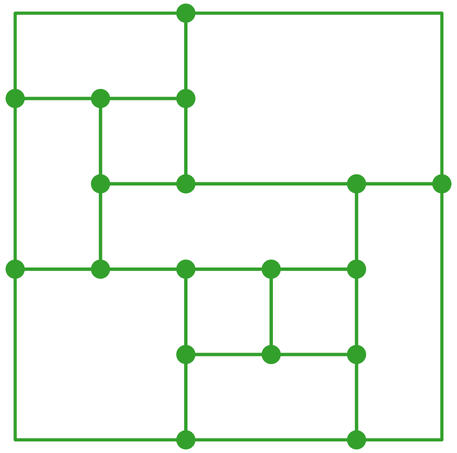
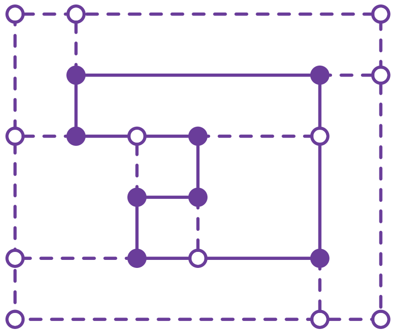
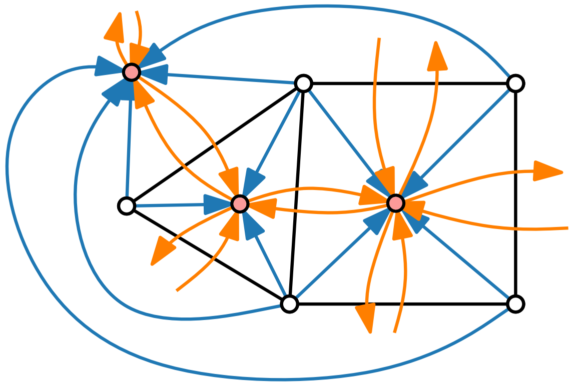
Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Visualization of Graphs



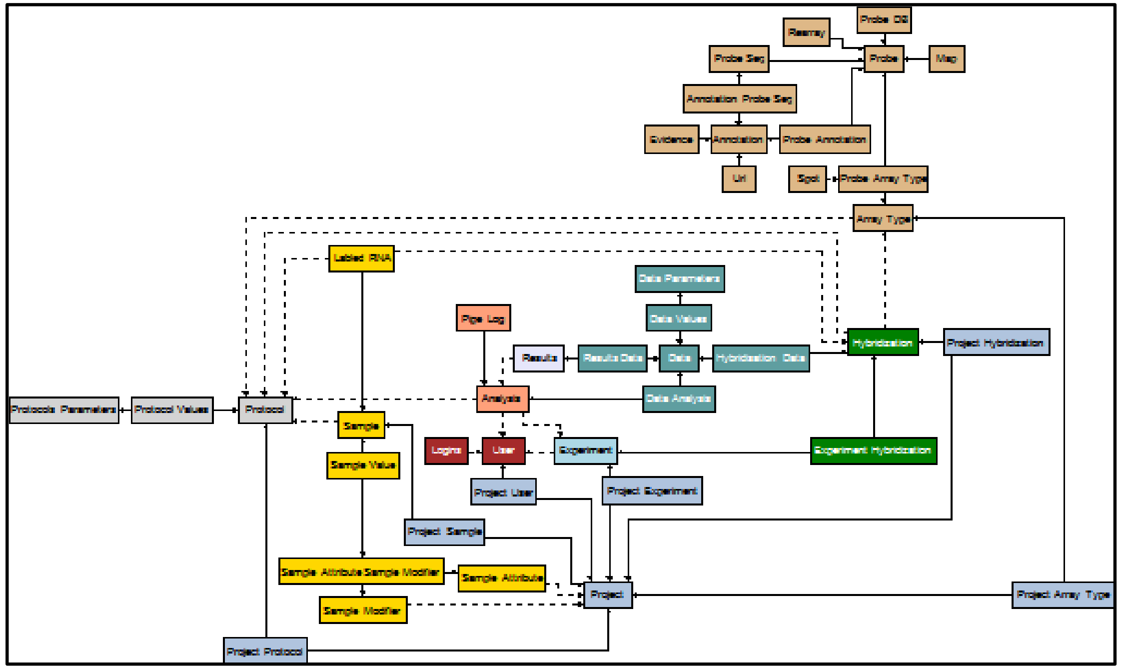
Lecture 6:
Orthogonal Layouts



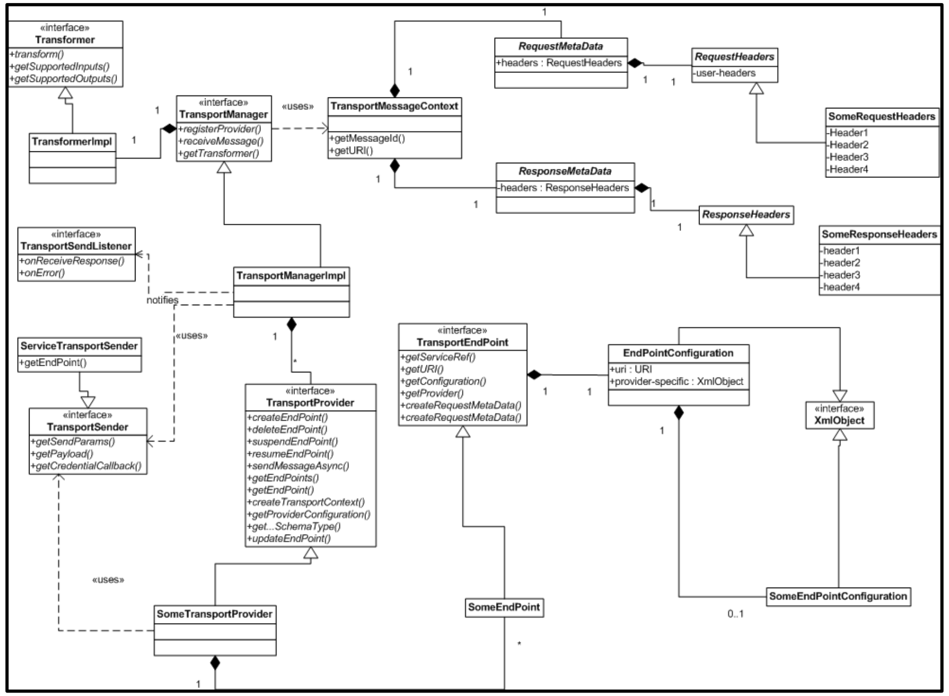


Johannes Zink

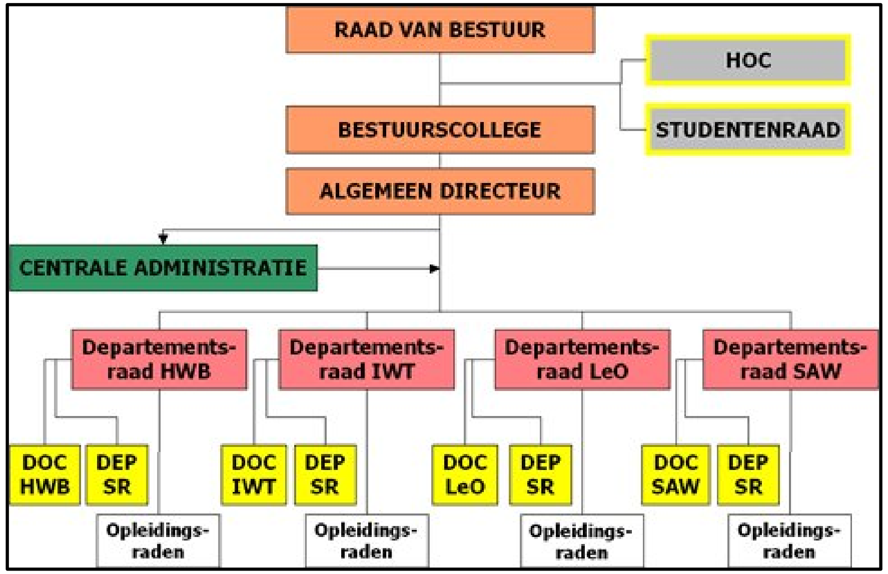
Orthogonal Layout – Applications



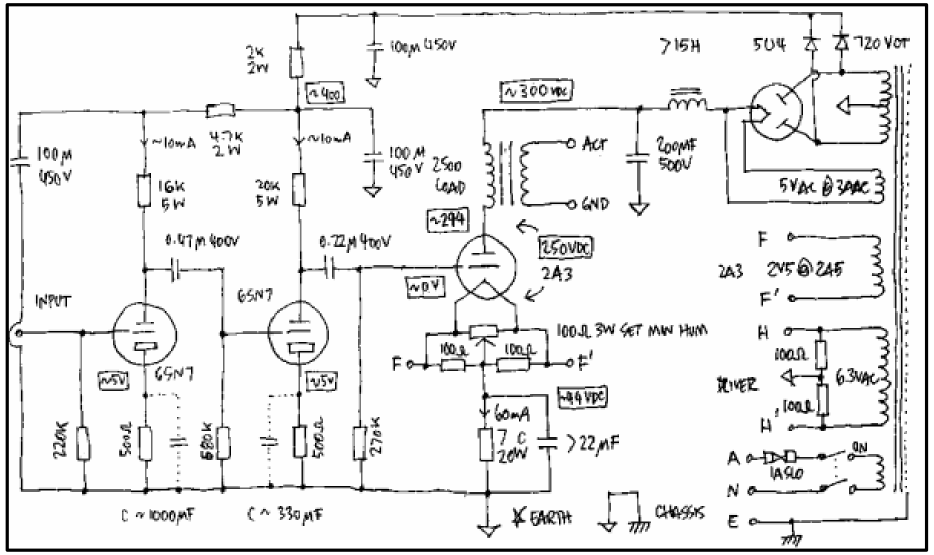
ER diagram in OGDF



UML diagram by Oracle

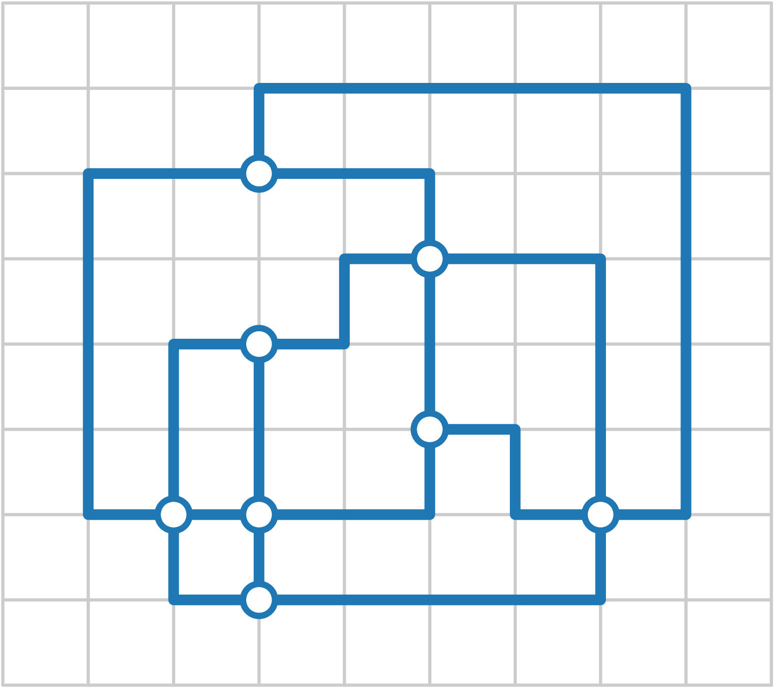


Organigram of HS Limburg



Circuit diagram by Jeff Atwood

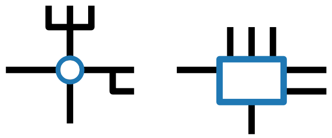
Orthogonal Layout – Definition



Definition.
A drawing Γ 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 line segments of the grid, and
- pairs of edges are disjoint or cross orthogonally.

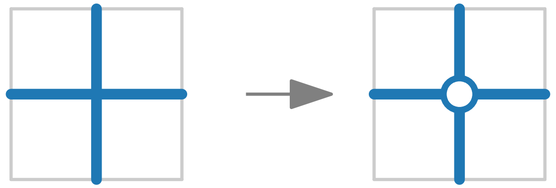
Observations.

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



Planarization.

- Fix embedding
- Crossings become vertices



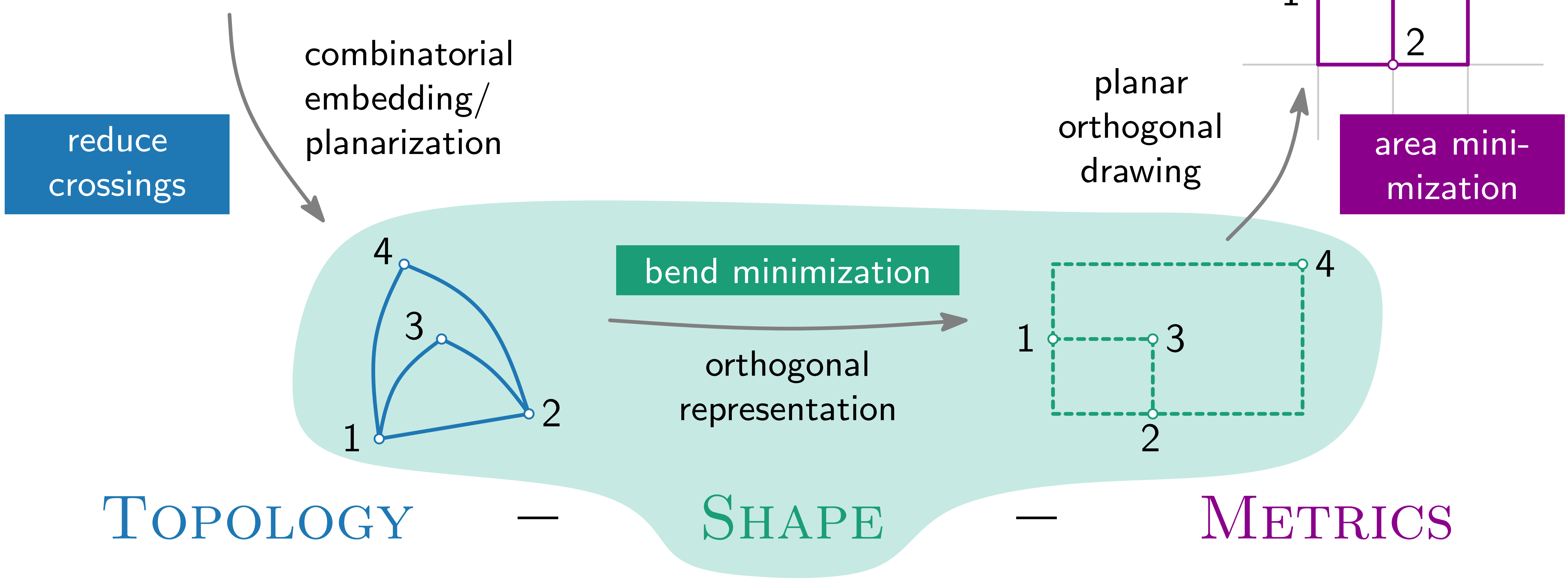
Aesthetic criteria to optimize.

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

Topology – Shape – Metrics

Three-step approach: [Tamassia 1987]

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

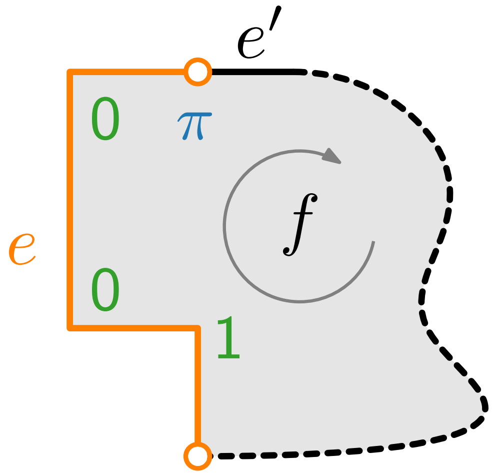


Orthogonal Representation

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 w.r.t. f is a triple (e, δ, α) where
  - $\delta \in \{0,1\}^*$ (where $0 =$ right bend, $1 =$ left bend)
  - α is angle $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ between e and next edge e'



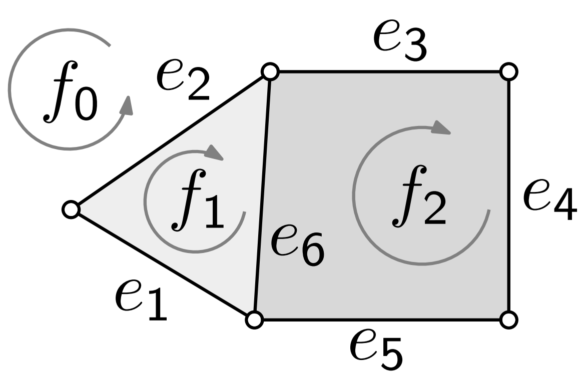
$(e, 100, \pi)$

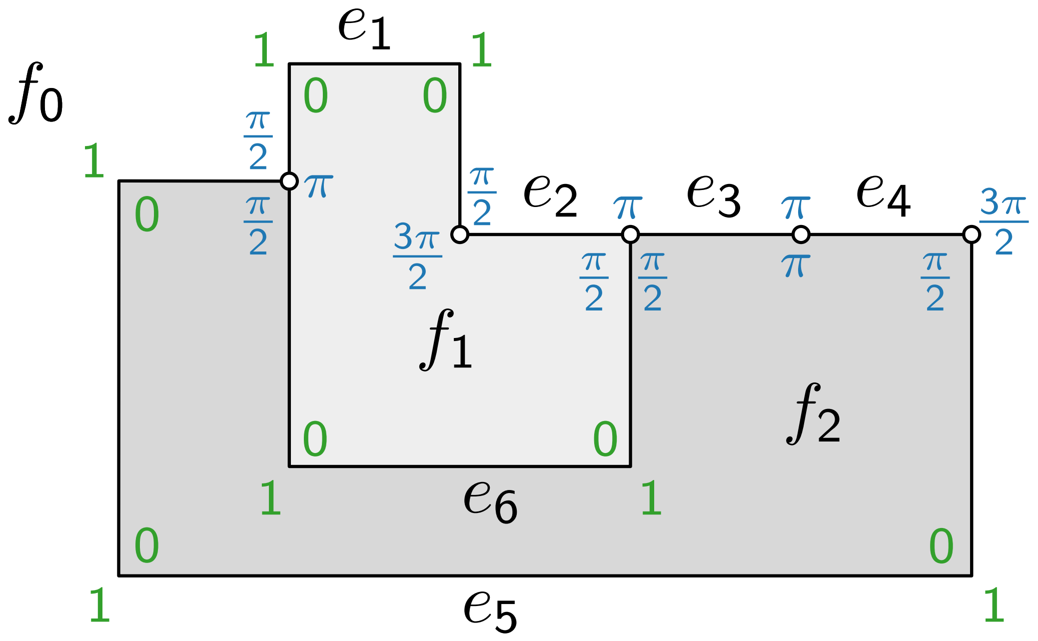
- A **face representation** $H(f)$ of a face f is a clockwise ordered sequence $(e_1, \delta_1, \alpha_1), (e_2, \delta_2, \alpha_2), \dots, (e_{\deg(f)}, \delta_{\deg(f)}, \alpha_{\deg(f)})$ of edge descriptions w.r.t. f.
- An **orthogonal representation** $H(G)$ of G is defined as

$$H(G) = \{H(f) \mid f \in F\}.$$

Orthogonal Representation – Example

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





Concrete coordinates are not fixed yet!

Correctness of an Orthogonal Representation

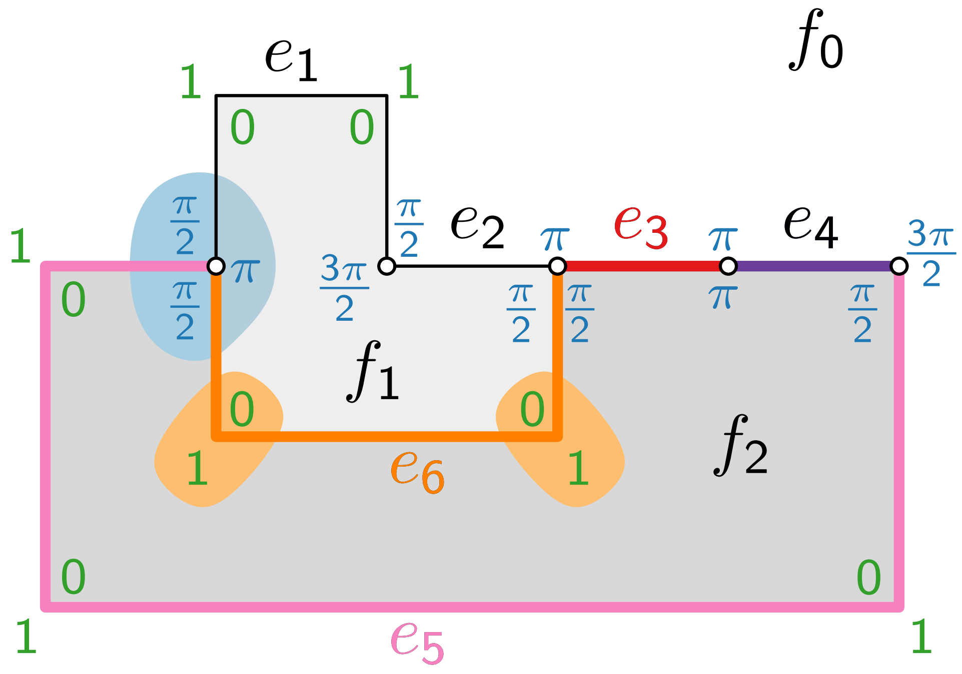
(H1) $H(G)$ corresponds to F, f_0.

(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 δ_1 is like δ_2, but reversed and inverted.

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

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

(H4) For each **vertex** v, the sum of incident angles is 2π.



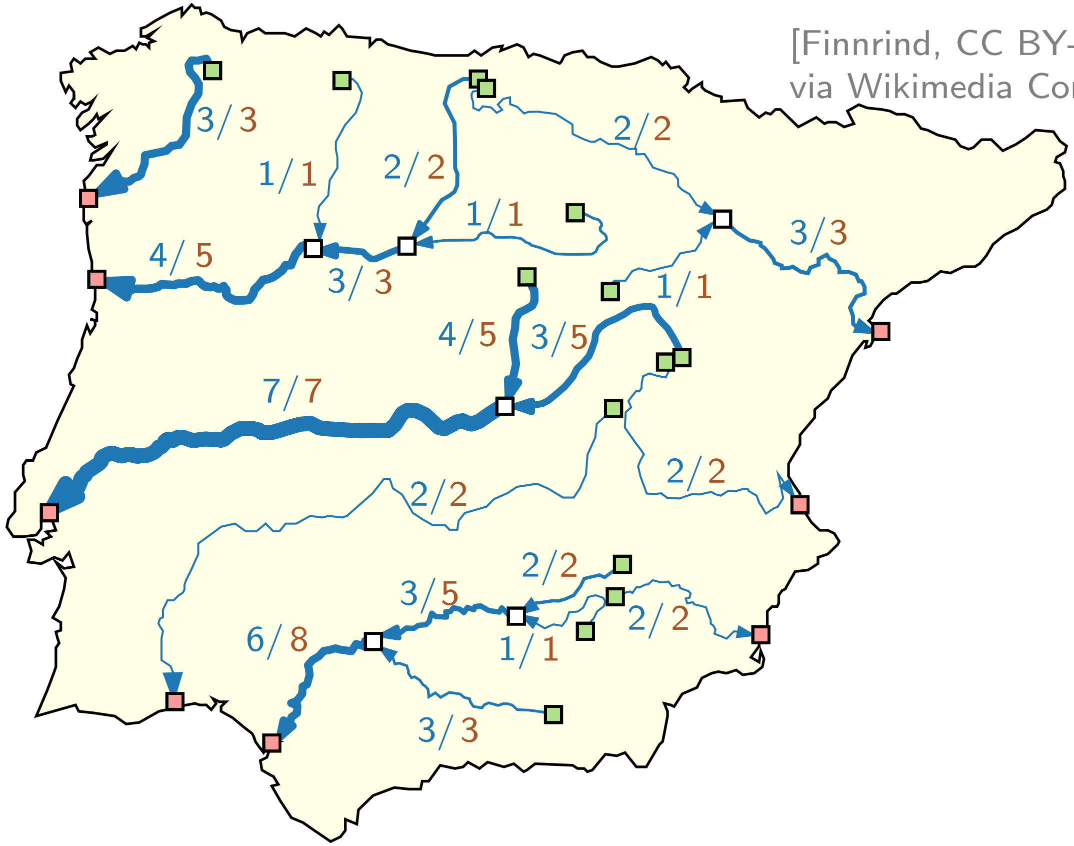
$C(e_3) = 0 - 0 - 2 + 2 = 0$

$C(e_4) = 0 - 0 - 1 + 2 = 1$

$C(e_5) = 3 - 0 - 1 + 2 = 4$

$C(e_6) = 0 - 2 - 1 + 2 = -1$

Reminder: s-t-Flow Networks



[Finnrind, CC BY-SA 3.0, via Wikimedia Commons]

Flow network $(G = (V, E); S, T; u)$ with

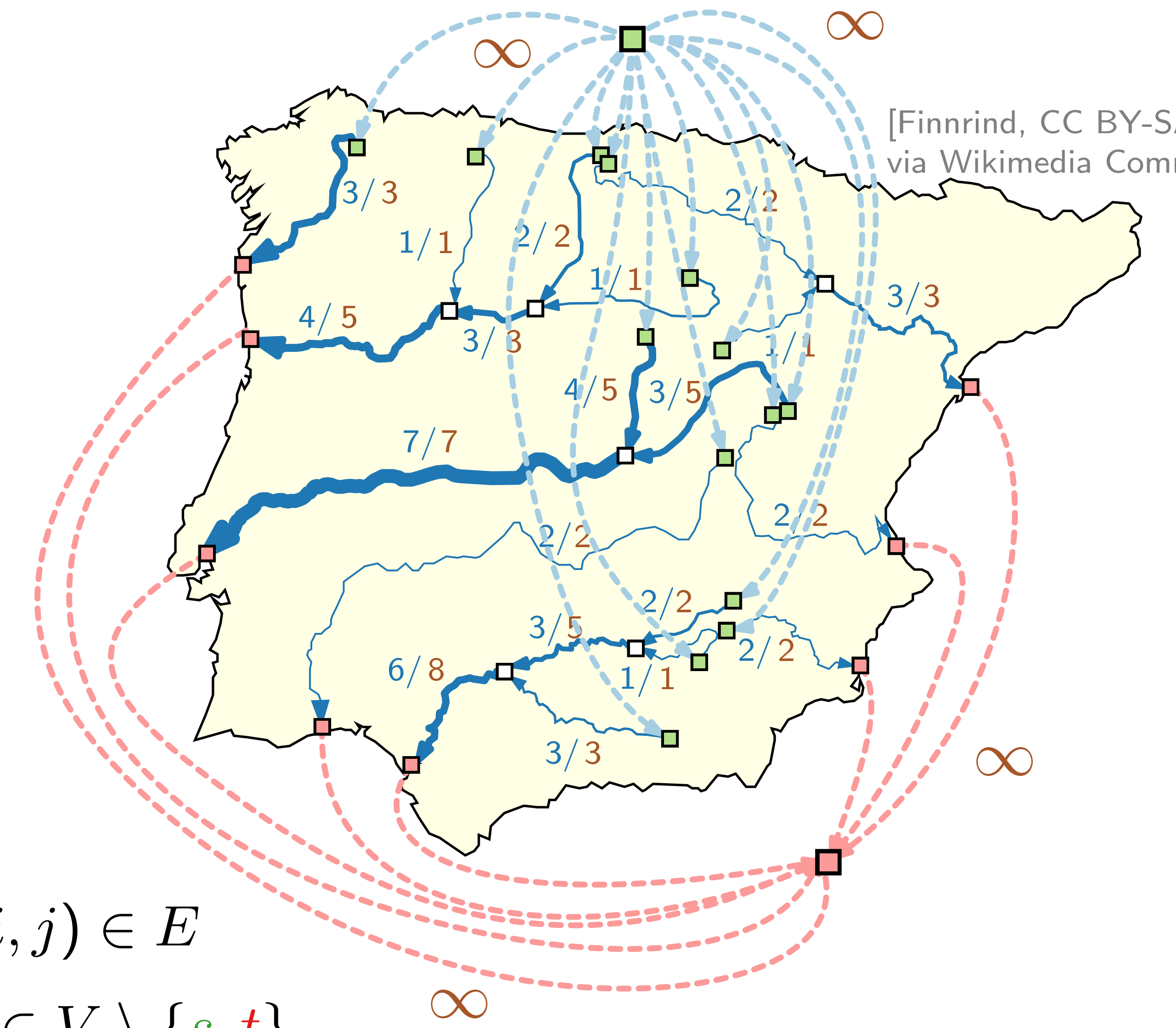
- directed graph $G = (V, E)$
- *sources* $S \subseteq V$, *sinks* $T \subseteq V$
- edge *capacity* $u \colon E \to \mathbb{R}_0^+ \cup \{\infty\}$

A function $X \colon E \to \mathbb{R}_0^+$ is called S–T **flow** if:

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

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



[Finnrind, CC BY-SA 3.0, via Wikimedia Commons]

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

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

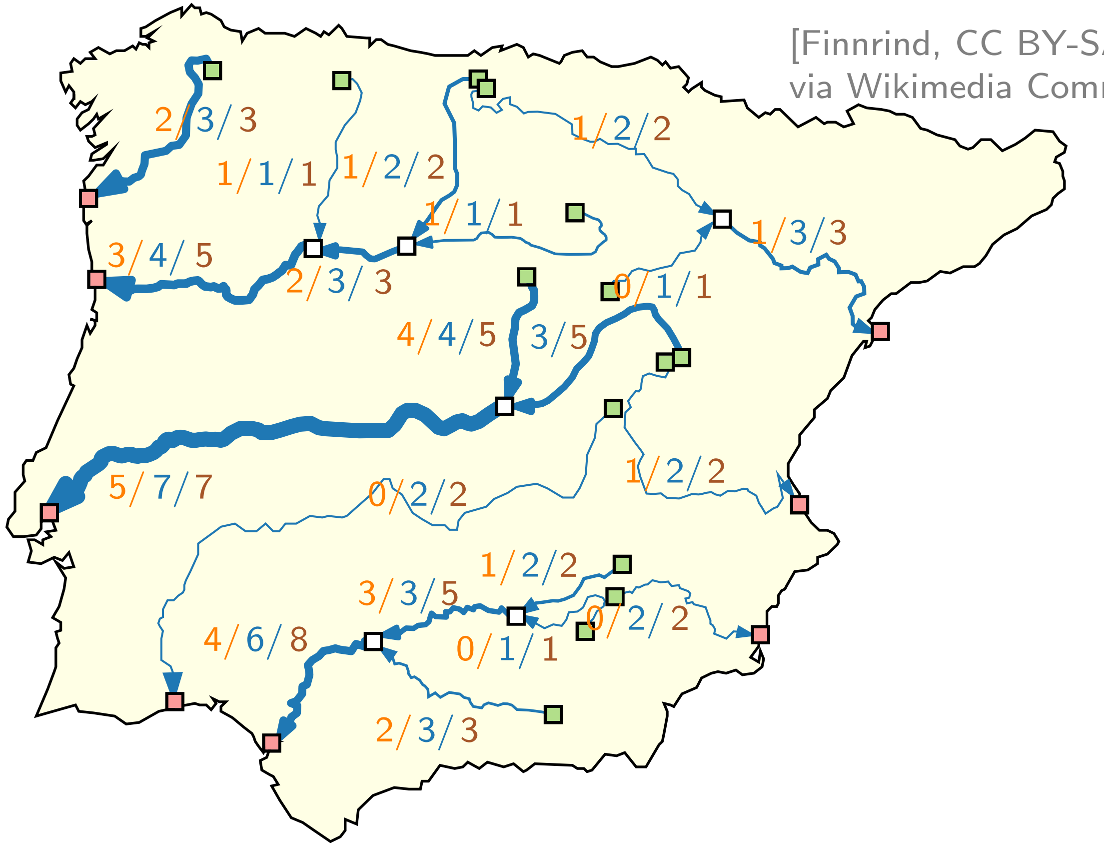
A function $X \colon E \to \mathbb{R}_0^+$ is called s–t **flow** if:

$$0 \le X(i,j) \le u(i,j) \qquad \forall (i,j) \in E$$

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

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 $(G = (V, E); S, T; \ell; u)$ with

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

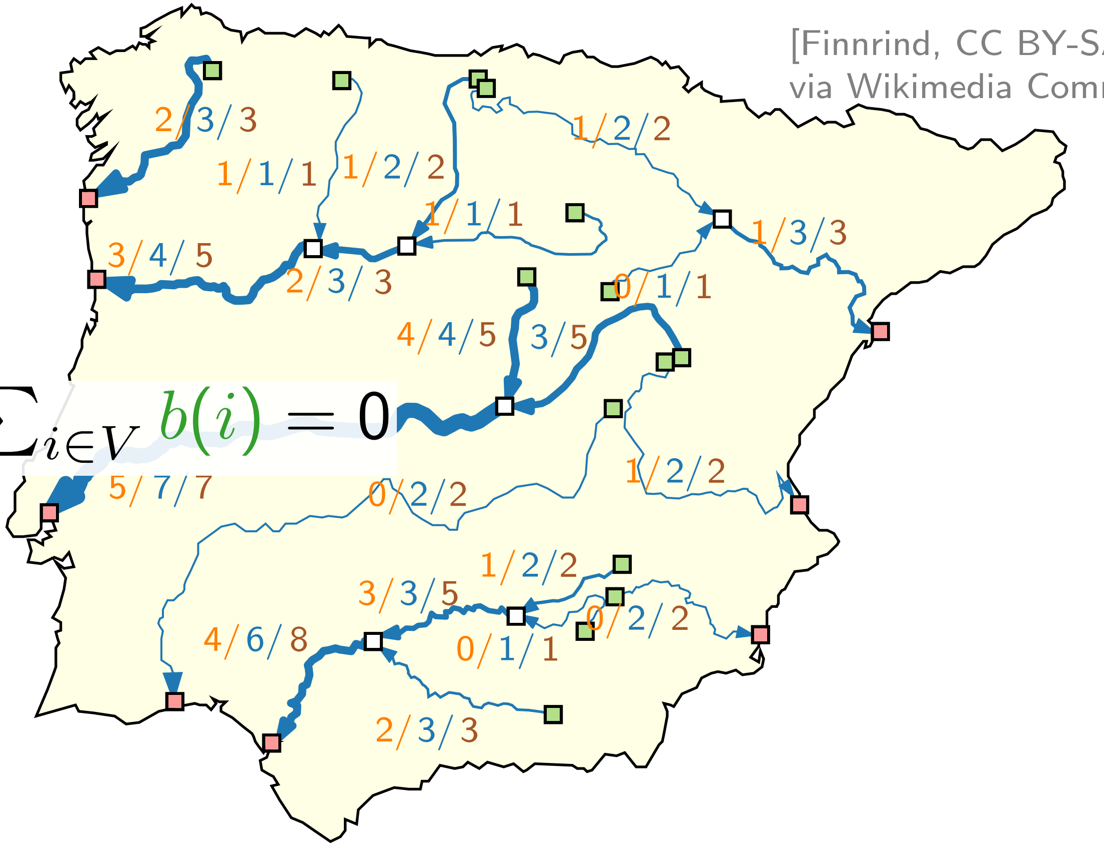
A function $X \colon E \to \mathbb{R}_0^+$ is called S–T **flow** if:

$$\ell(i,j) \le X(i,j) \le u(i,j) \qquad \forall (i,j) \in E$$

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

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



[Finnrind, CC BY-SA 3.0, via Wikimedia Commons]

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

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

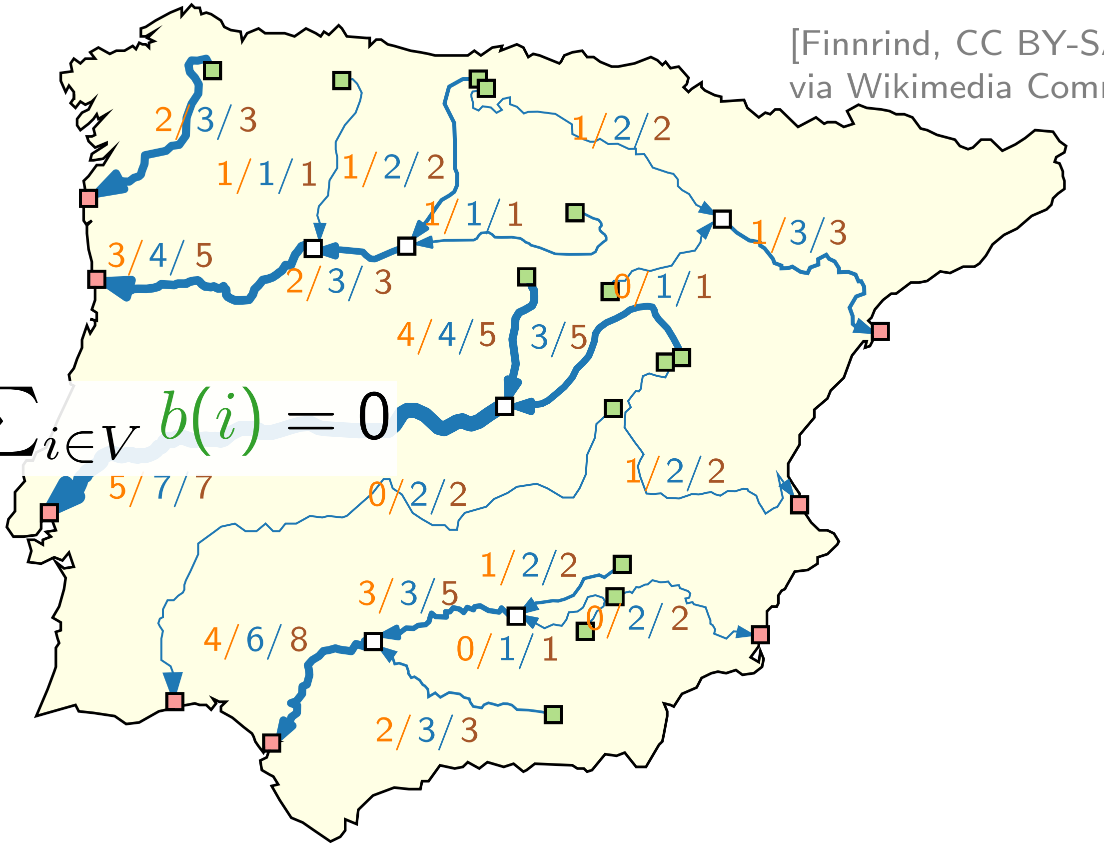
A function $X \colon E \to \mathbb{R}_0^+$ is called **valid flow**, if:

$$\ell(i,j) \le X(i,j) \le u(i,j) \qquad \forall (i,j) \in E$$

$$\sum_{(i,j) \in E} X(i,j) - \sum_{(j,i) \in E} X(j,i) = b(i) \qquad \forall i \in V$$

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* $b \colon V \to \mathbb{R}$ with $\sum_{i \in V} b(i) = 0$
- edge *lower bound* $\ell \colon E \to \mathbb{R}_0^+$
- edge *capacity* $u \colon E \to \mathbb{R}_0^+ \cup \{\infty\}$

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

$$\ell(i,j) \le X(i,j) \le u(i,j) \qquad \forall (i,j) \in E$$

$$\sum_{(i,j) \in E} X(i,j) - \sum_{(j,i) \in E} X(j,i) = b(i) \qquad \forall i \in V$$

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

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

Strongly Polynomial Algorithms

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

S(n, m)	=	O(m + n log n)	Fredman and Tarjan [1984]
S(n, m, C)	=	O(Min (m + n$\sqrt{\log C}$), (m log log C))	Ahuja, Mehlhorn, Orlin and Tarjan [1990]; Van Emde Boas, Kaas and Zijlstra[1977]
M(n, m)	=	O(min (nm + $n^{2+\varepsilon}$, nm log n) where ε is any fixed constant.	King, Rao, and Tarjan [1991]
M(n, m, U)	=	O(nm log ($\frac{n}{m}\sqrt{\log U}$ + 2))	Ahuja, Orlin and Tarjan [1989]

[Orlin 1991]

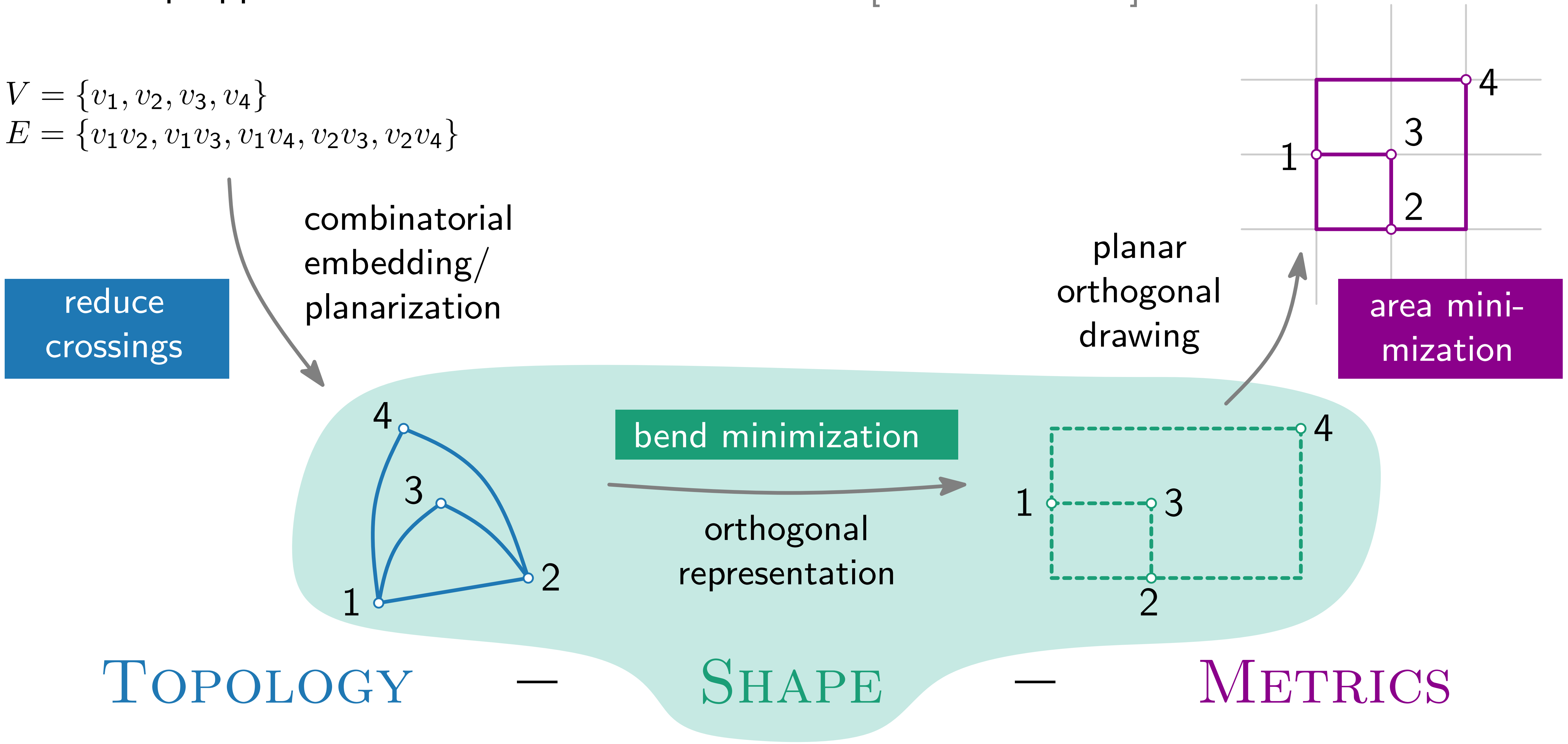
Theorem. [Orlin 1991]
The minimum cost flow problem can be solved in $O(n^2 \log^2 n + m^2 \log n)$ time.

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

Topology – Shape – Metrics

Three-step approach: [Tamassia 1987]

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



Bend Minimization with Given Embedding

Geometric orthogonal 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 orthogonal 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 orthogonal 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 $= \measuredangle \frac{\pi}{2}$
- vertices $\xrightarrow{\measuredangle}$ faces ($\# \measuredangle \frac{\pi}{2}$ per face)
- faces $\xrightarrow{\measuredangle}$ 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 δ_1 is reversed and inverted δ_2.

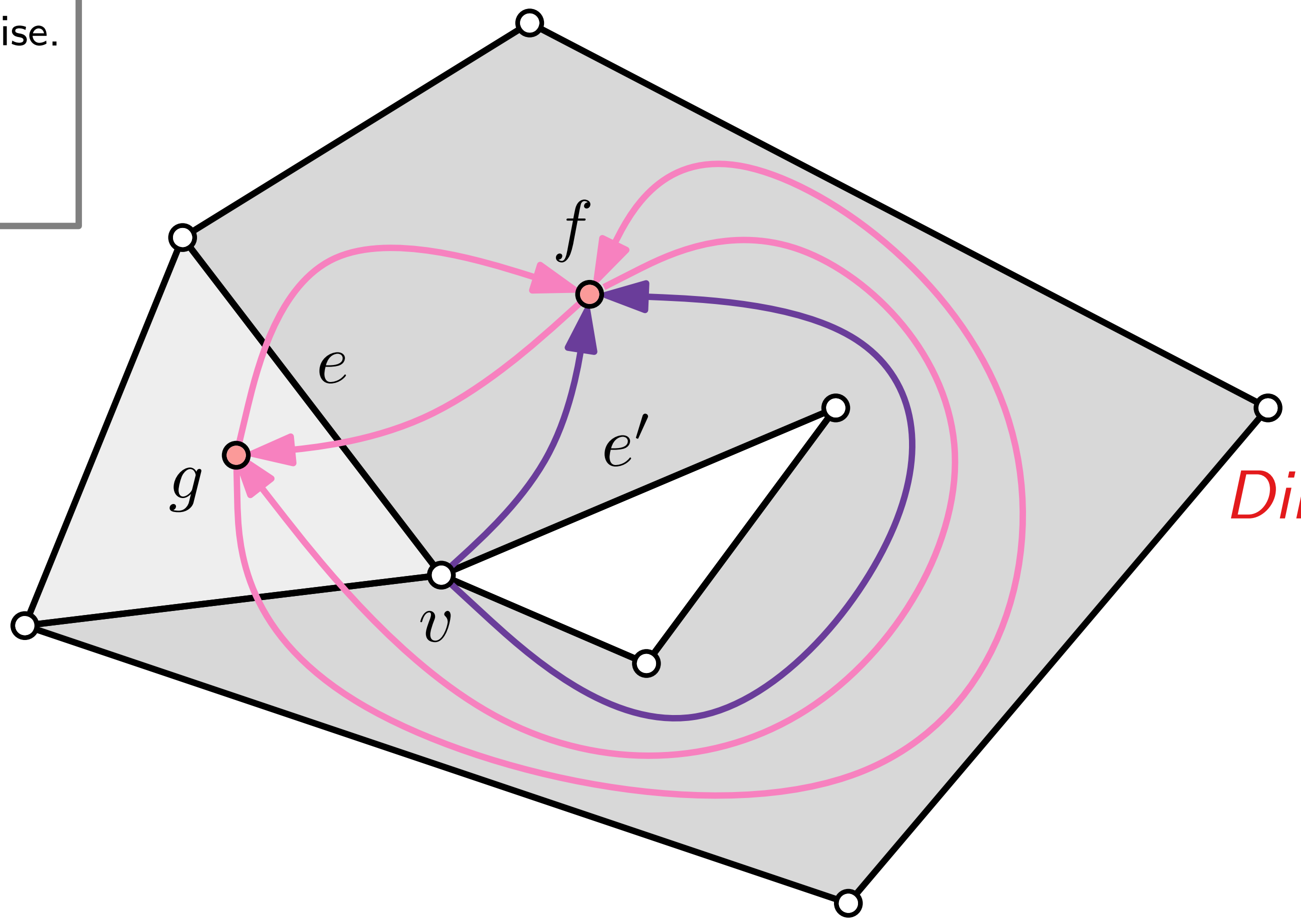
(H3) For each **face** f it holds that:

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

(H4) For each **vertex** v the sum of incident angles is 2π.

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

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



Directed multigraph!

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 δ_1 is reversed and inverted δ_2.

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

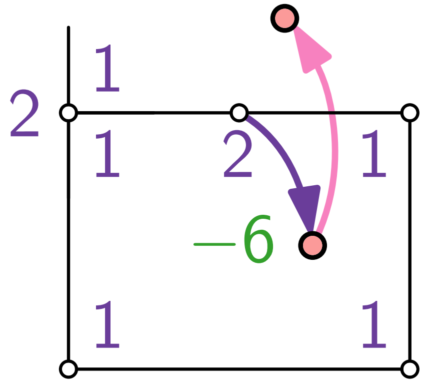
$$\sum_{r \in H(f)} C(r) = \begin{cases} -4 & \text{if } f = f_0 \\ +4 & \text{otherwise.} \end{cases}$$

(H4) For each **vertex** v the sum of incident angles is 2π.

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)

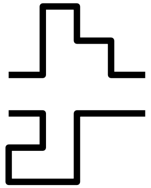


$\forall (v, f) \in E', v \in V, f \in F$ $\quad \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)$

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

$\forall (f, g) \in E', f, g \in F$ $\quad \ell(f, g) := 0 \leq X(f, g) \leq \infty =: u(f, g)$

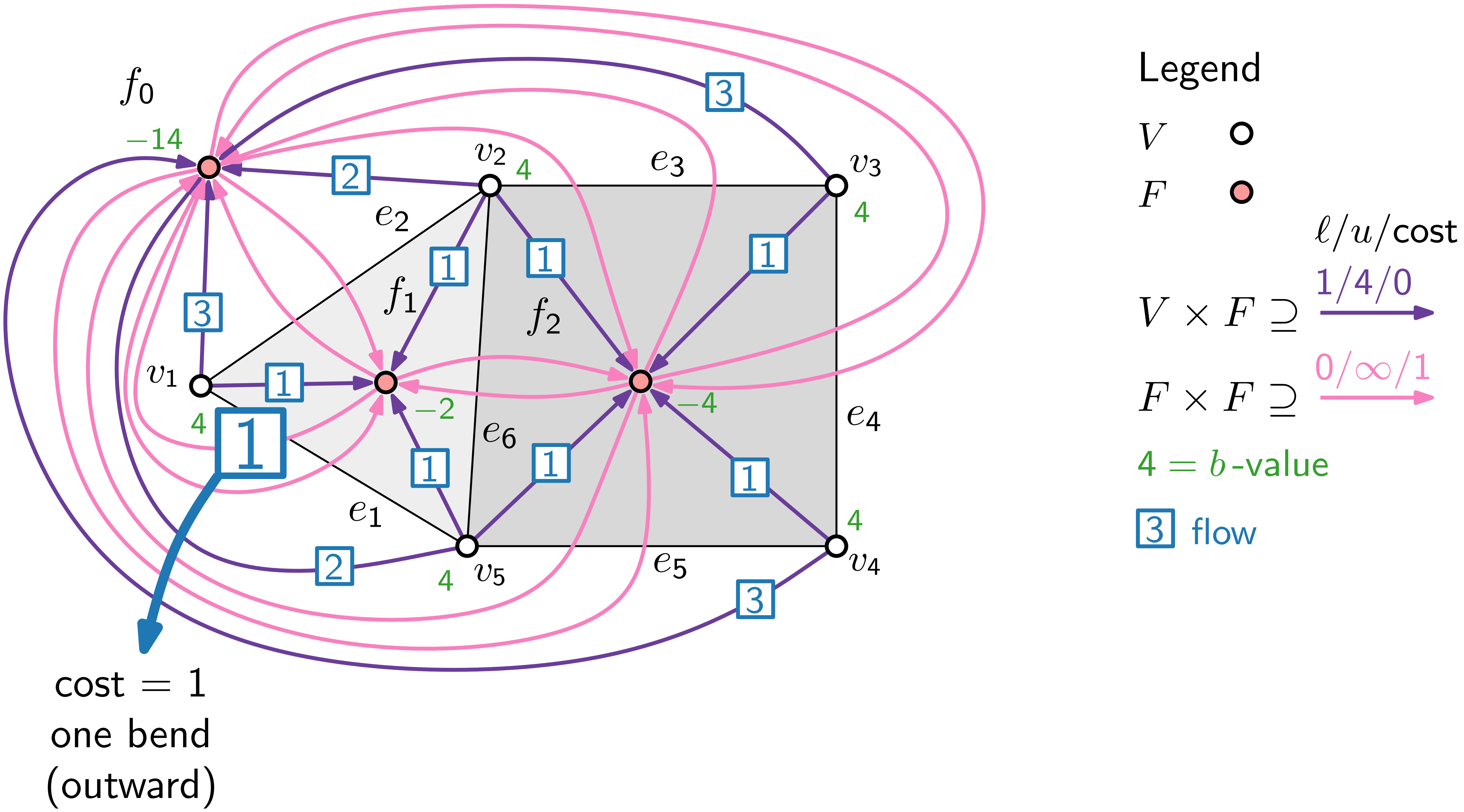
$\text{cost}(f, g) = 1$



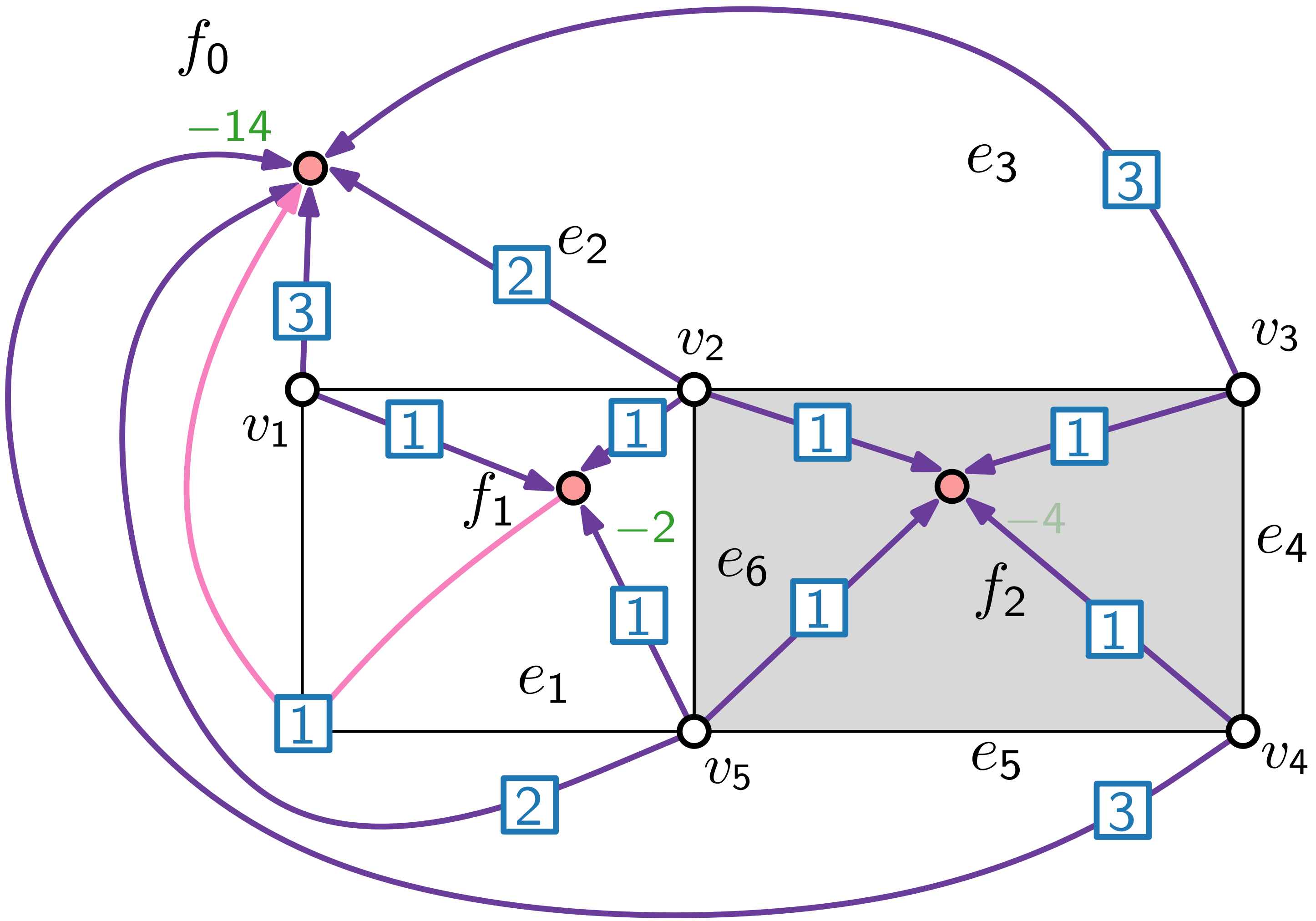
We model only the *number* of bends. Why is it enough?

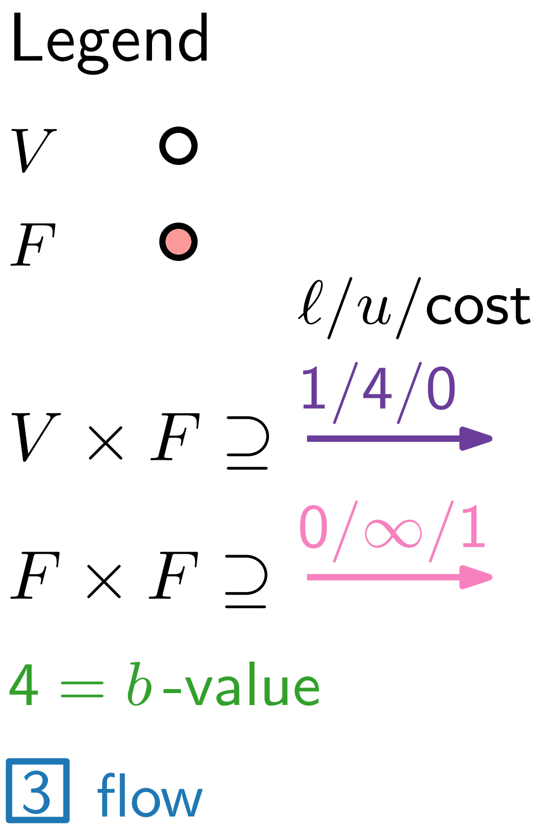
→ *Exercise!*

Flow Network Example



Flow Network Example





Bend Minimization – Result

Theorem. [Tamassia '87]
A plane graph (G, F, f_0) has a valid orthogonal representation $H(G)$ with k bends. $\Leftrightarrow$
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.
- Show properties (H1)–(H4).

(H1) H(G) matches F, f_0 ✓

(H2) Bend order inverted and reversed on opposite sides ✓

(H3) Angle sum of $f = \pm 4$ ✓ $\rightarrow$ *Exercise.*

(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 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 2π.

Bend Minimization – Result

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

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

Proof.

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

- Define flow $X \colon E' \to \mathbb{R}_0^+$.
- Show that X is a valid flow and has cost k.

(N1) $X(vf) = 1/2/3/4$ ✓

(N2) $X((fg)_e) = |\delta|_0$, where (e, δ, x) describes edge e in $H(f)$ ✓

(N3) capacities, deficit/demand coverage ✓

(N4) cost $= k$ ✓

Bend Minimization – Remarks

- The theorem implies that the combinatorial orthogonal bend minimization problem for plane graphs can be solved using an algorithm for min-cost flow.

Theorem. [Garg & Tamassia 1996]
The min-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 min-cost flow problem for planar graphs with bounded costs and face sizes can be solved in $O(n^{3/2})$ time.

Theorem. [Garg & Tamassia 2001]
Bend minimization without given combinatorial embedding is NP-hard.

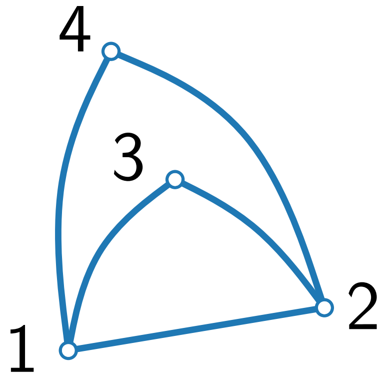
Topology – Shape – Metrics

Three-step approach: [Tamassia 1987]

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

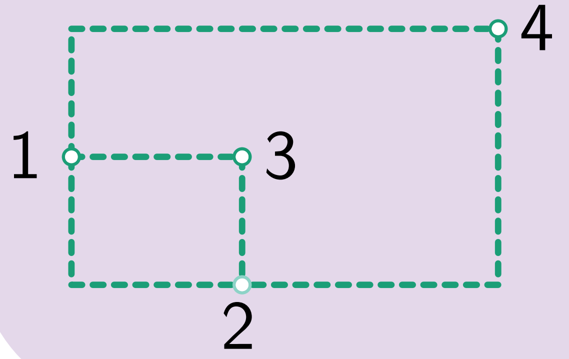
combinatorial embedding/ planarization

reduce crossings

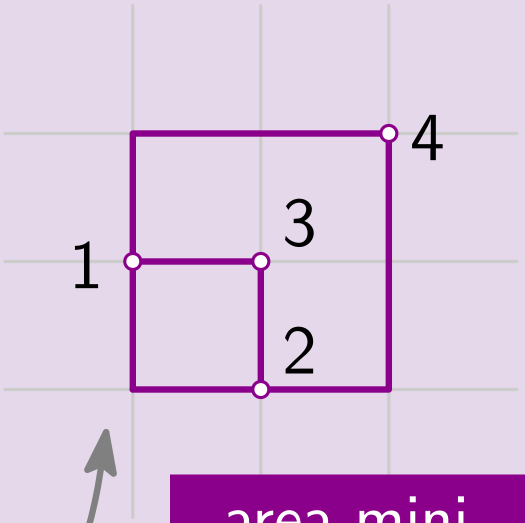


bend minimization

orthogonal representation



planar orthogonal drawing



area minimization

TOPOLOGY — SHAPE — METRICS

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.

$\rightarrow$ guarantees
- minimum total edge length
- minimum area

Properties.

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

Idea.

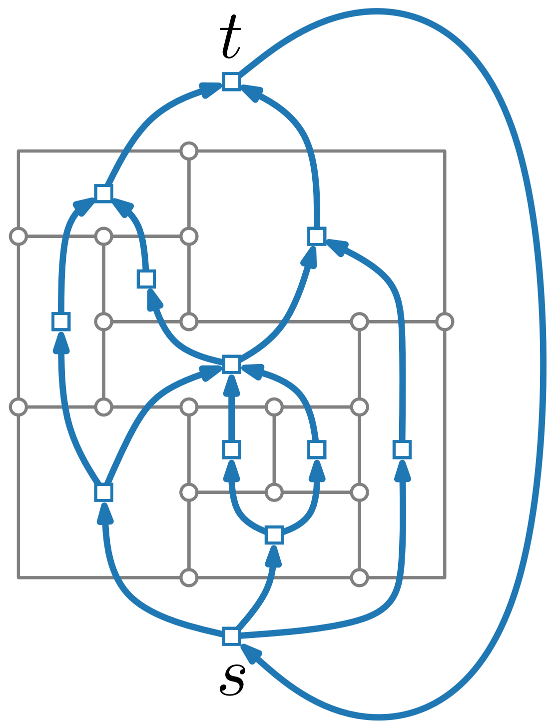
- Formulate flow network for horizontal/vertical compaction

Flow Network for Edge-Length Assignment

Definition.

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

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

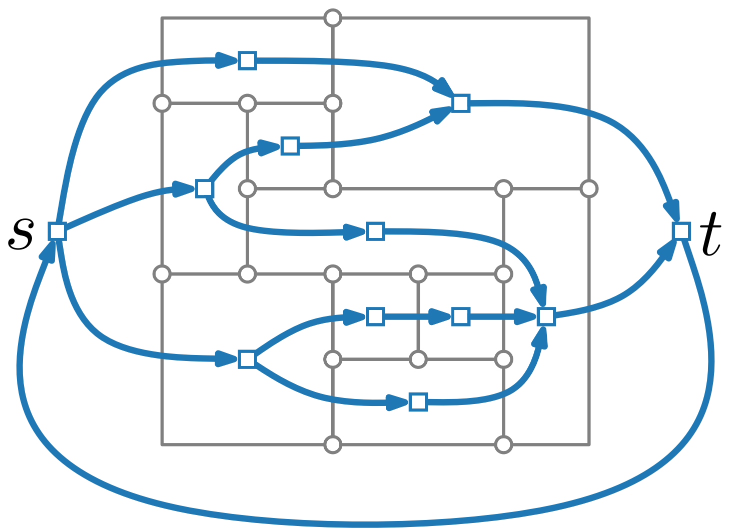


Flow Network for Edge-Length Assignment

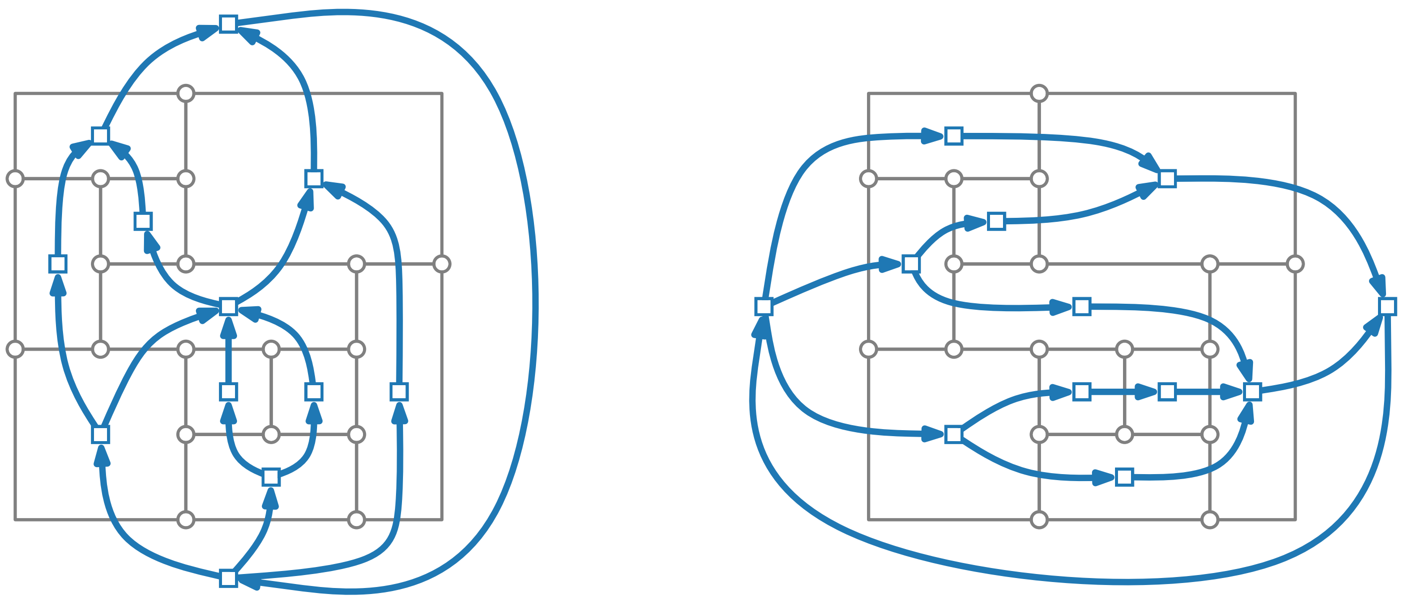
Definition.

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

- $W_{\text{ver}} = F \setminus \{f_0\} \cup \{s, t\}$ □
- $E_{\text{ver}} = \{(f, g) \mid f, g$ share a *vertical* segment and f 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



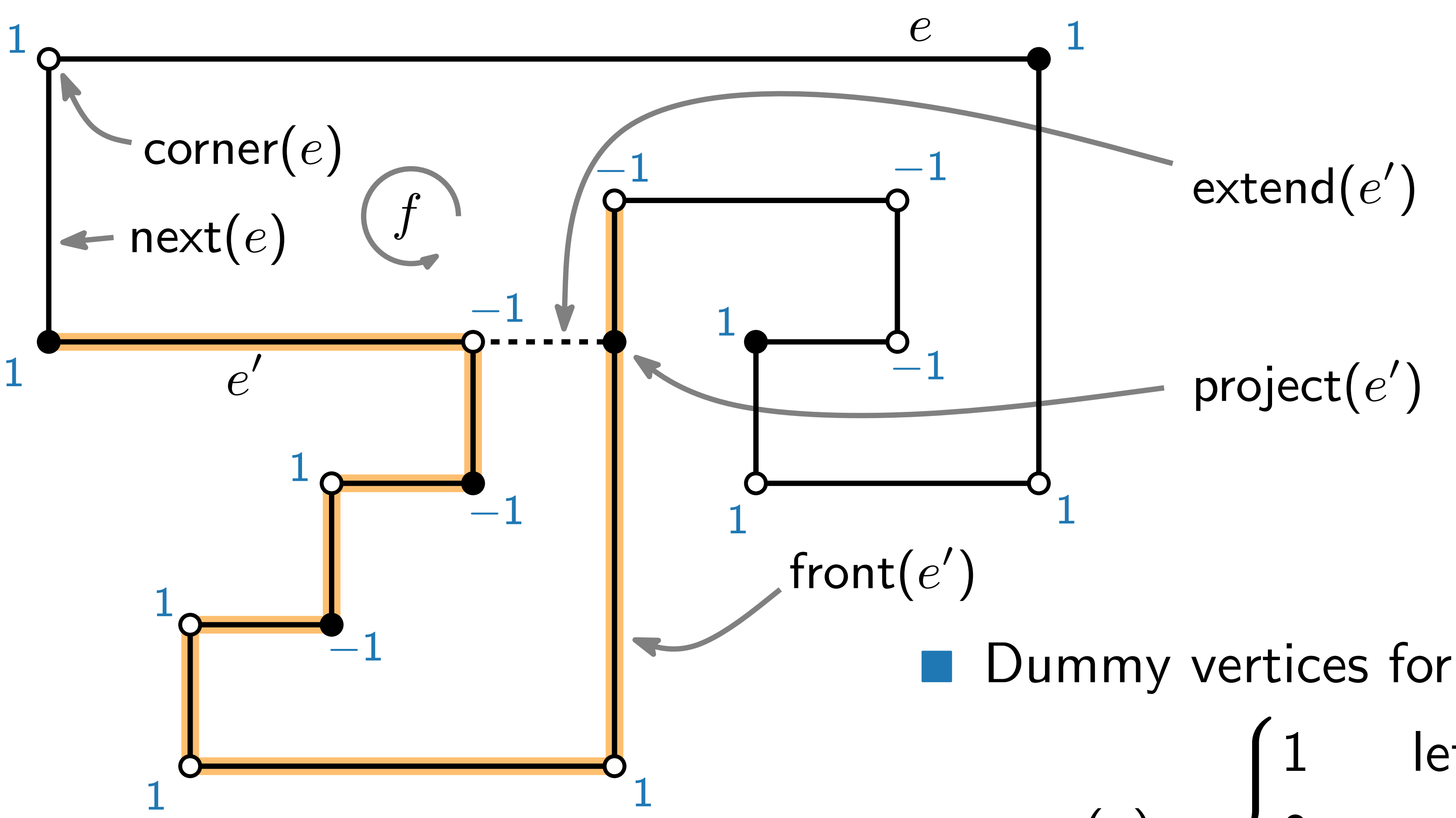
What if not all faces are rectangular?

Theorem.
A valid flow for N_{hor} and N_{ver} exists $\Leftrightarrow$ corresponding edge lengths induce an orthogonal drawing.

What values of the drawing do the following quantities represent?

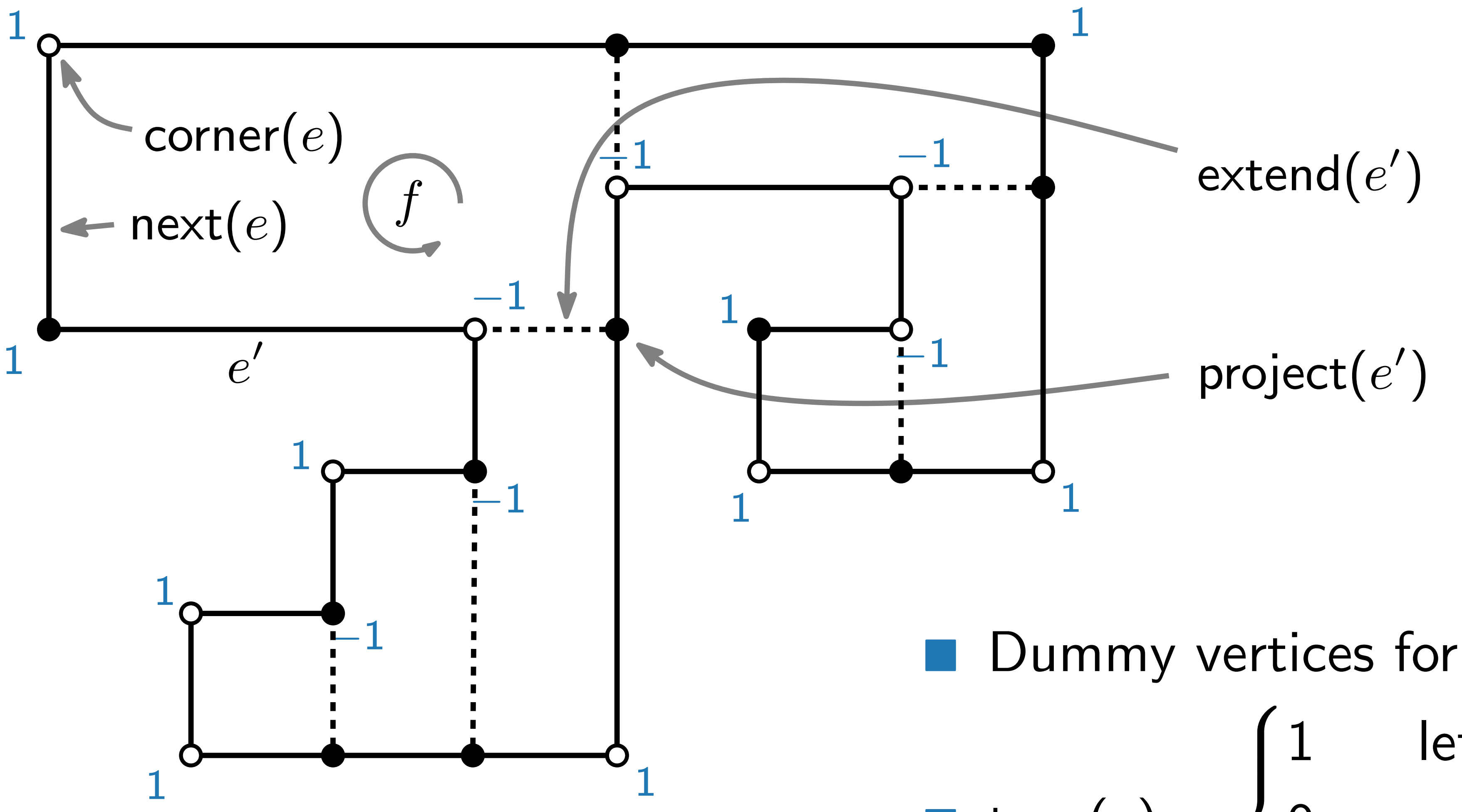
- $|X_{\text{hor}}(t,s)|$ and $|X_{\text{ver}}(t,s)|$? width and height of the drawing
- $\sum_{e \in E_{\text{hor}}} X_{\text{hor}}(e) + \sum_{e \in E_{\text{ver}}} X_{\text{ver}}(e)$ total edge length

Refinement of G and $H(G)$ – Inner Face



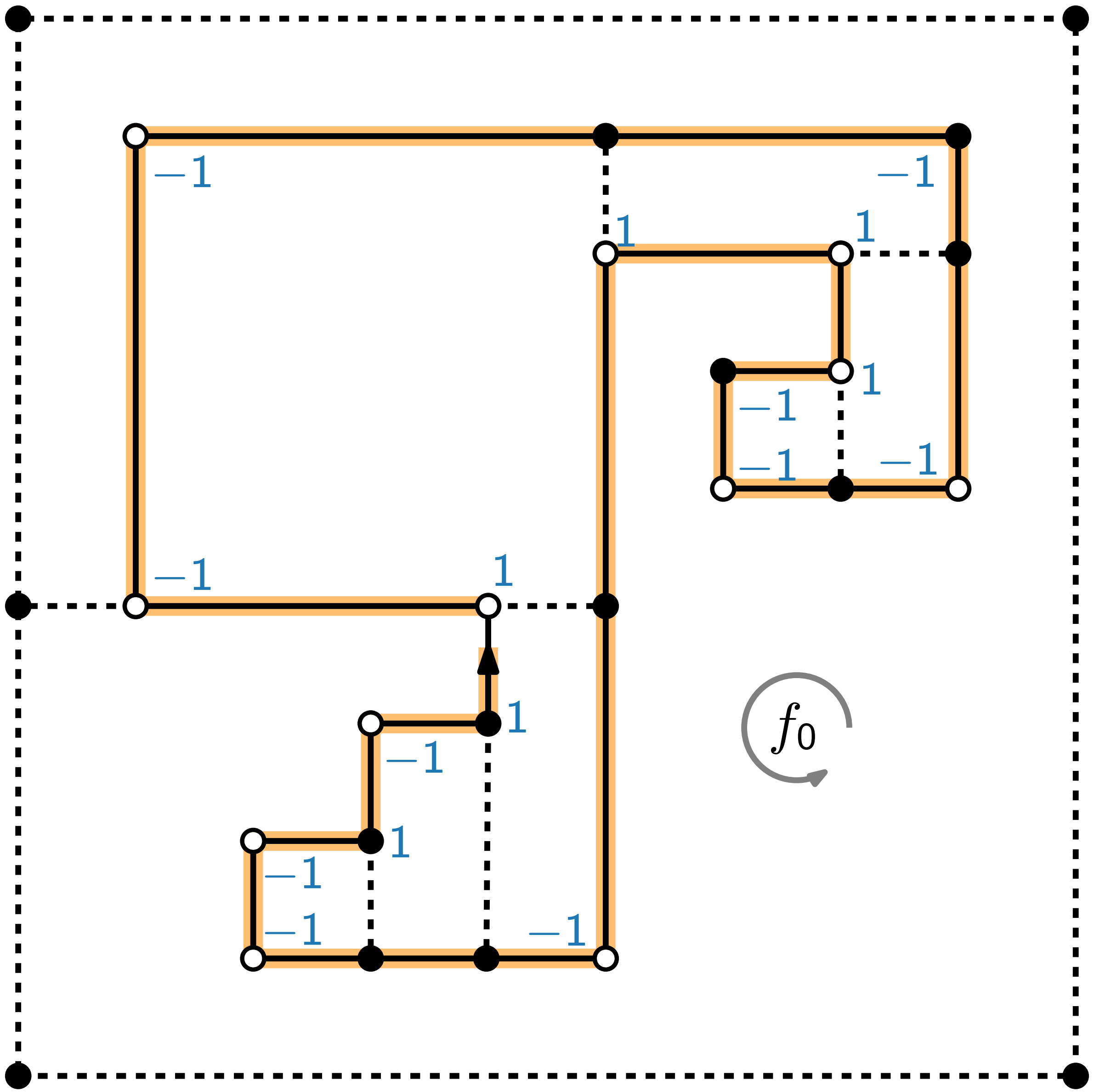
- Dummy vertices for bends
- $\text{turn}(e) = \begin{cases} 1 & \text{left turn} \\ 0 & \text{no turn} \\ -1 & \text{right turn} \end{cases}$

Refinement of G and $H(G)$ – Inner Face

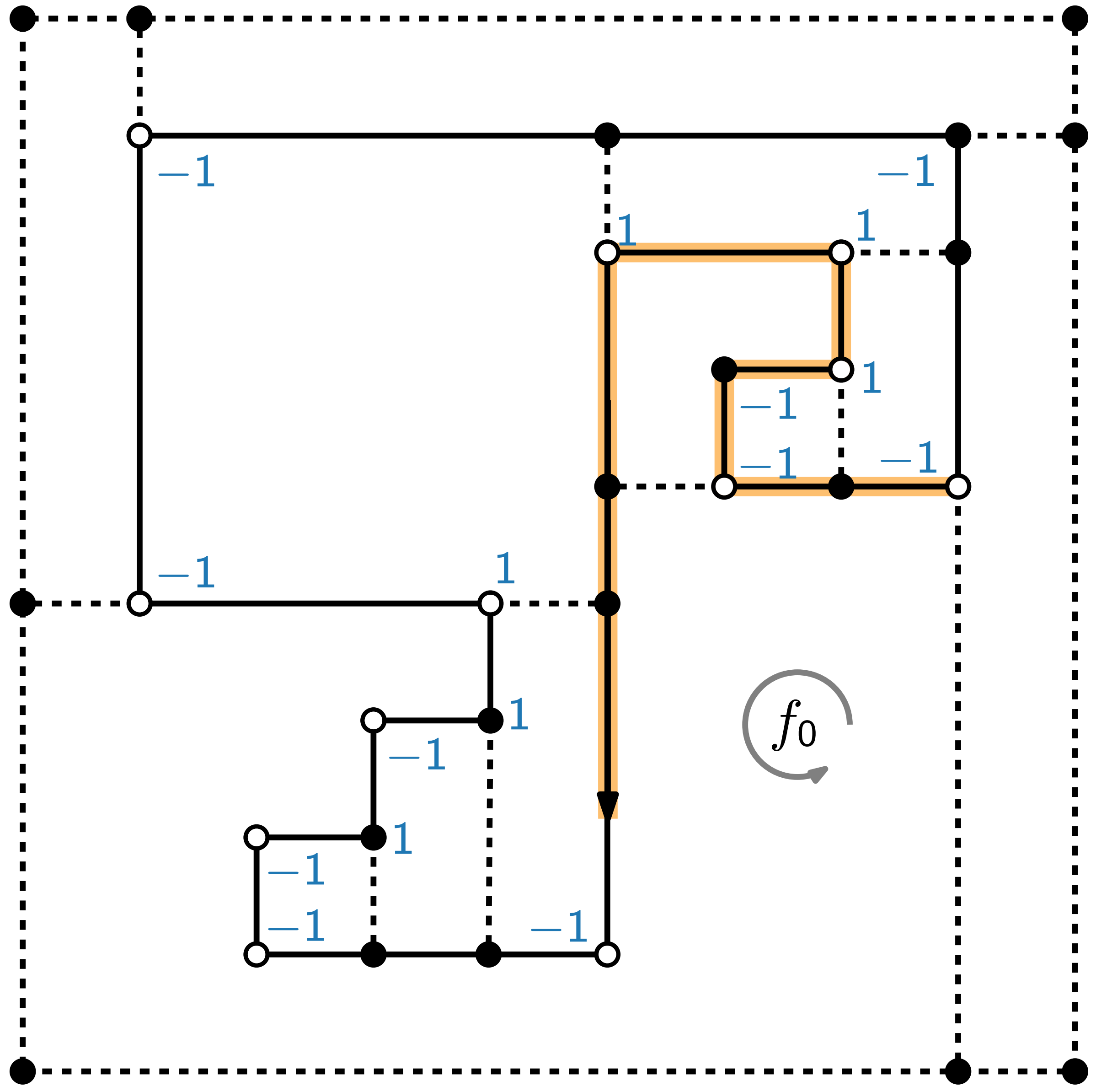


- Dummy vertices for bends
- $\text{turn}(e) = \begin{cases} 1 & \text{left turn} \\ 0 & \text{no turn} \\ -1 & \text{right turn} \end{cases}$

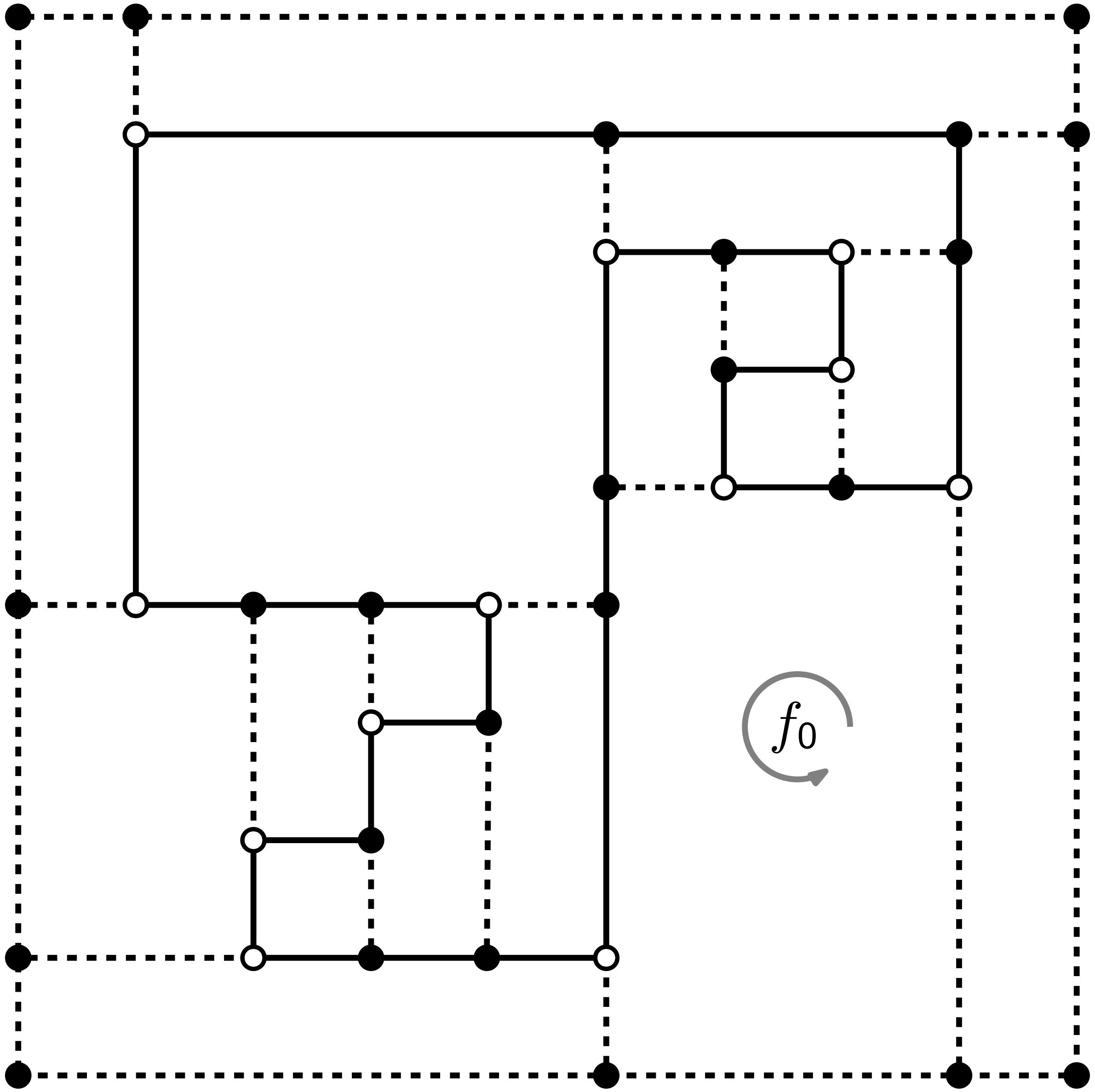
Refinement of G and $H(G)$ – Outer Face



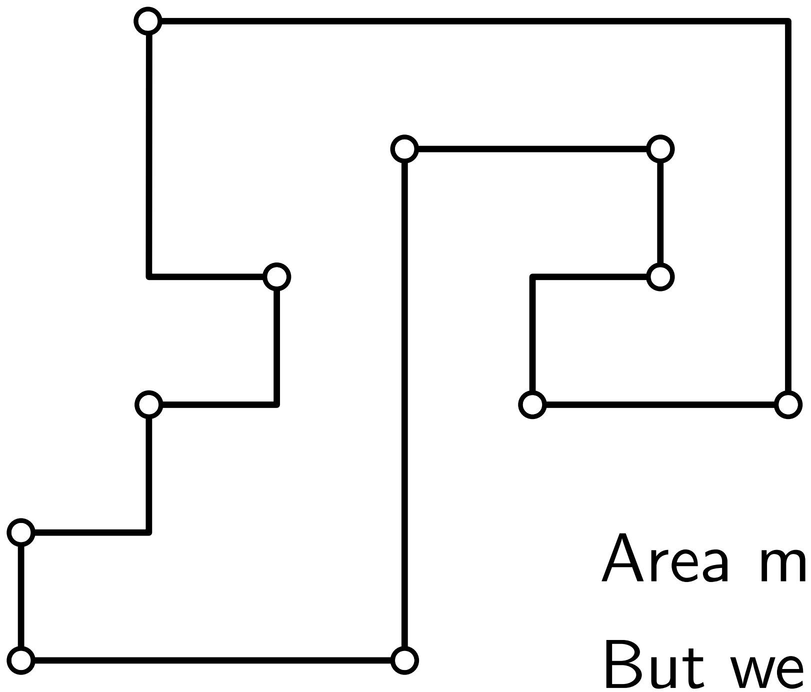
Refinement of G and $H(G)$ – Outer Face



Refinement of G and $H(G)$ – Outer Face



Refinement of G and $H(G)$ – Outer Face



Area minimized? No!

But we get bound $O((n + b)^2)$ on the area.

Theorem. [Patrignani 2001]
Compaction for a given orthogonal representation is NP-hard in general.

Theorem. [EFKSSW 2022]
Compaction is NP-hard even for orthogonal representations of *cycles*.

Compaction is NP-hard

Polynomial-time reduction from the NP-complete satisfiability problem (SAT).

In an instance of the SAT problem we have:

- set of n Boolean variables $X = \{x_1, x_2, \ldots, x_n\}$
- m clauses $C_1, C_2, \ldots, C_m$, where each clause is a disjunction of literals from X, e.g., $C_1 = x_1 \vee \neg x_2 \vee x_3$

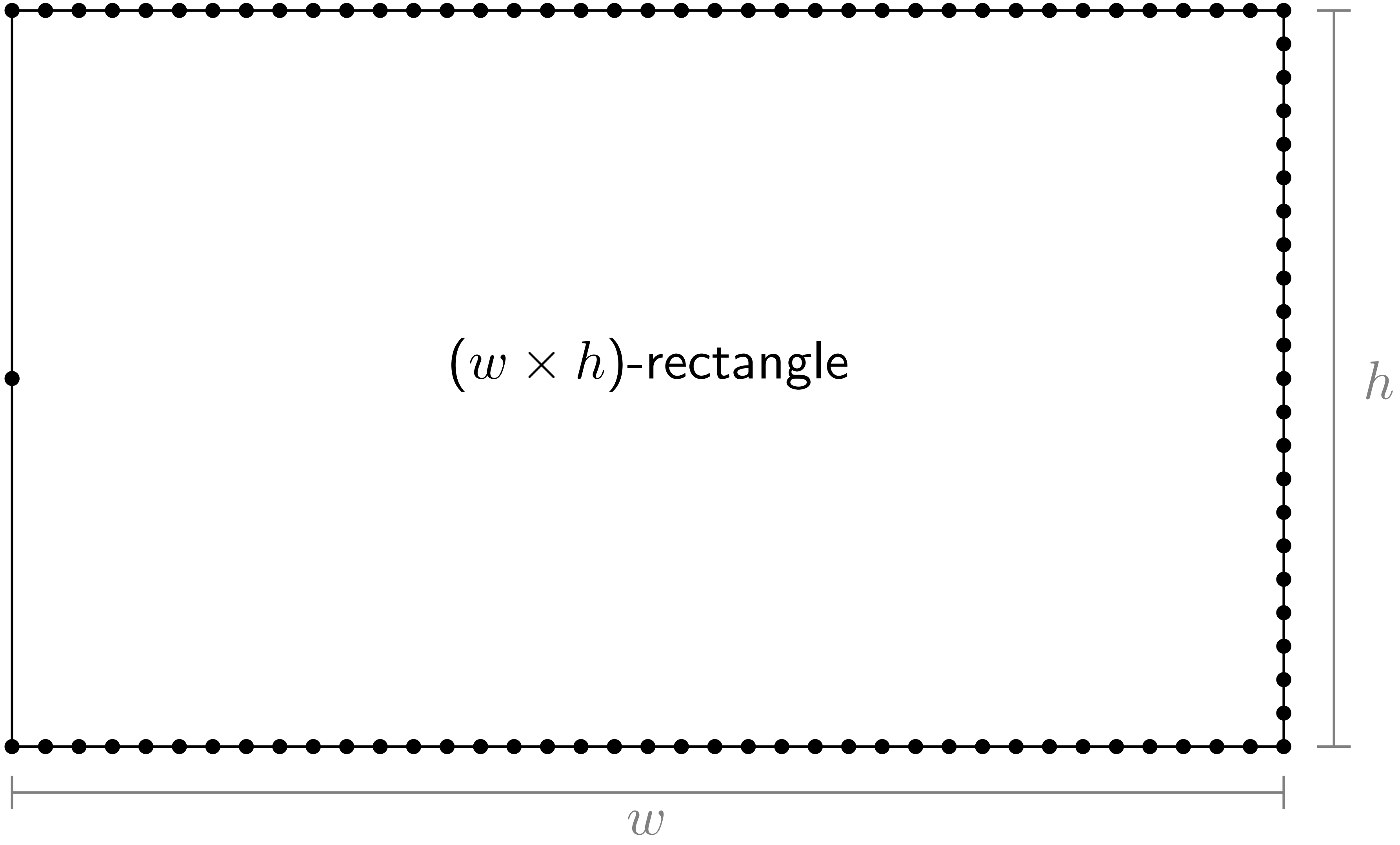
  a literal is a variable x or a negated variable $\neg x$
- Boolean formula $\Phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$

Question: Is there an assignment of truth values to the variables in X such that Φ is true?

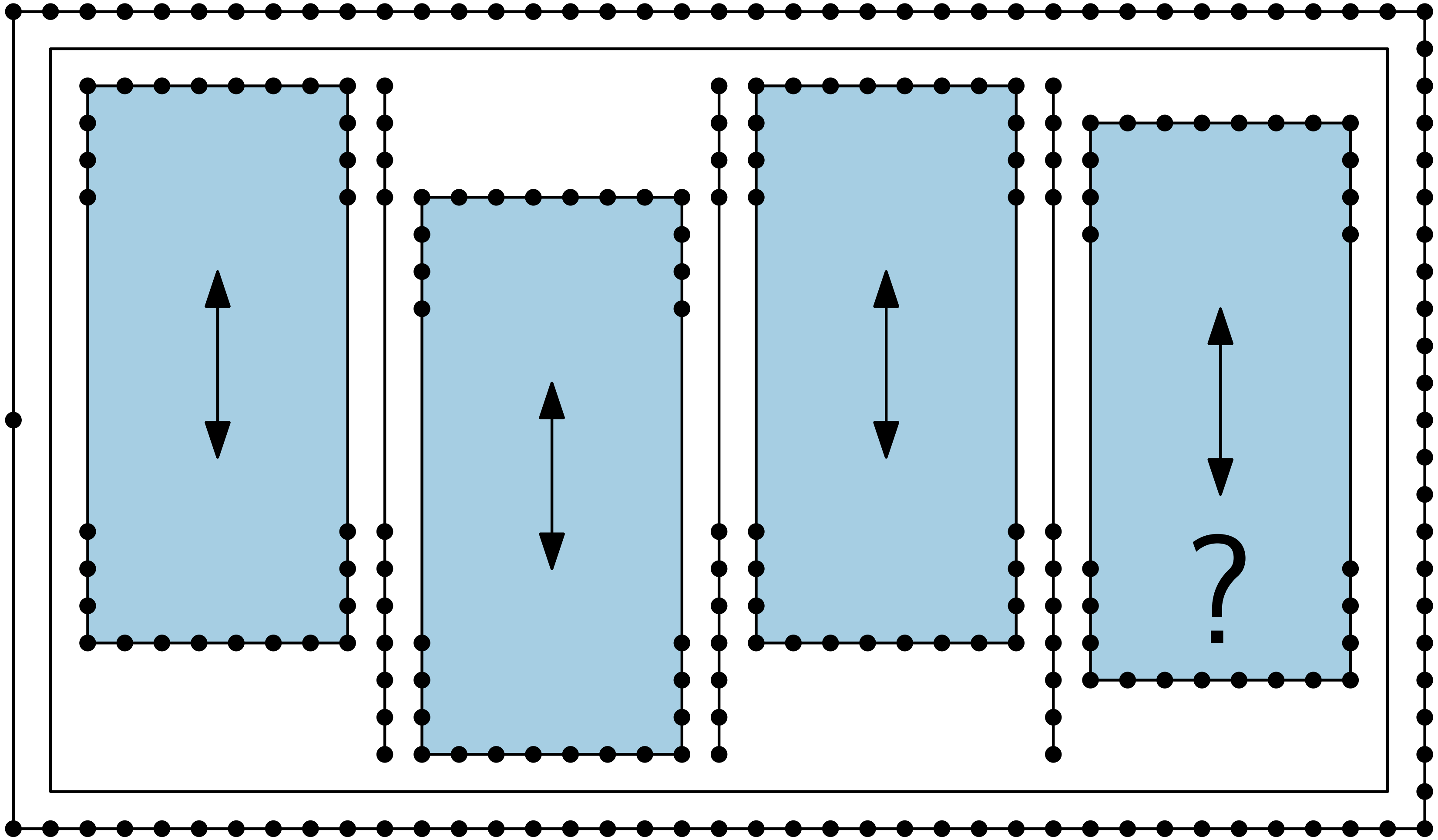
Idea of the reduction:

- Given SAT instance $\Phi \Rightarrow$ construct a plane graph G and a orthogonal description $H(G)$
- Φ is satisfiable $\Leftrightarrow$ G can be drawn w.r.t. $H(G)$ in area K for some specific number K

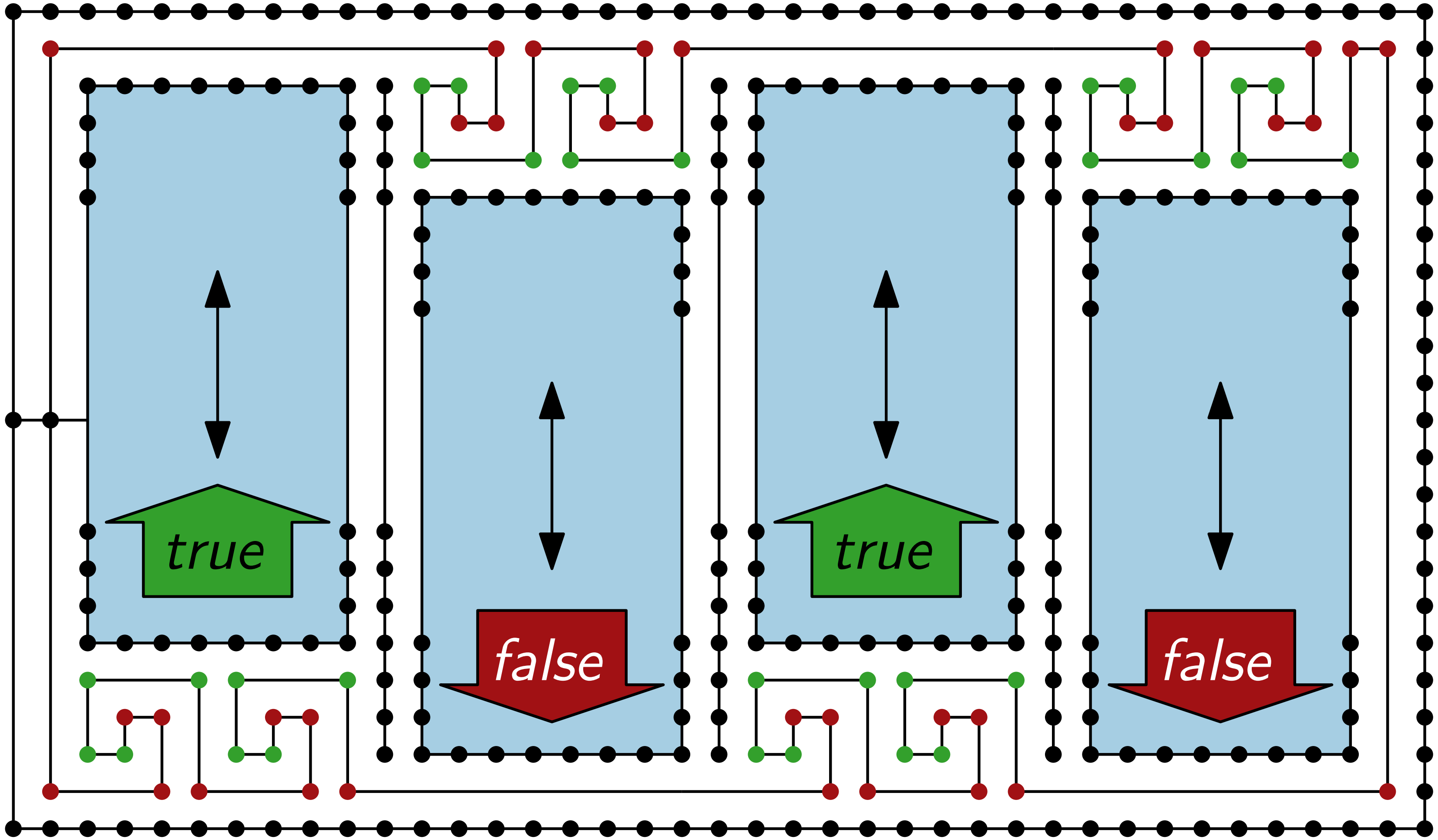
Boundary, Belt, and "Piston" Gadget



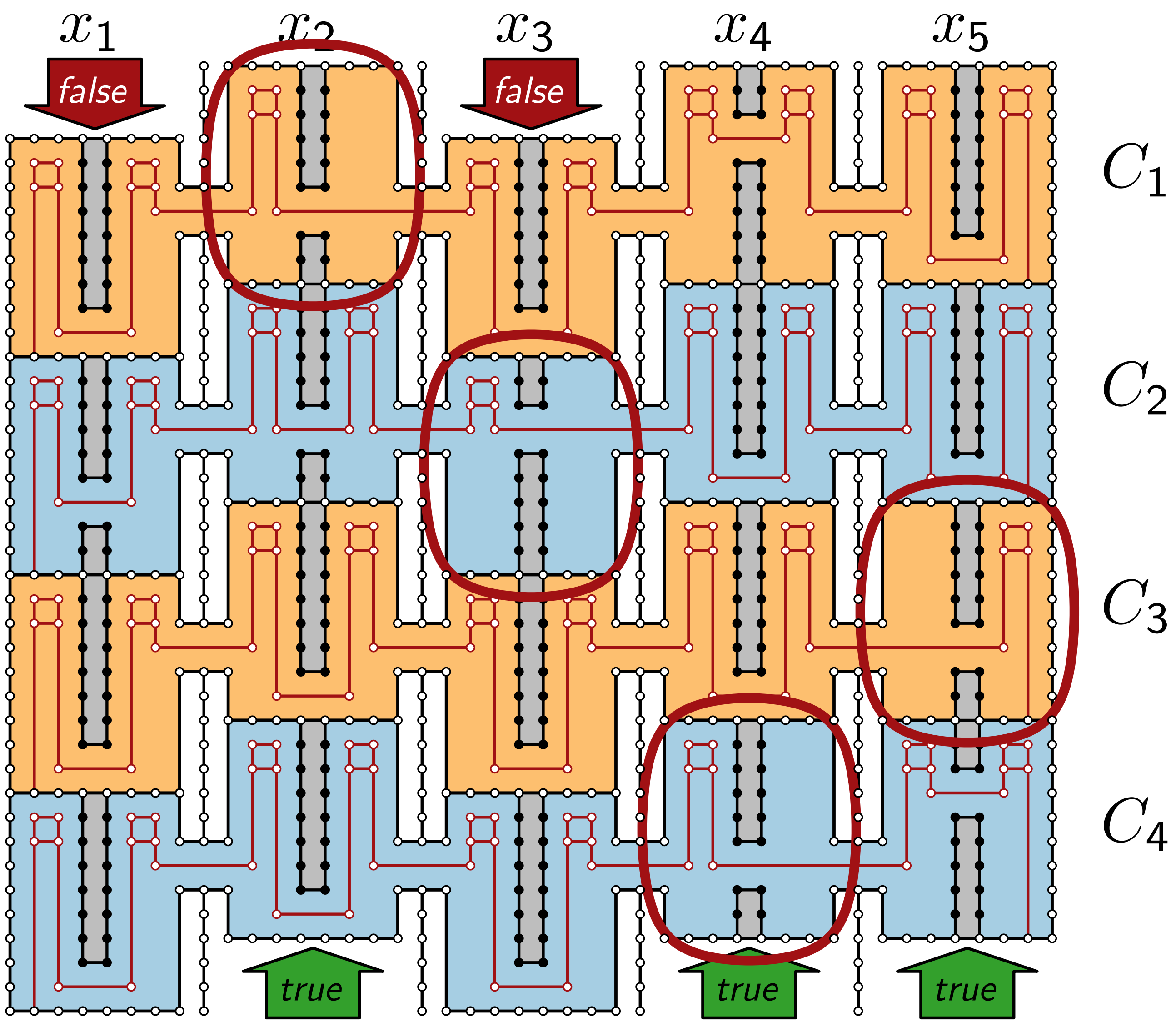
Boundary, Belt, and “Piston” Gadget



Boundary, Belt, and “Piston” Gadget

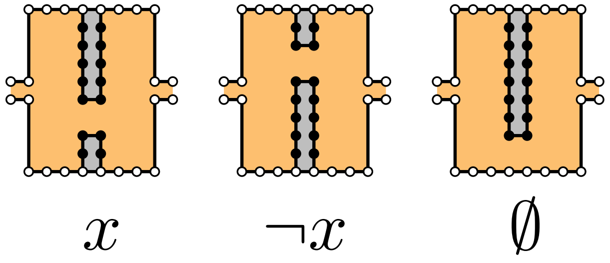


Clause Gadgets



Example:

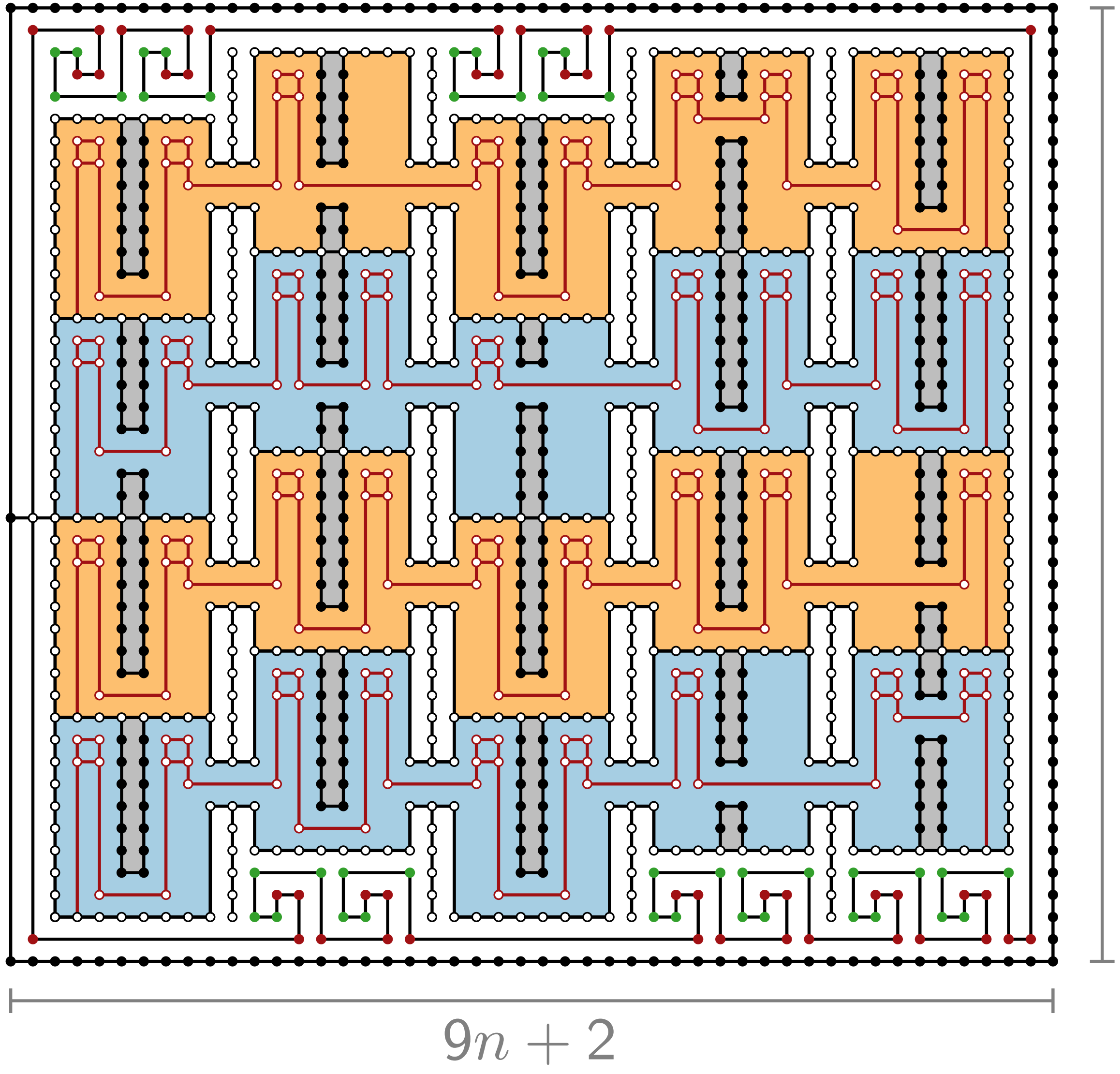
$C_1 = x_2 \vee \neg x_4$
$C_2 = x_1 \vee x_2 \vee \neg x_3$
$C_3 = x_5$
$C_4 = x_4 \vee \neg x_5$



insert $(2n - 1)$-chain through each clause

$\rightarrow$ for every clause, there needs to be ≥ 1 "gap of a literal" to be on the same height as the "tunnel" to the next literal

Complete Reduction



Pick
$K = (9n + 2) \times (9m + 7)$

Then:

G under $H(G)$ has an orthogonal drawing in area K

$\Leftrightarrow$

Φ satisfiable

□

Literature

- [GD Ch. 5] for detailed explanation
- [Tamassia 1987] "On embedding a graph in the grid with the minmum number of bends"
  Original paper on flow for bend minimization.
- [Patrignani 2001] "On the complexity of orthogonal compaction"
  NP-hardness proof for orthogonal representation of planar max-degree-4 graphs.
- [Evans, Fleszar, Kindermann, Saeedi, Shin, Wolff 2022]
  "Minimum rectilinear polygons for given angle sequences"
  NP-hardness proof for compaction of cycles.