathbb{R}^+_0\) is called *S-T-flow*, if:

\[
\ell(i, j) \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E \\
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = 0 \quad \forall i \in V \setminus (S \cup T)
\]

A **maximum** *S-T-flow* is an *S-T-flow* where \(\sum_{(i, j) \in E, i \in S} X(i, j)\) is maximized.
General Flow Network

Flow network \((G = (V, E); b; \ell; u)\) with
- directed graph \(G = (V, E)\)
- node production/consumption \(b: V \rightarrow \mathbb{R}\) with \(\sum_{i \in V} b(i) = 0\)
- edge lower bound \(\ell: E \rightarrow \mathbb{R}^+_0\)
- edge capacity \(u: E \rightarrow \mathbb{R}^+_0\)

A function \(X: E \rightarrow \mathbb{R}^+_0\) is called valid flow, if:

\[
\ell(i, j) \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i,j) \in E} X(i,j) - \sum_{(j,i) \in E} X(j,i) = b(i) \quad \forall i \in V
\]

A maximum \(S-T\)-flow is an \(S-T\)-flow where \(\sum_{(i,j) \in E, i \in S} X(i,j)\) is maximized.
General Flow Network

**Flow network** \((G = (V, E); b; \ell; u)\) with
- directed graph \(G = (V, E)\)
- node *production/consumption* \(b: V \to \mathbb{R}\) with \(\sum_{i \in V} b(i) = 0\)
- edge *lower bound* \(\ell : E \to \mathbb{R}_0^+\)
- edge *capacity* \(u : E \to \mathbb{R}_0^+\)

A function \(X : E \to \mathbb{R}_0^+\) is called **valid flow**, if:

\[
\ell(i, j) \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = b(i) \quad \forall i \in V
\]

- **Cost function** \(\text{cost}: E \to \mathbb{R}_0^+\)

A **maximum** \(S-T\)-flow is an \(S-T\)-flow where \(\sum_{(i, j) \in E, i \in S} X(i, j)\) is maximized.
General Flow Network

**Flow network** \((G = (V, E); b; \ell; u)\) with

- directed graph \(G = (V, E)\)
- node *production/consumption* \(b : V \to \mathbb{R}\) with \(\sum_{i \in V} b(i) = 0\)
- edge *lower bound* \(\ell : E \to \mathbb{R}_0^+\)
- edge *capacity* \(u : E \to \mathbb{R}_0^+\)

A function \(X : E \to \mathbb{R}_0^+\) is called **valid flow**, if:

\[
\ell(i, j) \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = b(i) \quad \forall i \in V
\]

**Cost function** \(\text{cost} : E \to \mathbb{R}_0^+\) and \(\text{cost}(X) := \sum_{(i, j) \in E} \text{cost}(i, j) \cdot X(i, j)\)

A **maximum** \(S-T\)-flow is an \(S-T\)-flow where \(\sum_{(i, j) \in E, i \in S} X(i, j)\) is maximized.
General Flow Network

Flow network \((G = (V, E); b; \ell; u)\) with
- directed graph \(G = (V, E)\)
- node production/consumption \(b: V \rightarrow \mathbb{R}\) with \(\sum_{i \in V} b(i) = 0\)
- edge lower bound \(\ell: E \rightarrow \mathbb{R}^+_0\)
- edge capacity \(u: E \rightarrow \mathbb{R}^+_0\)

A function \(X: E \rightarrow \mathbb{R}^+_0\) is called valid flow, if:

\[
\ell(i, j) \leq X(i, j) \leq u(i, j) \quad \forall (i, j) \in E
\]

\[
\sum_{(i, j) \in E} X(i, j) - \sum_{(j, i) \in E} X(j, i) = b(i) \quad \forall i \in V
\]

- Cost function \(\text{cost}: E \rightarrow \mathbb{R}^+_0\) and \(\text{cost}(X) := \sum_{(i, j) \in E} \text{cost}(i, j) \cdot X(i, j)\)

A minimum cost flow is a valid flow where \(\text{cost}(X)\) is minimized.
# General Flow Network – Algorithms

## Polynomial Algorithms

<table>
<thead>
<tr>
<th>#</th>
<th>Due to</th>
<th>Year</th>
<th>Running Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Edmonds and Karp</td>
<td>1972</td>
<td>$O((n + m) \log U S(n, m, nC))$</td>
</tr>
<tr>
<td>2</td>
<td>Rock</td>
<td>1980</td>
<td>$O((n + m) \log U S(n, m, nC))$</td>
</tr>
<tr>
<td>3</td>
<td>Rock</td>
<td>1980</td>
<td>$O(n \log C M(n, m, U))$</td>
</tr>
<tr>
<td>4</td>
<td>Bland and Jensen</td>
<td>1985</td>
<td>$O(m \log C M(n, m, U))$</td>
</tr>
<tr>
<td>5</td>
<td>Goldberg and Tarjan</td>
<td>1987</td>
<td>$O(nm \log (n^2/m) \log (nC))$</td>
</tr>
<tr>
<td>6</td>
<td>Goldberg and Tarjan</td>
<td>1988</td>
<td>$O(nm \log n \log (nC))$</td>
</tr>
<tr>
<td>7</td>
<td>Ahuja, Goldberg, Orlin and Tarjan</td>
<td>1988</td>
<td>$O(nm \log \log U \log (nC))$</td>
</tr>
</tbody>
</table>

## Strongly Polynomial Algorithms

<table>
<thead>
<tr>
<th>#</th>
<th>Due to</th>
<th>Year</th>
<th>Running Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tardos</td>
<td>1985</td>
<td>$O(m^4)$</td>
</tr>
<tr>
<td>2</td>
<td>Orlin</td>
<td>1984</td>
<td>$O((n + m)^2 \log n S(n, m))$</td>
</tr>
<tr>
<td>3</td>
<td>Fujishige</td>
<td>1986</td>
<td>$O((n + m)^2 \log n S(n, m))$</td>
</tr>
<tr>
<td>4</td>
<td>Galil and Tardos</td>
<td>1986</td>
<td>$O(n^2 \log n S(n, m))$</td>
</tr>
<tr>
<td>5</td>
<td>Goldberg and Tarjan</td>
<td>1987</td>
<td>$O(nm^2 \log n \log (n^2/m))$</td>
</tr>
<tr>
<td>6</td>
<td>Goldberg and Tarjan</td>
<td>1988</td>
<td>$O(nm^2 \log^2 n)$</td>
</tr>
<tr>
<td>7</td>
<td>Orlin (this paper)</td>
<td>1988</td>
<td>$O((n + m) \log n S(n, m))$</td>
</tr>
</tbody>
</table>

\[
S(n, m) = O(m + n \log n) \\
S(n, m, C) = O\left( m + \sqrt{n \log C} \right) \\
\left( m \log \log C \right) \\
M(n, m) = O\left( \min(nm + n^{2+\epsilon}, nm \log n) \right) \\
where \epsilon is any fixed constant. \\
M(n, m, U) = O(nm \log \left( \frac{n}{m} \sqrt{\log U + 2} \right)) \\
\]

Fredman and Tarjan [1984]  
Ahuja, Mehlhorn, Orlin and Tarjan [1990]  
Van Emde Boas, Kaas and Zijlstra [1977]  
King, Rao, and Tarjan [1991]  
Ahuja, Orlin and Tarjan [1989]
Theorem. [Orlin 1991]

The minimum cost flow problem can be solved in $O(n^2 \log^2 n + m^2 \log n)$ time.
Theorem. [Cornelsen & Karrenbauer 2011]
The minimum cost flow problem for planar graphs with bounded costs and faze sizes can be solved in $O(n^{3/2})$ time.

Theorem. [Orlin 1991]
The minimum cost flow problem can be solved in $O(n^2 \log^2 n + m^2 \log n)$ time.
Topology – Shape – Metrics

Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

- combinatorial embedding/planarization
- planar orthogonal drawing
- area minimization
- bend minimization
- orthogonal representation
- reduce crossings

[Tamassia 1987]
Bend Minimization with Given Embedding

**Geometric bend minimization.**

Given:

Find:
Bend Minimization with Given Embedding

**Geometric bend minimization.**

Given:  ■ Plane graph $G = (V, E)$ with maximum degree 4

Find:
Bend Minimization with Given Embedding

**Geometric bend minimization.**

Given:
- Plane graph \( G = (V, E) \) with maximum degree 4
- Combinatorial embedding \( F \) and outer face \( f_0 \)

Find:
**Bend Minimization with Given Embedding**

<table>
<thead>
<tr>
<th>Geometric bend minimization.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Given:</strong></td>
</tr>
<tr>
<td>- Plane graph $G = (V, E)$ with maximum degree 4</td>
</tr>
<tr>
<td>- Combinatorial embedding $F$ and outer face $f_0$</td>
</tr>
<tr>
<td><strong>Find:</strong></td>
</tr>
<tr>
<td>Orthogonal drawing with minimum number of bends that preserves the embedding.</td>
</tr>
</tbody>
</table>
Bend Minimization with Given Embedding

**Geometric bend minimization.**

Given:
- Plane graph $G = (V, E)$ with maximum degree 4
- Combinatorial embedding $F$ and outer face $f_0$

Find:
Orthogonal drawing with minimum number of bends that preserves the embedding.

Compare with the following variation.

**Combinatorial bend minimization.**

Given:

Find:
Bend Minimization with Given Embedding

**Geometric bend minimization.**

Given: ▪ Plane graph $G = (V, E)$ with maximum degree 4  
▪ Combinatorial embedding $F$ and outer face $f_0$

Find: Orthogonal drawing with minimum number of bends that preserves the embedding.

Compare with the following variation.

**Combinatorial bend minimization.**

Given: ▪ Plane graph $G = (V, E)$ with maximum degree 4  
▪ Combinatorial embedding $F$ and outer face $f_0$

Find:
Bend Minimization with Given Embedding

Geometric bend minimization.
Given:  ■ Plane graph $G = (V,E)$ with maximum degree 4
        ■ Combinatorial embedding $F$ and outer face $f_0$
Find:  Orthogonal drawing with minimum number of bends that preserves the embedding.

Compare with the following variation.

Combinatorial bend minimization.
Given:  ■ Plane graph $G = (V,E)$ with maximum degree 4
        ■ Combinatorial embedding $F$ and outer face $f_0$
Find:  **Orthogonal representation** $H(G)$ with minimum number of bends that preserves the embedding.
Combinatorial Bend Minimization

**Combinatorial bend minimization.**

**Given:**
- Plane graph $G = (V, E)$ with maximum degree 4
- Combinatorial embedding $F$ and outer face $f_0$

**Find:** Orthogonal representation $H(G)$ with minimum number of bends that preserves the embedding
Combinatorial Bend Minimization

**Combinatorial bend minimization.**

**Given:**
- Plane graph \( G = (V, E) \) with maximum degree 4
- Combinatorial embedding \( F \) and outer face \( f_0 \)

**Find:** **Orthogonal representation** \( H(G) \) with minimum number of bends that preserves the embedding

**Idea.**
Formulate as a network flow problem:
Combinatorial Bend Minimization

Combinatorial bend minimization.
Given: ■ Plane graph \( G = (V, E) \) with maximum degree 4
■ Combinatorial embedding \( F \) and outer face \( f_0 \)
Find: Orthogonal representation \( H(G) \) with minimum number of bends that preserves the embedding

Idea.
Formulate as a network flow problem:
■ a unit of flow = \( \angle \frac{\pi}{2} \)
Combinatorial Bend Minimization

Combinatorial bend minimization.

Given:
- Plane graph $G = (V, E)$ with maximum degree 4
- Combinatorial embedding $F$ and outer face $f_0$

Find: **Orthogonal representation** $H(G)$ with minimum number of bends that preserves the embedding

Idea.
Formulate as a network flow problem:
- a unit of flow $= \angle \frac{\pi}{2}$
- vertices $\rightarrow$ faces ($\# \angle \frac{\pi}{2}$ per face)
Combinatorial Bend Minimization

Combinatorial bend minimization.

Given:
- Plane graph \( G = (V, E) \) with maximum degree 4
- Combinatorial embedding \( F \) and outer face \( f_0 \)

Find: Orthogonal representation \( H(G) \) with minimum number of bends that preserves the embedding

Idea.
Formulate as a network flow problem:
- a unit of flow = \( \angle \frac{\pi}{2} \)
- vertices \( \xrightarrow{\angle} \) faces (\# \( \angle \frac{\pi}{2} \) per face)
- faces \( \xrightarrow{\angle} \) neighbouring faces (\# bends toward the neighbour)
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u,v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
\]

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 
(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 
Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

1. $H(G)$ corresponds to $F, f_0$.
2. For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.
3. For each face $f$ it holds that:
   
   $\sum_{r \in H(f)} C(r) = \begin{cases} 
   -4 & \text{if } f = f_0 \\
   +4 & \text{otherwise.} 
   \end{cases}$
4. For each vertex $v$ the sum of incident angles is $2\pi$. 
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$: 
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$. 

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$. 

(H3) For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\}$
Flow Network for Bend Minimization

Define flow network $N(G) = ((V ∪ F, E); b; ℓ; u; \text{cost})$:

$$E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\}$$

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 
Flow Network for Bend Minimization

Define flow network \( N(G) = ((V \cup F, E); b; \ell; u; \text{cost}) \):

\[
E = \{(v, f)_{e e'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\}
\]

(H1) \( H(G) \) corresponds to \( F, f_0 \).

(H2) For each edge \( \{u, v\} \) shared by faces \( f \) and \( g \), sequence \( \delta_1 \) is reversed and inverted \( \delta_2 \).

(H3) For each face \( f \) it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.} 
\end{cases}
\]

(H4) For each vertex \( v \) the sum of incident angles is \( 2\pi \).
Flow Network for Bend Minimization

Define flow network \( N(G) = ((V \cup F, E); b; \ell; u; \text{cost}) \):

\[ E = \{ (v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f \} \]

(H1) \( H(G) \) corresponds to \( F, f_0 \).

(H2) For each edge \( \{u, v\} \) shared by faces \( f, g \), sequence \( \delta_1 \) is reversed and inverted \( \delta_2 \).

(H3) For each face \( f \) it holds that:

\[ \sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases} \]

(H4) For each vertex \( v \) the sum of incident angles is \( 2\pi \).
Flow Network for Bend Minimization

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v,f)_{ee'} \in V \times F | v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f,g)_e \in F \times F | f, g \text{ have common edge } e\}$

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

Directed multigraph!
Flow Network for Bend Minimization

Define flow network $N(G) = ((V ∪ F, E); b; ℓ; u; \text{cost})$:

- $E = \{(v, f) ee' \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} ∪ \{(f, g) e \in F \times F \mid f, g \text{ have common edge } e\}$

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $δ_1$ is reversed and inverted $δ_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Directed multigraph!
Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_{e} \in F \times F \mid f, g \text{ have common edge } e\}$

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Directed multigraph!
Flow Network for Bend Minimization

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$
- $b(v) = 4 \quad \forall v \in V$

(H1) $H(G)$ corresponds to $F, f_0$.
(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.
(H3) For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$
(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$

- $b(v) = 4 \ \forall v \in V$
Flow Network for Bend Minimization

(H1) \( H(G) \) corresponds to \( F, f_0 \).

(H2) For each edge \( \{u, v\} \) shared by faces \( f \) and \( g \), sequence \( \delta_1 \) is reversed and inverted \( \delta_2 \).

(H3) For each face \( f \) it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
\]

(H4) For each vertex \( v \) the sum of incident angles is \( 2\pi \).

Define flow network \( N(G) = ((V \cup F, E); b; \ell; u; \text{cost}) \):

- \( E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_{e} \in F \times F \mid f, g \text{ have common edge } e\} \)
- \( b(v) = 4 \quad \forall v \in V \)
- \( b(f) = \)

\[
\begin{array}{c}
2 \\
\end{array}
\begin{array}{c}
1 \\
1 \\
\end{array}
\begin{array}{c}
& 1 \\
\end{array}
\begin{array}{c}
& \\
\end{array}
\]

2
Flow Network for Bend Minimization

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise.} \end{cases}$

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

\[ \sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise.} \end{cases} \]

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

\[ H(G) \text{ corresponds to } F, f_0. \]

\[ \text{For each edge } \{u, v\} \text{ shared by faces } f \text{ and } g, \text{ sequence } \delta_1 \text{ is reversed and inverted } \delta_2. \]

\[ \text{For each face } f \text{ it holds that: } \sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise.} \end{cases} \]

\[ \text{For each vertex } v \text{ the sum of incident angles is } 2\pi. \]
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_{e} \in F \times F \mid f, g \text{ have common edge } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) =$

![Diagram of a flow network](image)
Flow Network for Bend Minimization

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases}$

(H1) $H(G)$ corresponds to $F, f_0$.
(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.
(H3) For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise}. \end{cases}$$
(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

\[
\begin{array}{c|c|c|c|c}
2 & 1 & & & \\
1 & 2 & 1 & & \\
& & & & -6 \\
& 1 & 1 & & \\
\end{array}
\]
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0, \\
+4 & \text{otherwise.}
\end{cases}
\]

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b, \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$

- $b(v) = 4 \quad \forall v \in V$

- $b(f) = -2 \deg_G(f) + \begin{cases} 
-4 & \text{if } f = f_0, \\
+4 & \text{otherwise.}
\end{cases}$

\[
\Rightarrow \sum_w b(w) = 0
\]
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F$, $f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b, \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases}$

$$\sum_w b(w) = 0 \quad \text{(Euler)}$$
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0, \\
+4 & \text{otherwise.} 
\end{cases}
\]

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_{ee} \in F \times F \mid f, g \text{ have common edge } e\}$

- $b(v) = 4 \quad \forall v \in V$

- $b(f) = -2 \deg_G(f) + \begin{cases} 
-4 & \text{if } f = f_0, \\
+4 & \text{otherwise} 
\end{cases} \Rightarrow \sum_w b(w) = 0 \quad \text{(Euler)}$

$\forall (v, f) \in E, v \in V, f \in F$
Flow Network for Bend Minimization

Define flow network \( N(G) = ((V \cup F, E); b; \ell; u; \text{cost}) \):

- \( E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\} \)

- \( b(v) = 4 \quad \forall v \in V \)

- \( b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases} \Rightarrow \sum_w b(w) = 0 \quad \text{(Euler)} \)

∀(v, f) ∈ E, v ∈ V, f ∈ F \quad \ell(v, f) := \leq X(v, f) \leq =: u(v, f) \\
\text{cost}(v, f) =
Flow Network for Bend Minimization

(H1) $H(G)$ corresponds to $F, f_0$.

(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.

(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$.

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost}):$

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$

- $b(v) = 4$ $\forall v \in V$

- $b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases} \Rightarrow \sum_w b(w) = 0$ (Euler)

$\forall(v, f) \in E, v \in V, f \in F$ $\ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)$

$\text{cost}(v, f) =$

[Diagram showing flow network with nodes and edges labeled]
Define flow network $N(G) = (((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_{e} \in F \times F \mid f, g \text{ have common edge } e\}$

- $b(v) = 4 \quad \forall v \in V$

- $b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases}$

- $\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases} \Rightarrow \sum_{w} b(w) = 0$ (Euler)

- $\forall (v, f) \in E, v \in V, f \in F \quad \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)$

- $\text{cost}(v, f) = 0$

- (H1) $H(G)$ corresponds to $F, f_0$.
- (H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.
- (H3) For each face $f$ it holds that:
  $\sum_{r \in H(f)} C(r) \leq 0$

- (H4) For each vertex $v$ the sum of incident angles is $2\pi$. 
Flow Network for Bend Minimization

- \( H(G) \) corresponds to \( F, f_0 \).
- For each edge \( \{u, v\} \) shared by faces \( f \) and \( g \), sequence \( \delta_1 \) is reversed and inverted \( \delta_2 \).
- For each face \( f \) it holds that:
  \[
  \sum_{r \in H(f)} C(r) = \begin{cases} 
  -4 & \text{if } f = f_0, \\
  +4 & \text{otherwise}.
  \end{cases}
  \]
- For each vertex \( v \) the sum of incident angles is \( 2\pi \).

Define flow network \( N(G) = ((V \cup F, E); b; \ell; u; \text{cost}) : \)

- \( E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between } e, e' \text{ of } \partial f\} \cup \{(f, g)_{e} \in F \times F \mid f, g \text{ have common edge } e\} \)
- \( b(v) = 4 \ \forall v \in V \)
- \( b(f) = -2 \deg_G(f) + \begin{cases} 
  -4 & \text{if } f = f_0, \\
  +4 & \text{otherwise}
  \end{cases} \Rightarrow \sum_w b(w) = 0 \) (Euler)

\[
\forall (v, f) \in E, v \in V, f \in F \quad \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)
\]

\[
\forall (f, g) \in E, f, g \in F \quad \ell(f, g) := \leq X(f, g) \leq =: u(f, g)
\]
Flow Network for Bend Minimization

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \ | \ v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \ | \ f, g \text{ have common edge } e\}$
- $b(v) = 4 \ \forall v \in V$
- $b(f) = -2 \text{ deg}_G(f) + \begin{cases} -4 \text{ if } f = f_0, \\ +4 \text{ otherwise} \end{cases}$
- $\forall (v, f) \in E, v \in V, f \in F \quad \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)$
- $\forall (f, g) \in E, f, g \in F \quad \ell(f, g) := 0 \leq X(f, g) \leq \infty =: u(f, g)$
- $\text{cost}(f, g) = 1$

(H1) $H(G)$ corresponds to $F, f_0$.
(H2) For each edge $\{u, v\}$ shared by faces $f$ and $g$, sequence $\delta_1$ is reversed and inverted $\delta_2$.
(H3) For each face $f$ it holds that:

$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 \text{ if } f = f_0, \\ +4 \text{ otherwise} \end{cases}$$

(H4) For each vertex $v$ the sum of incident angles is $2\pi$. 

\[\sum_w b(w) = 0 \quad \text{(Euler)}\]
Flow Network for Bend Minimization

Define flow network \( N(G) = ((V \cup F, E); b; \ell; u; \text{cost}) \):

- \( E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_{e} \in F \times F \mid f, g \text{ have common edge } e\} \)

- \( b(v) = 4 \quad \forall v \in V \)

- \( b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases} \Rightarrow \sum_w b(w) = 0 \) (Euler)

\((\text{H1})\) \( H(G) \) corresponds to \( F, f_0 \).

\((\text{H2})\) For each edge \( \{u, v\} \) shared by faces \( f \) and \( g \), sequence \( \delta_1 \) is reversed and inverted \( \delta_2 \).

\((\text{H3})\) For each face \( f \) it holds that:

\[ \sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases} \]

\((\text{H4})\) For each vertex \( v \) the sum of incident angles is \( 2\pi \).

We model only the number of bends. Why is it enough?
Flow Network for Bend Minimization

Define flow network $N(G) = ((V \cup F, E); b; \ell; u; \text{cost})$:

- $E = \{(v, f)_{ee'} \in V \times F \mid v \text{ between edges } e, e' \text{ of } \partial f\} \cup \{(f, g)_e \in F \times F \mid f, g \text{ have common edge } e\}$

- $b(v) = 4 \quad \forall v \in V$

- $b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases}$

$\Rightarrow \sum_w b(w) = 0$ (Euler)

We model only the number of bends. Why is it enough?
Flow Network Example

\[ f_0 \]

\[ f_1 \]

\[ f_2 \]

\[ e_1 \]

\[ e_2 \]

\[ e_3 \]

\[ e_4 \]

\[ e_5 \]

\[ e_6 \]
Flow Network Example

Legend

$V$  ○
$F$  ○
Flow Network Example

Legend

$V$  
$F$  

$V \times F \supseteq \ell/u/cost$  

$1/4/0$
Flow Network Example

Legend
- $V$: Black circles
- $F$: Red circles
- $\ell/u/cost$

$V \times F \supseteq 1/4/0$
Flow Network Example

Legend

- $V$ (white dots)
- $F$ (red dots)
- $\ell/u/cost$

$V \times F \supseteq \frac{1}{4}/0$
Flow Network Example

Legend
- $V$: black circles
- $F$: red circles

Legend
- $\ell/u/cost$:
- $V \times F \supseteq \frac{1}{4}/0$
Flow Network Example

Legend

- $V$ (black circle) represents vertices.
- $F$ (red circle) represents a flow.

- The symbols $\ell/u/cost$ indicate the flow parameters.

- $V \times F \supseteq \begin{array}{c} 1/4/0 \end{array}$
- $F \times F \supseteq \begin{array}{c} 0/\infty/1 \end{array}$
Flow Network Example

Legend

- $V$ (nodes)
- $F$ (flows)

Properties:
- $V \times F \supseteq 1/4/0$
- $F \times F \supseteq 0/\infty/1$
Flow Network Example

Legend
- $V$ (black circle)
- $F$ (red circle)
- $\ell/u/cost$

$V \times F \supseteq \begin{array}{c} 1/4/0 \\ \end{array}$

$F \times F \supseteq \begin{array}{c} 0/\infty/1 \\ \end{array}$
Flow Network Example

Legend

- $V$: Nodes
- $F$: Edges

$\ell/u/cost$:
- $V \times F \supseteq 1/4/0$
- $F \times F \supseteq 0/\infty/1$
Flow Network Example

Legend

\[ V \quad \circ \]

\[ F \quad \circ \]

\[ \ell/u/cost \]

\[ V \times F \supseteq 1/4/0 \]

\[ F \times F \supseteq 0/\infty/1 \]
Flow Network Example

Legend

- $V$: Source nodes
- $F$: Sink nodes
- $\ell/u/cost$: Flow values

$V \times F \supseteq 1/4/0$

$F \times F \supseteq 0/\infty/1$
Flow Network Example

Legend

- $V$ (source nodes)
- $F$ (sink nodes)

$V \times F \succeq 1/4/0$

$F \times F \succeq 0/\infty/1$
Flow Network Example

Legend:
- $V$ (circles)
- $F$ (circles)
- $\ell/u/cost$

$V \times F \supseteq 1/4/0$
$F \times F \supseteq 0/\infty/1$

$4 = b$-value
Flow Network Example

Legend

- $V$: $\bullet$
- $F$: $\circ$

$\ell/u/cost$

$V \times F \supseteq \begin{pmatrix} 1/4/0 \end{pmatrix}$

$F \times F \supseteq \begin{pmatrix} 0/\infty/1 \end{pmatrix}$

$4 = b$-value

flow

Graphical representation of a flow network with nodes, edges, and flow values.
Flow Network Example

Legend

\[ V \quad \bullet \]
\[ F \quad \circ \]
\[ \ell/u/cost \]

\[ V \times F \supseteq \begin{array}{c} 1/4/0 \\ 0/\infty/1 \end{array} \]

\[ 4 = b\text{-value} \]

3 flow
Flow Network Example

Legend

- $V$: source $v_1$ and sink $v_4$
- $F$: middle vertices
- $\ell/u/cost$: edge labels

- $V \times F \supseteq$: $1/4/0$
- $F \times F \supseteq$: $0/\infty/1$
- $4 = b$-value

Legend:
- $\{1\}$ flow
- $\{2\}$ flow
- $\{3\}$ flow
- $\{4\}$ flow

Graph:
- $f_0$ from $v_1$ to $v_4$ with capacity $-14$
- $e_1$: $v_1$ to $v_2$ with capacity $4$
- $e_2$: $v_2$ to $v_3$ with capacity $1$
- $e_3$: $v_2$ to $v_4$ with capacity $4$
- $e_4$: $v_4$ to $v_1$ with capacity $4$
- $e_5$: $v_1$ to $v_3$ with capacity $4$
- $e_6$: $v_1$ to $v_5$ with capacity $-2$
- $f_1$: $v_1$ to $v_3$ with capacity $1$
- $f_2$: $v_2$ to $v_4$ with capacity $1$
- $f_3$: $v_3$ to $v_4$ with capacity $3$
Flow Network Example

Legend

\( V \quad \bullet \)

\( F \quad \circ \)

\( \ell/u/cost \)

\( V \times F \supseteq \frac{1}{4}/0 \)

\( F \times F \supseteq 0/\infty/1 \)

4 = b-value

flow

cost = 1
one bend (outward)
Flow Network Example

Legend

$V$ → ●

$F$ → ○

$\ell/u/cost$

$V \times F \supseteq 1/4/0$

$F \times F \supseteq 0/\infty/1$

$4 = b$-value

flow

$\{V, F, \{\ell/u/cost\}\}$
Flow Network Example

Legend

- $V$: Vertices
- $F$: Edges

- $\ell/u/cost$:
  - $V \times F \supseteq 1/4/0$
  - $F \times F \supseteq 0/\infty/1$

- $4 = b$-value
- $f_0$ = $b$-value

(flow)}
Flow Network Example

Legend
- $V$ (black circle) represents vertices.
- $F$ (red circle) represents flows.
- $\ell/u/cost$ indicates the cost of each edge.
- $V \times F \supseteq 1/4/0$ for an edge from $v_2$ to $v_3$.
- $F \times F \supseteq 0/\infty/1$ for an edge from $v_3$ to $v_4$.
- $4 = b$-value
- $3$ flow

The diagram shows a flow network with vertices $v_1, v_2, v_3, v_4, v_5, v_6$ and edges $e_1, e_2, e_3, e_4, e_5, e_6$.
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
Bend Minimization – Result

**Theorem.** [Tamassia ‘87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\(\iff\) Given valid flow \(X\) in \(N(G)\) with cost \(k\).
Bend Minimization – Result

Theorem. [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

Proof.
\(\iff\) Given valid flow \(X\) in \(N(G)\) with cost \(k\).

Construct orthogonal representation \(H(G)\) with \(k\) bends.
Bend Minimization – Result

Theorem. [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

Proof.
\(\Leftarrow\) Given valid flow \(X\) in \(N(G)\) with cost \(k\).
   Construct orthogonal representation \(H(G)\) with \(k\) bends.
- Transform from flow to orthogonal description.
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[
\iff \quad \text{Given valid flow } X \text{ in } N(G) \text{ with cost } k.
\]
Construct orthogonal representation \(H(G)\) with \(k\) bends.

- Transform from flow to orthogonal description.
- Show properties (H1)–(H4).

(H1) \(H(G)\) corresponds to \(F, f_0\).

(H2) For each edge \({u, v}\) shared by faces \(f\) and \(g\), sequence \(\delta_1\) is reversed and inverted \(\delta_2\).

(H3) For each face \(f\) it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise}.
\end{cases}
\]

(H4) For each vertex \(v\) the sum of incident angles is \(2\pi\).
Bend Minimization – Result

**Theorem.** [Tamassia '87] A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[\iff \]
\[\leq\qquad \text{Given valid flow } X \text{ in } N(G) \text{ with cost } k. \]

Construct orthogonal representation \(H(G)\) with \(k\) bends.

- Transform from flow to orthogonal description.
- Show properties (H1)–(H4).

<table>
<thead>
<tr>
<th>(H1)</th>
<th>(H(G)) matches (F, f_0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(H2)</td>
<td>For each edge ({u, v}) shared by faces (f) and (g), sequence (\delta_1) is reversed and inverted (\delta_2).</td>
</tr>
<tr>
<td>(H3)</td>
<td>For each face (f) it holds that: [\sum_{r \in H(f)} C(r) = \begin{cases} -4 &amp; \text{if } f = f_0 \ +4 &amp; \text{otherwise}. \end{cases} ]</td>
</tr>
<tr>
<td>(H4)</td>
<td>For each vertex (v) the sum of incident angles is (2\pi).</td>
</tr>
</tbody>
</table>
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[
\iff
\]
- Given valid flow \(X\) in \(N(G)\) with cost \(k\).
  
  Construct orthogonal representation \(H(G)\) with \(k\) bends.

  ■ Transform from flow to orthogonal description.

  ■ Show properties (H1)–(H4).

  (H1) \(H(G)\) matches \(F, f_0\)
  
  (H2) For each edge \(\{u, v\}\) shared by faces \(f\) and \(g\), sequence \(\delta_1\) is reversed and inverted \(\delta_2\).

  (H3) For each face \(f\) it holds that:
  \[
  \sum_{r \in H(f)} C(r) = \begin{cases} 
  -4 & \text{if } f = f_0 \\
  +4 & \text{otherwise.}
  \end{cases}
  \]

  (H4) For each vertex \(v\) the sum of incident angles is \(2\pi\).
Bend Minimization – Result

**Theorem.** [Tamassia ‘87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[\iff\]
- Given valid flow \(X\) in \(N(G)\) with cost \(k\).
  - Construct orthogonal representation \(H(G)\) with \(k\) bends.
  - Transform from flow to orthogonal description.
  - Show properties (H1)–(H4).

(H1) \(H(G)\) matches \(F, f_0\)

(H2) Bend order inverted and reversed on opposite sides

(H3) For each face \(f\) it holds that:
\[\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}\]

(H4) For each vertex \(v\) the sum of incident angles is \(2\pi\).
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[\Leftrightarrow\]
Given valid flow \(X\) in \(N(G)\) with cost \(k\).
Construct orthogonal representation \(H(G)\) with \(k\) bends.

- Transform from flow to orthogonal description.
- Show properties (H1)–(H4).

(H1) \(H(G)\) matches \(F, f_0\).
(H2) Bend order inverted and reversed on opposite sides.
(H3) Angle sum of \(f = \pm 4\).
(H4) Total angle at each vertex = \(2\pi\).

(H1) \(H(G)\) corresponds to \(F, f_0\).
(H2) For each edge \(\{u, v\}\) shared by faces \(f\) and \(g\), sequence \(\delta_1\) is reversed and inverted \(\delta_2\).
(H3) For each face \(f\) it holds that:
\[
\sum_{r \in H(f)} C(r) = \begin{cases} 
-4 & \text{if } f = f_0 \\
+4 & \text{otherwise.}
\end{cases}
\]
(H4) For each vertex \(v\) the sum of incident angles is \(2\pi\).
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[
\Rightarrow \text{Given an orthogonal representation } H(G) \text{ with } k \text{ bends. Construct valid flow } X \text{ in } N(G) \text{ with cost } k.
\]
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph $(G, F, f_0)$ has a valid orthogonal representation $H(G)$ with $k$ bends iff the flow network $N(G)$ has a valid flow $X$ with cost $k$.

**Proof.**

$\Rightarrow$ Given an orthogonal representation $H(G)$ with $k$ bends.

- Construct valid flow $X$ in $N(G)$ with cost $k$.
- Define flow $X : E \rightarrow \mathbb{R}_0^+$.
- Show that $X$ is a valid flow and has cost $k$. 
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**

\[ \Rightarrow \] Given an orthogonal representation \(H(G)\) with \(k\) bends.

Construct valid flow \(X\) in \(N(G)\) with cost \(k\).

- Define flow \(X: E \rightarrow \mathbb{R}_0^+\).
- Show that \(X\) is a valid flow and has cost \(k\).

- \(b(v) = 4\ \ \forall v \in V\)
- \(b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases}\)
- \(\ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)\)
- \(\text{cost}(v, f) = 0\)
- \(\ell(f, g) := 0 \leq X(f, g) \leq \infty =: u(f, g)\)
- \(\text{cost}(f, g) = 1\)
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[\Rightarrow \text{ Given an orthogonal representation } H(G) \text{ with } k \text{ bends.} \]
\[\text{Construct valid flow } X \text{ in } N(G) \text{ with cost } k. \]

- Define flow \(X : E \mapsto \mathbb{R}^+_0\).
- Show that \(X\) is a valid flow and has cost \(k\).

\((N1)\) \(X(vf) = 1/2/3/4\) 

\[\begin{align*}
& b(v) = 4 \quad \forall v \in V \\
& b(f) = -2 \deg_G(f) + \begin{cases} 
-4 & \text{if } f = f_0, \\
+4 & \text{otherwise}
\end{cases} \\
& \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f) \\
& \text{cost}(v, f) = 0 \\
& \ell(f, g) := 0 \leq X(f, g) \leq \infty =: u(f, g) \\
& \text{cost}(f, g) = 1
\end{align*}\]
Bend Minimization – Result

**Theorem.** [Tamassia '87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\(\Rightarrow\) Given an orthogonal representation \(H(G)\) with \(k\) bends.
Construct valid flow \(X\) in \(N(G)\) with cost \(k\).

■ Define flow \(X : E \to \mathbb{R}_0^+\).
■ Show that \(X\) is a valid flow and has cost \(k\).

(N1) \(X(vf) = 1/2/3/4\)

(N2) \(X(fg) = |\delta_{fg}|_0, (e, \delta_{fg}, x)\) describes \(e^* = fg\) from \(f\)
Bend Minimization – Result

**Theorem.** [Tamassia ’87]
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**
\[\Rightarrow\] Given an orthogonal representation \(H(G)\) with \(k\) bends.
Construct valid flow \(X\) in \(N(G)\) with cost \(k\).

- Define flow \(X : E \rightarrow \mathbb{R}^+_0\).
- Show that \(X\) is a valid flow and has cost \(k\).

\((N1)\) \(X(vf) = 1/2/3/4\)

\((N2)\) \(X(fg) = |\delta_{fg}|_0, (e, \delta_{fg}, x)\) describes \(e^* = fg\) from \(f\)

\((N3)\) capacities, deficit/demand coverage

- \(b(v) = 4\) \(\forall v \in V\)
- \(b(f) = -2 \deg_G(f) + \begin{cases} -4 & \text{if } f = f_0, \\ +4 & \text{otherwise} \end{cases}\)
- \(\ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)\)
  \(\text{cost}(v, f) = 0\)
- \(\ell(f, g) := 0 \leq X(f, g) \leq \infty =: u(f, g)\)
  \(\text{cost}(f, g) = 1\)
Bend Minimization – Result

**Theorem.** [Tamassia ’87]  
A plane graph \((G, F, f_0)\) has a valid orthogonal representation \(H(G)\) with \(k\) bends iff the flow network \(N(G)\) has a valid flow \(X\) with cost \(k\).

**Proof.**  
⇒ Given an orthogonal representation \(H(G)\) with \(k\) bends.  
   Construct valid flow \(X\) in \(N(G)\) with cost \(k\).

■ Define flow \(X : E \rightarrow \mathbb{R}_+^+\).

■ Show that \(X\) is a valid flow and has cost \(k\).

(N1) \(X(vf) = 1/2/3/4\)  
(N2) \(X(fg) = |\delta_{fg}|_0, (e, \delta_{fg}, x)\) describes \(e^* = fg\) from \(f\)  
(N3) capacities, deficit/demand coverage  
(N4) \(\text{cost} = k\)
Bend Minimization – Remarks

- From Theorem follows that the combinatorial orthogonal bend minimization problem for plane graphs can be solved using an algorithm for the Min-Cost-Flow problem.
From Theorem follows that the combinatorial orthogonal bend minimization problem for plane graphs can be solved using an algorithm for the Min-Cost-Flow problem.

**Theorem.** [Garg & Tamassia 1996]
The minimum cost flow problem can be solved in $O(|X^*|^{3/4} m^{1/4} \sqrt{\log n})$ time.
Bend Minimization – Remarks

- From Theorem follows that the combinatorial orthogonal bend minimization problem for plane graphs can be solved using an algorithm for the Min-Cost-Flow problem.

**Theorem.** [Garg & Tamassia 1996]
The minimum cost flow problem for planar graphs with bounded costs and vertex degrees can be solved in $O(n^{7/4} \sqrt{\log n})$ time.
Bend Minimization – Remarks

- From Theorem follows that the combinatorial orthogonal bend minimization problem for plane graphs can be solved using an algorithm for the Min-Cost-Flow problem.

**Theorem.** [Cornelsen & Karrenbauer 2011] The minimum cost flow problem for planar graphs with bounded costs and face sizes can be solved in $O(n^{3/2})$ time.

**Theorem.** [Garg & Tamassia 1996] The minimum cost flow problem for planar graphs with bounded costs and vertex degrees can be solved in $O(n^{7/4} \sqrt{\log n})$ time.
**Bend Minimization – Remarks**

- From Theorem follows that the combinatorial orthogonal bend minimization problem for plane graphs can be solved using an algorithm for the Min-Cost-Flow problem.

**Theorem.** [Garg & Tamassia 1996]
The minimum cost flow problem for planar graphs with bounded costs and vertex degrees can be solved in $O(n^{7/4}\sqrt{\log n})$ time.

**Theorem.** [Cornelsen & Karrenbauer 2011]
The minimum cost flow problem for planar graphs with bounded costs and faze sizes can be solved in $O(n^{3/2})$ time.

**Theorem.** [Garg & Tamassia 2001]
Bend Minimization without a given combinatorial embedding is an NP-hard problem.
Visualization of Graphs

Lecture 5:
Orthogonal Layouts

Part IV:
Area Minimization

Jonathan Klawitter
Topology – Shape – Metrics

Three-step approach:

\[ V = \{v_1, v_2, v_3, v_4\} \]
\[ E = \{v_1v_2, v_1v_3, v_1v_4, v_2v_3, v_2v_4\} \]

-combinatorial embedding/planarization-
-reduce crossings-

-planar orthogonal drawing-
-bend minimization-
-orthogonal representation-

-planar orthogonal drawing-
-area minimization-

-TOPOLOGY -- SHAPE -- METRICS-

[Tamassia 1987]
Compaction problem.

Given:

Find:
Compaction problem.

Given: ■ Plane graph $G = (V, E)$ with maximum degree 4

Find:
Compaction problem.

Given:
- Plane graph $G = (V, E)$ with maximum degree 4
- Orthogonal representation $H(G)$

Find:
Compaction

Compaction problem.

Given:
- Plane graph $G = (V, E)$ with maximum degree 4
- Orthogonal representation $H(G)$

Find:
- Compact orthogonal layout of $G$ that realizes $H(G)$
Compaction

**Compaction problem.**

Given:
- Plane graph $G = (V, E)$ with maximum degree 4
- Orthogonal representation $H(G)$

Find:
Compact orthogonal layout of $G$ that realizes $H(G)$

**Special case.**

All faces are rectangles.
Compaction

**Compaction problem.**

*Given:*  
- Plane graph $G = (V, E)$ with maximum degree 4  
- Orthogonal representation $H(G)$

*Find:*  
Compact orthogonal layout of $G$ that realizes $H(G)$

**Special case.**

All faces are rectangles.

→ Guarantees possible
Compaction

**Compaction problem.**
Given:
- Plane graph $G = (V, E)$ with maximum degree 4
- Orthogonal representation $H(G)$

Find:
Compact orthogonal layout of $G$ that realizes $H(G)$

**Special case.**
All faces are rectangles.

→ Guarantees possible
- minimum total edge length
- minimum area
Compaction

Compaction problem.
Given:  ■ Plane graph \( G = (V, E) \) with maximum degree 4
        ■ Orthogonal representation \( H(G) \)
Find:   Compact orthogonal layout of \( G \) that realizes \( H(G) \)

Special case.
All faces are rectangles.

→ Guarantees possible   ■ minimum total edge length
     ■ minimum area
Compaction

Compaction problem.
Given:  ■ Plane graph $G = (V, E)$ with maximum degree 4  
        ■ Orthogonal representation $H(G)$
Find:   Compact orthogonal layout of $G$ that realizes $H(G)$

Special case.
All faces are rectangles.

→ Guarantees possible  ■ minimum total edge length  
   ■ minimum area

Properties.
Compaction

Compaction problem.
Given:  ■ Plane graph $G = (V, E)$ with maximum degree 4
        ■ Orthogonal representation $H(G)$
Find:   Compact orthogonal layout of $G$ that realizes $H(G)$

Special case.
All faces are rectangles.
→ Guarantees possible
    ■ minimum total edge length
    ■ minimum area

Properties.
■ bends only on the outer face
Compaction

**Compaction problem.**
- Given:  
  - Plane graph $G = (V, E)$ with maximum degree 4
  - Orthogonal representation $H(G)$
- Find:  
  - Compact orthogonal layout of $G$ that realizes $H(G)$

**Special case.**
All faces are rectangles.
→ Guarantees possible
  - minimum total edge length
  - minimum area

**Properties.**
- bends only on the outer face
- opposite sides of a face have the same length
Compaction

**Compaction problem.**
Given:  
- Plane graph $G = (V, E)$ with maximum degree 4  
- Orthogonal representation $H(G)$
Find:  
- Compact orthogonal layout of $G$ that realizes $H(G)$

**Special case.**
All faces are rectangles.

→ Guarantees possible  
- minimum total edge length  
- minimum area

**Properties.**
- bends only on the outer face  
- opposite sides of a face have the same length

**Idea.**
- Formulate flow network for horizontal/vertical compaction
Flow Network for Edge Length Assignment
Flow Network for Edge Length Assignment

**Definition.**
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; l; u; \text{cost})$
Flow Network for Edge Length Assignment

**Definition.**

Flow Network \( N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost}) \)

- \( W_{\text{hor}} = F \setminus \{f_0\} \)
- \( E_{\text{hor}} = \{(f,g) | f,g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t,s)\} \)
- \( \ell(a) = 1 \forall a \in E_{\text{hor}} \)
- \( u(a) = \infty \forall a \in E_{\text{hor}} \)
- \( \text{cost}(a) = 1 \forall a \in E_{\text{hor}} \)
- \( b(f) = 0 \forall f \in W_{\text{hor}} \)
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; ℓ; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \forall a \in E_{\text{hor}}$
- $u(a) = \infty \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \forall a \in E_{\text{hor}}$
- $b(f) = 0 \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

**Definition.**
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1$ for all $a \in E_{\text{hor}}$
- $u(a) = \infty$ for all $a \in E_{\text{hor}}$
- $\text{cost}(a) = 1$ for all $a \in E_{\text{hor}}$
- $b(f) = 0$ for all $f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \forall a \in E_{\text{hor}}$
- $u(a) = \infty \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \forall a \in E_{\text{hor}}$
- $b(f) = 0 \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; ℓ; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\}$

\[
\begin{align*}
W_{\text{hor}} & = F \setminus \{f_0\} \cup \{s, t\} \\
E_{\text{hor}} & = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\}
\end{align*}
\]
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) | f, g \text{ share a horizontal segment and } f \text{ lies below } g\}$

- $\ell(a) = 1 \forall a \in E_{\text{hor}}$
- $u(a) = \infty \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \forall a \in E_{\text{hor}}$
- $b(f) = 0 \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

**Definition.**
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

**Definition.**
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$

- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

Definition.
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

**Definition.**

Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
Flow Network for Edge Length Assignment

**Definition.**
Flow Network $N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost})$

- $W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{hor}} = \{(f, g) | f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in E_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$
Flow Network for Edge Length Assignment

**Definition.**
Flow Network \( N_{\text{hor}} = ((W_{\text{hor}}, E_{\text{hor}}); b; \ell; u; \text{cost}) \)

- \( W_{\text{hor}} = F \setminus \{f_0\} \cup \{s, t\} \)
- \( E_{\text{hor}} = \{(f, g) \mid f, g \text{ share a horizontal segment and } f \text{ lies below } g\} \cup \{(t, s)\} \)
- \( \ell(a) = 1 \quad \forall a \in E_{\text{hor}} \)
- \( u(a) = \infty \quad \forall a \in E_{\text{hor}} \)
- \( \text{cost}(a) = 1 \quad \forall a \in E_{\text{hor}} \)
- \( b(f) = 0 \quad \forall f \in W_{\text{hor}} \)
Flow Network for Edge Length Assignment

**Definition.**
Flow Network $N_{\text{ver}} = ((W_{\text{ver}}, E_{\text{ver}}); b; \ell; u; \text{cost})$

- $W_{\text{ver}} = F \setminus \{f_0\} \cup \{s, t\}$
- $E_{\text{ver}} = \{(f, g) \mid f, g \text{ share a vertical segment and } f \text{ lies to the left of } g\} \cup \{(t, s)\}$
- $\ell(a) = 1 \quad \forall a \in E_{\text{ver}}$
- $u(a) = \infty \quad \forall a \in E_{\text{ver}}$
- $\text{cost}(a) = 1 \quad \forall a \in E_{\text{ver}}$
- $b(f) = 0 \quad \forall f \in W_{\text{ver}}$
Compaction – Result

Theorem. Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.
Compaction – Result

Theorem.
Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.

What values of the drawing represent the following?

$|X_{\text{hor}}(t,s)|$ and $|X_{\text{ver}}(t,s)|$?

$\sum_{e \in E_{\text{hor}}} X_{\text{hor}}(e) + \sum_{e \in E_{\text{ver}}} X_{\text{ver}}(e)$
Compaction – Result

Theorem.
Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.

What values of the drawing represent the following?
- $|X_{\text{hor}}(t, s)|$ and $|X_{\text{ver}}(t, s)|$?
Theorem.
Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.

What values of the drawing represent the following?

$|X_{\text{hor}}(t, s)|$ and $|X_{\text{ver}}(t, s)|$?

width and height of drawing
Compaction – Result

Theorem.
Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.

What values of the drawing represent the following?

- $|X_{\text{hor}}(t, s)|$ and $|X_{\text{ver}}(t, s)|$?
- $\sum_{e \in E_{\text{hor}}} X_{\text{hor}}(e) + \sum_{e \in E_{\text{ver}}} X_{\text{ver}}(e)$

width and height of drawing
Theorem.
Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.

What values of the drawing represent the following?

- $|X_{\text{hor}}(t, s)|$ and $|X_{\text{ver}}(t, s)|$?
- $\sum_{e \in E_{\text{hor}}} X_{\text{hor}}(e) + \sum_{e \in E_{\text{ver}}} X_{\text{ver}}(e)$
  width and height of drawing
  total edge length
Compaction – Result

What values of the drawing represent the following?

- $|X_{\text{hor}}(t,s)|$ and $|X_{\text{ver}}(t,s)|$?
- $\sum_{e \in E_{\text{hor}}} X_{\text{hor}}(e) + \sum_{e \in E_{\text{ver}}} X_{\text{ver}}(e)$

What if not all faces rectangular?

**Theorem.**
Valid min-cost-flows for $N_{\text{hor}}$ and $N_{\text{ver}}$ exists iff corresponding edge lengths induce orthogonal drawing.

width and height of drawing

total edge length
Refinement of \((G, H)\) – Inner Face
Refinement of \((G, H)\) – Inner Face

![Diagram of a graph with labeled vertices and edges, with a note about dummy vertices for bends.]

- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

- Dummy vertices for bends
Refinement of \((G, H) – \text{Inner Face}\)

- **corner(e)**
- **Dummy vertices for bends**
Refinement of \((G, H) – \text{Inner Face}\)

Diagram showing:
- \(e\)
- \(\text{corner}(e)\)
- \(\text{next}(e)\)
- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

- \(\text{corner}(e)\)
- \(\text{next}(e)\)
- \(f\)

- Dummy vertices for bends
Refinement of \((G, H) \) – Inner Face

\[ \text{corner}(e) \]
\[ \text{next}(e) \]

\[ \text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn} 
\end{cases} \]

- Dummy vertices for bends
Refinement of \((G, H) – \text{Inner Face}\)

\[\text{corner}(e)\]
\[\text{next}(e)\]

\[\begin{array}{c}
\text{turn}(e) = \\
\begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn}
\end{cases}
\end{array}\]

- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

\[
\text{corner}(e)\quad \text{next}(e)\quad \text{front}(e') \quad \text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn}
\end{cases}
\]

- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

\[\text{corner}(e)\]

\[\text{next}(e)\]

\[\text{front}(e')\]

\[\text{extend}(e')\]

\[\text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn} 
\end{cases}\]
Refinement of \((G, H) – \text{Inner Face}\)

\[\text{corner}(e)\]

\[\text{next}(e)\]

\[\text{front}(e')\]

\[\text{project}(e')\]

\[\text{extend}(e')\]

\[\text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn} 
\end{cases}\]

- Dummy vertices for bends
Refinement of \((G, H) – Inner Face\)

- \(\text{next}(e)\)
- \(\text{corner}(e)\)
- \(\text{project}(e')\)
- \(\text{extend}(e')\)

\[
\text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn} 
\end{cases}
\]

- Dummy vertices for bends
Refinement of \((G, H) – Inner Face\)

\[ \begin{align*}
\text{corner}(e) & = 1 \\
\text{next}(e) & = -1 \\
\text{project}(e') & = f \\
\text{extend}(e') & = 1 \\
\text{turn}(e) & = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn}
\end{cases}
\end{align*} \]
Refinement of \((G, H) - \text{Inner Face}\)

- corner\((e)\)
- next\((e)\)
- extend\((e')\)
- project\((e')\)

\[
\text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn}
\end{cases}
\]

- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

\[\text{corner}(e)\]

\[\text{next}(e)\]

\[f\]

\[\text{project}(e')\]

\[\text{extend}(e')\]

\[\text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn} \end{cases}\]

- Dummy vertices for bends
Refinement of \((G, H)\) – Inner Face

\[
\text{turn}(e) = \begin{cases} 
1 & \text{left turn} \\
0 & \text{no turn} \\
-1 & \text{right turn}
\end{cases}
\]

- Dummy vertices for bends

Graphical representation of the refinement process with arrows indicating next\((e)\), corner\((e)\), extend\((e')\), and project\((e')\).
Refinement of \((G, H) - \) Inner Face

- \text{corner}(e)
- \text{next}(e)
- \text{project}(e')
- \text{extend}(e')
- \text{turn}(e) = \begin{cases} 1 & \text{left turn} \\ 0 & \text{no turn} \\ -1 & \text{right turn} \end{cases}

- Dummy vertices for bends
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H)\) – Outer Face

\[ f_0 \]
Refinement of $(G, H)$ – Outer Face
Refinement of $(G, H)$ – Outer Face
Refinement of $(G, H)$ – Outer Face
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H) – \text{Outer Face}\)
Refinement of \((G, H)\) – Outer Face
Refinement of \((G, H)\) – Outer Face
Refinement of $(G, H) – Outer Face$

Area minimized?
Refinement of \((G, H)\) – Outer Face

Area minimized? No!
Refinement of \((G, H)\) – Outer Face

Area minimized? \textbf{No!}

But we get bound \(O((n + b)^2)\) on the area.
Refinement of \((G, H)\) – Outer Face

Area minimized? No!

But we get bound \(O((n + b)^2)\) on the area.

Theorem. \([\text{Patrignani 2001}]\)
Compaction for given orthogonal representation is in general NP-hard.
Visualization of Graphs

Lecture 5: Orthogonal Layouts

Part V: NP-hardness

Jonathan Klawitter
Boundary, belt, and “piston” gadget

$$(w \times h)$$-rectangle
Boundary, belt, and “piston” gadget
Boundary, belt, and “piston” gadget
Boundary, belt, and “piston” gadget
Boundary, belt, and “piston” gadget
Boundary, belt, and “piston” gadget
Boundary, belt, and “piston” gadget
Boundary, belt, and “piston” gadget
Clause gadgets

\begin{figure}
\centering
\includegraphics[width=\textwidth]{clause_gadgets.png}
\end{figure}
Example:

\[ C_1 = x_2 \lor \overline{x_4} \]
\[ C_2 = x_1 \lor x_2 \lor \overline{x_3} \]
\[ C_3 = x_5 \]
\[ C_4 = x_4 \lor \overline{x_5} \]
Clause gadgets

Example:

\[ C_1 = x_2 \lor \overline{x_4} \]
\[ C_2 = x_1 \lor x_2 \lor \overline{x_3} \]
\[ C_3 = x_5 \]
\[ C_4 = x_4 \lor \overline{x_5} \]
Clause gadgets

Example:
\[ C_1 = x_2 \lor \overline{x_4} \]
\[ C_2 = x_1 \lor x_2 \lor \overline{x_3} \]
\[ C_3 = x_5 \]
\[ C_4 = x_4 \lor \overline{x_5} \]

insert \((2n - 1)\)-chain through each clause
Clause gadgets

Example:

\[ C_1 = x_2 \lor \overline{x_4} \]
\[ C_2 = x_1 \lor x_2 \lor \overline{x_3} \]
\[ C_3 = x_5 \]
\[ C_4 = x_4 \lor \overline{x_5} \]

Insert \((2n - 1)\)-chain through each clause
Clause gadgets

Example:

\[ C_1 = x_2 \lor \overline{x_4} \]
\[ C_2 = x_1 \lor x_2 \lor \overline{x_3} \]
\[ C_3 = x_5 \]
\[ C_4 = x_4 \lor \overline{x_5} \]

insert \((2n-1)\)-chain through each clause
Complete reduction

9m + 7

9n + 2
Complete reduction

Pick
$$K = (9n + 2) \cdot (9m + 7)$$
Complete reduction

Pick
$K = (9n + 2) \cdot (9m + 7)$

Then:
$(G, H)$ has an area $K$
drawing
$\iff$
$\Phi$ satisfiable
Literature

- [GD Ch. 5] for detailed explanation

- [Tamassia 1987] “On embedding a graph in the grid with the minimum number of bends”
  original paper on flow for bend minimisation

  NP-hardness proof of compactification