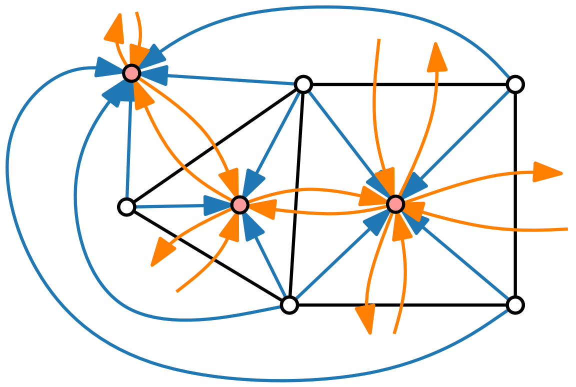
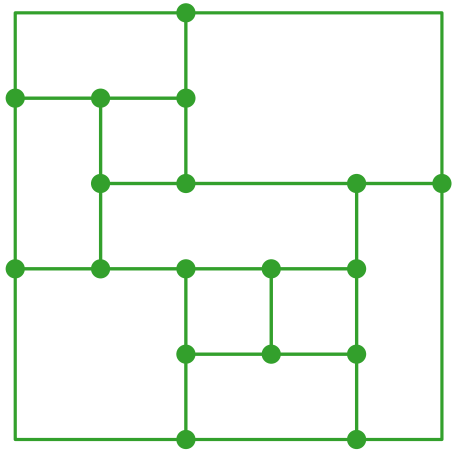
Visualisation of graphs

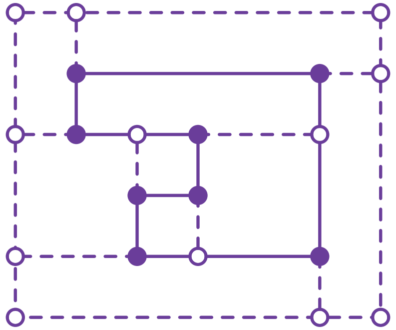
Orthogonal layouts

Flow methods

Jonathan Klawitter · Summer semester 2020

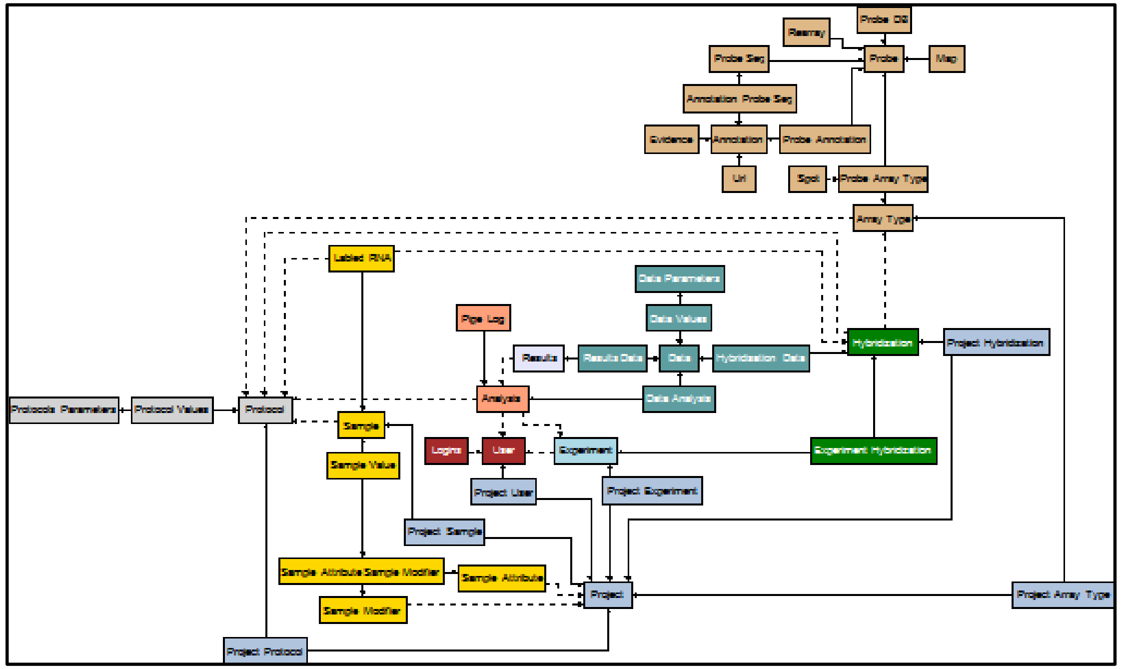




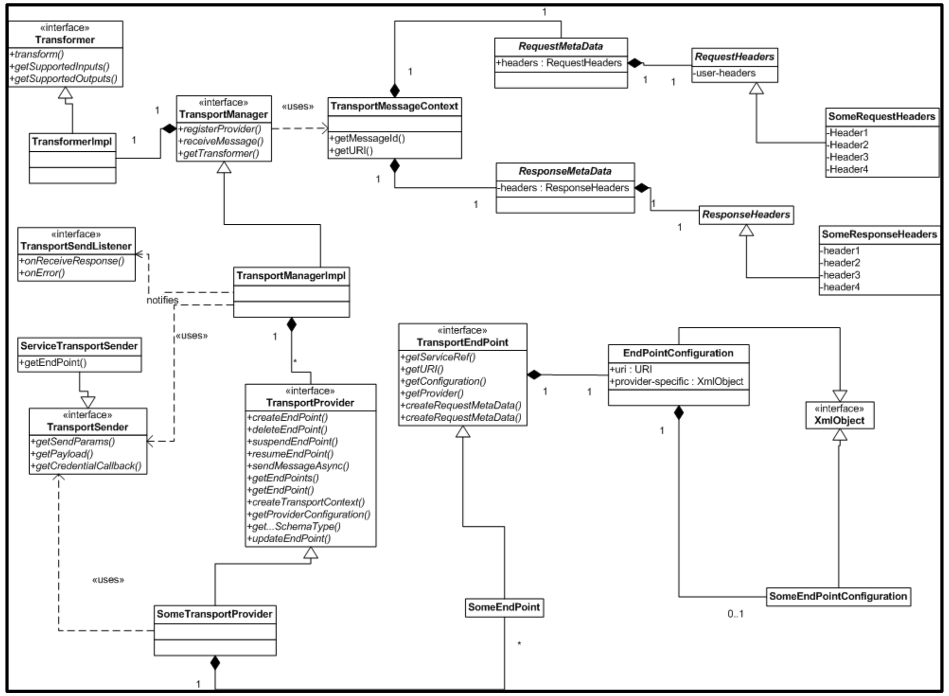


Orthogonal layout – applications

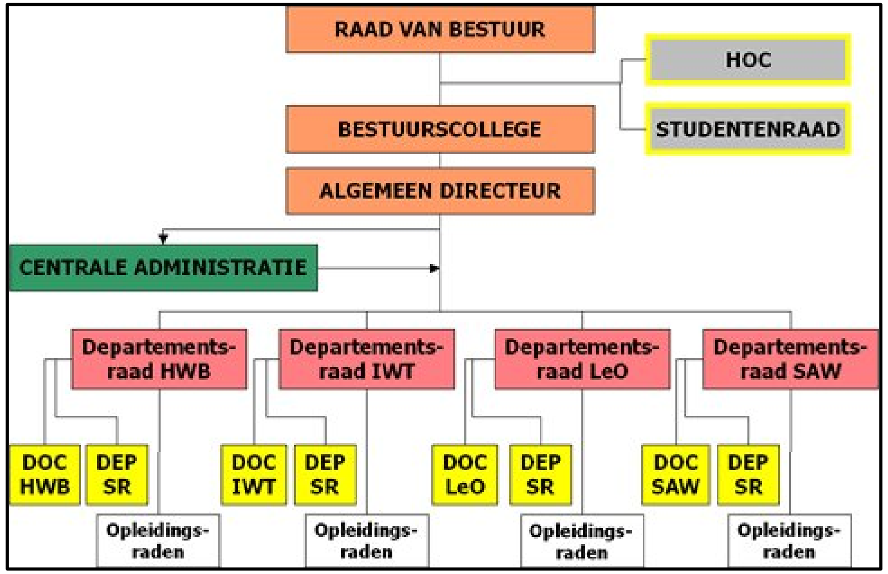
ER diagram in OGDF



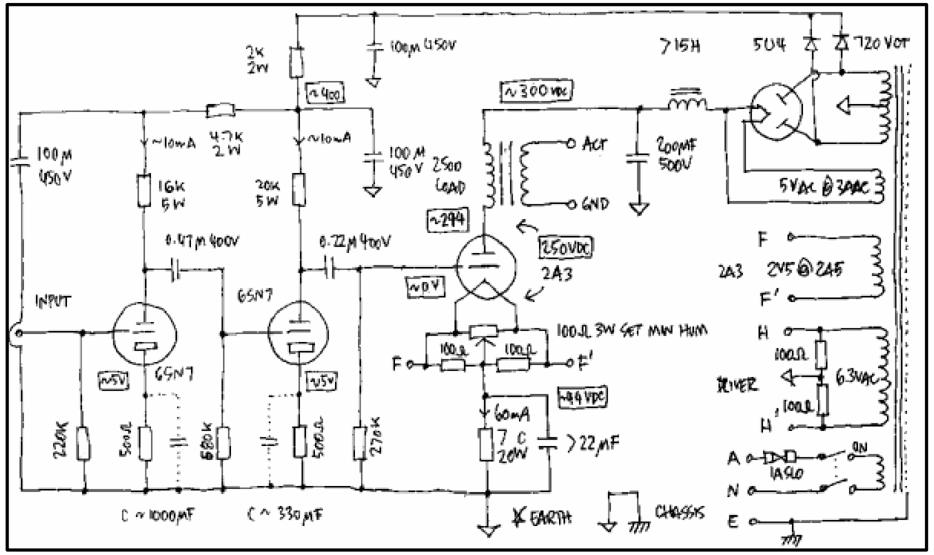
UML diagram by Oracle



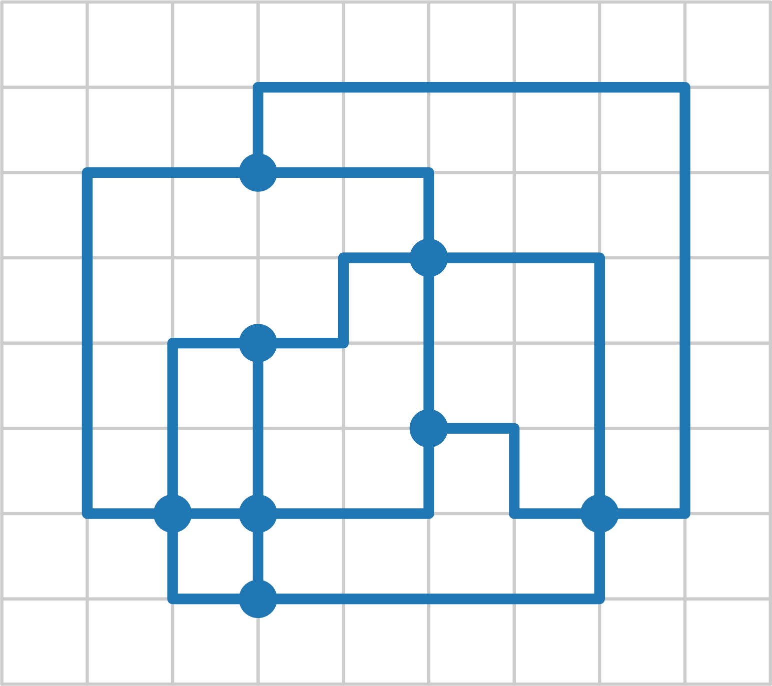
Organigram of HS Limburg



Circuit diagram by Jeff Atwood



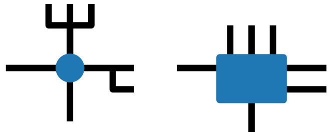
Orthogonal layout – definition



Definition.
A drawing Γ of a graph $G = (V, E)$ is called **orthogonal** if

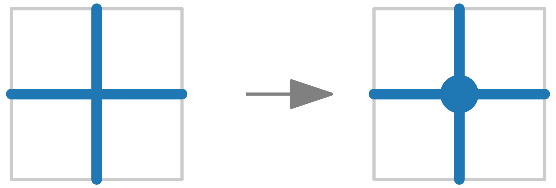
- veritices are drawn as points on a grid,
- each edge is represented as a sequence of alternating horizontal and vertical segments, and
- pairs of edges are disjoint or cross orthogonally.

Observations.

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

Planarisation.

- Fix embedding
- Crossings become vertices



Aesthetic criteria.

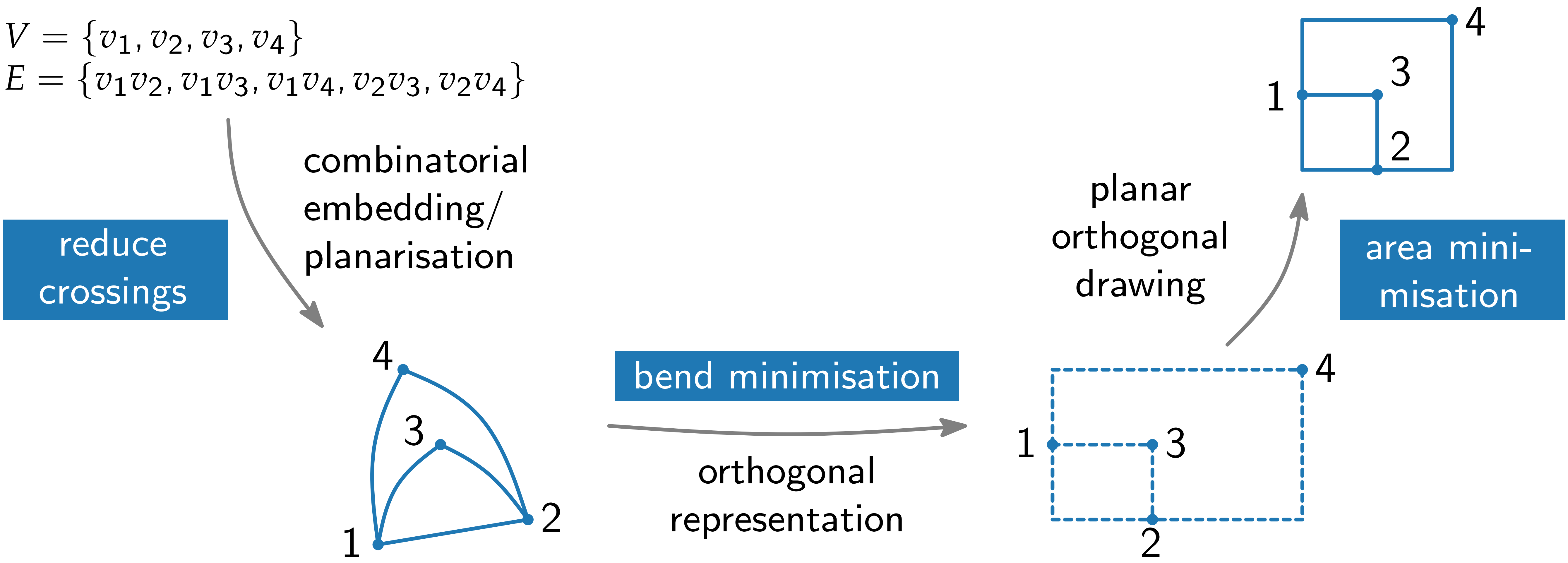
- Number of bends
- Length of edges
- Width, height, area
- Monotonicity of edges
- ...

Topology - Shape - Metrics

Three-step approach:

[Tam87]

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

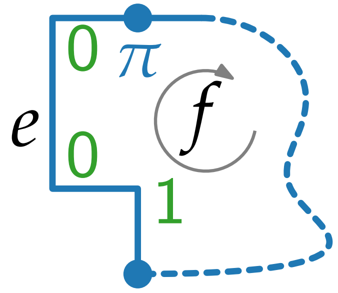


Orthogonal representation

Idea.
Describe orthogonal drawing 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, δ, α) where
  - δ is a sequence of $\{0,1\}^*$ ($0 =$ 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'



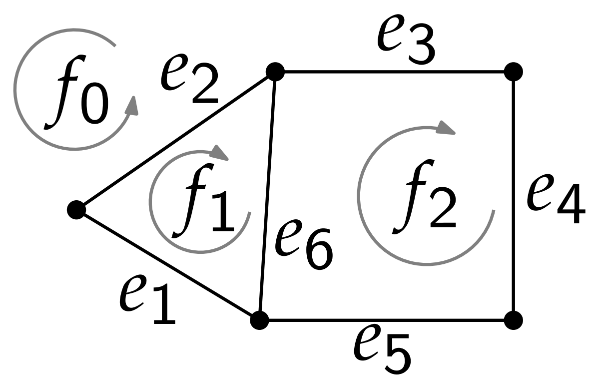
$(e, 100, \pi)$

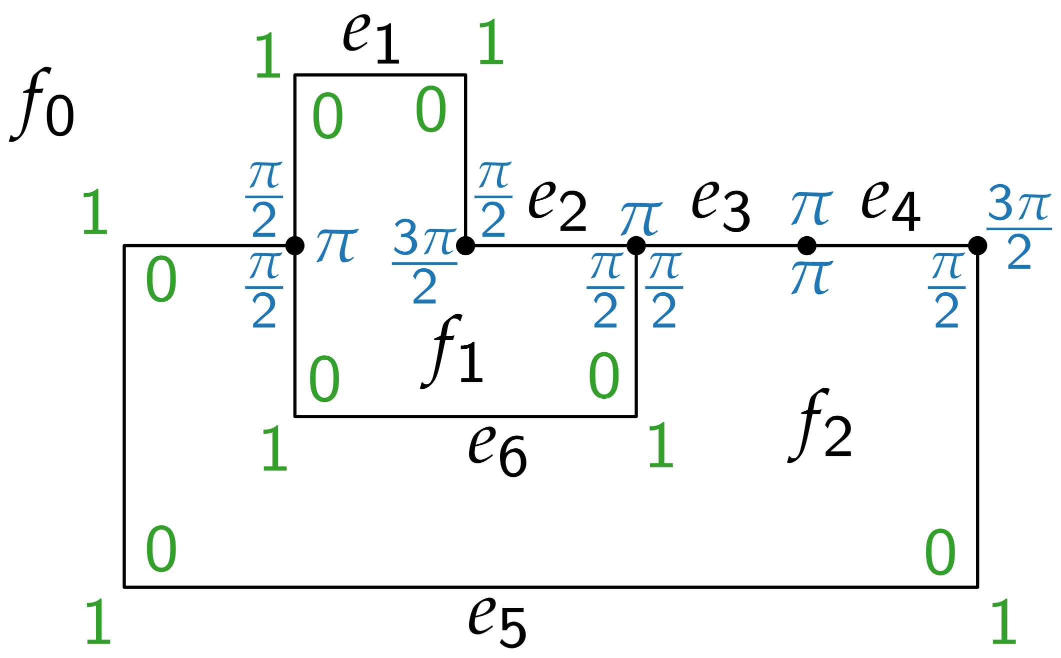
- A **face representation** $H(f)$ of f is a clockwise ordered sequence of edge descriptions (e, δ, α).
- An **orthogonal representation** $H(G)$ of G is defined as

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

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

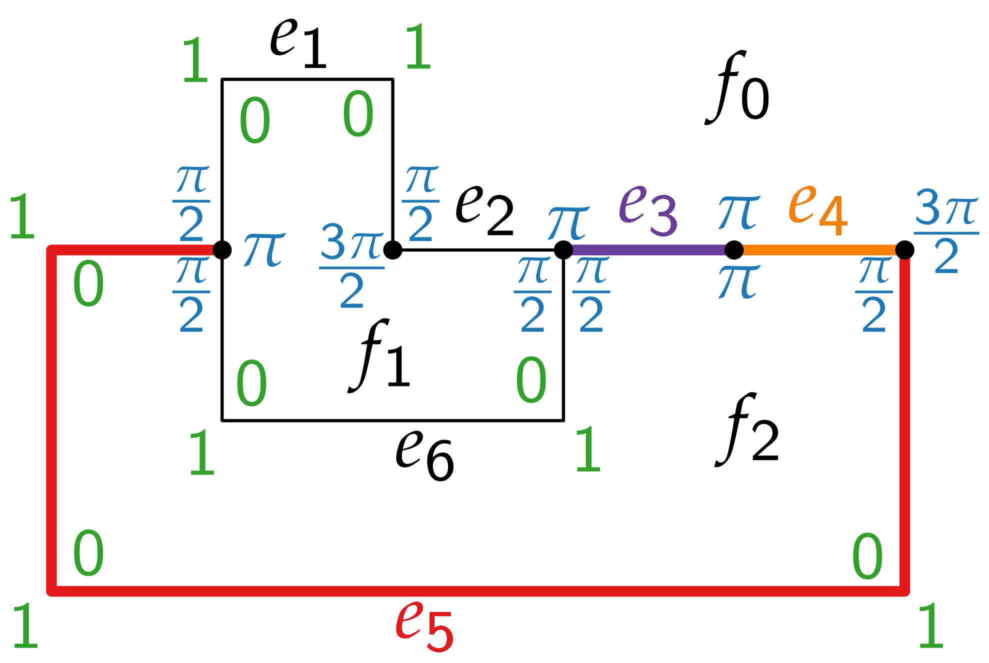
(H1) $H(G)$ corresponds to F, f_0.

(H2) For an 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 δ_1 is reversed and inverted δ_2.

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



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

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

$$C(e_5) = 3 - 0 + 2 - \frac{2\pi}{2\pi} = 4$$

Bend minimisation with given embedding

Geometric bend minimisation.

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 minimisation.

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 minimisation

Combinatorial bend minimisation.

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)

Reminder: s-t flow network

Flow network $(D = (V, A); s, t; u)$ with

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

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

$$0 \leq X(i,j) \leq u(i,j) \qquad \forall (i,j) \in A \tag{1}$$

$$\sum_{(i,j)\in A} X(i,j) - \sum_{(j,i)\in A} X(j,i) = 0 \qquad \forall i \in V \setminus \{s,t\} \tag{2}$$

Reminder: general flow network

Flow network $(D = (V, A); \ell; u; b)$ with

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

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

$$\ell(i,j) \leq X(i,j) \leq u(i,j) \qquad \forall (i,j) \in A \tag{3}$$

$$\sum_{(i,j) \in A} X(i,j) - \sum_{(j,i) \in A} X(j,i) = b(i) \qquad \forall i \in V \tag{4}$$

Problems for flow networks

Valid flow problem.
Find a valid flow $X\colon A \to \mathbb{R}_0^+$, i.e., such that
- lower bounds $\ell(e)$ and capacities $u(e)$ are respected (inequalities (3)) and
- consumption/production $b(i)$ satisfied (inequalities (4)).

Additionally provided:

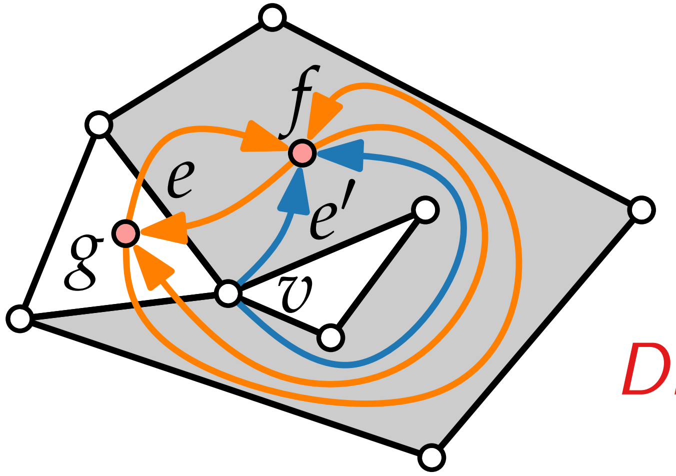
- *Cost function* $\text{cost}\colon A \to \mathbb{R}_0^+$ and $\text{cost}(X) := \sum_{(i,j)\in A} \text{cost}(i,j) \cdot X(i,j)$

Miminum cost flow problem.
Find a valid flow $X\colon A \to \mathbb{R}_0^+$, that minimises cost function $\text{cost}(X)$ (over all valid flows).

Flow network for bend minimisation

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

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



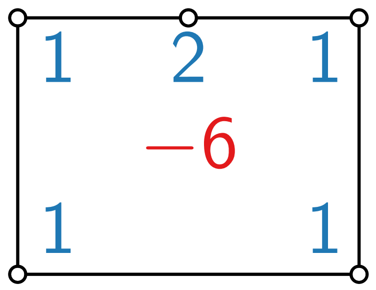
Directed multigraph!

Flow network for bend minimisation

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

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

$\Rightarrow \sum_w b(w) \underset{\text{(Euler)}}{=} 0$



Flow network for bend minimisation

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

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

$\Rightarrow \sum_w b(w) \underset{\text{(Euler)}}{=} 0$

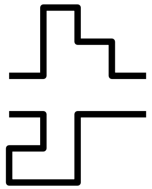
$\forall (v, f) \in A, v \in V, f \in F \qquad \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)$

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

$\forall (f, g) \in A, f, g \in F \qquad \ell(f, g) := 0 \leq X(f, g) \leq \infty =: u(f, g)$

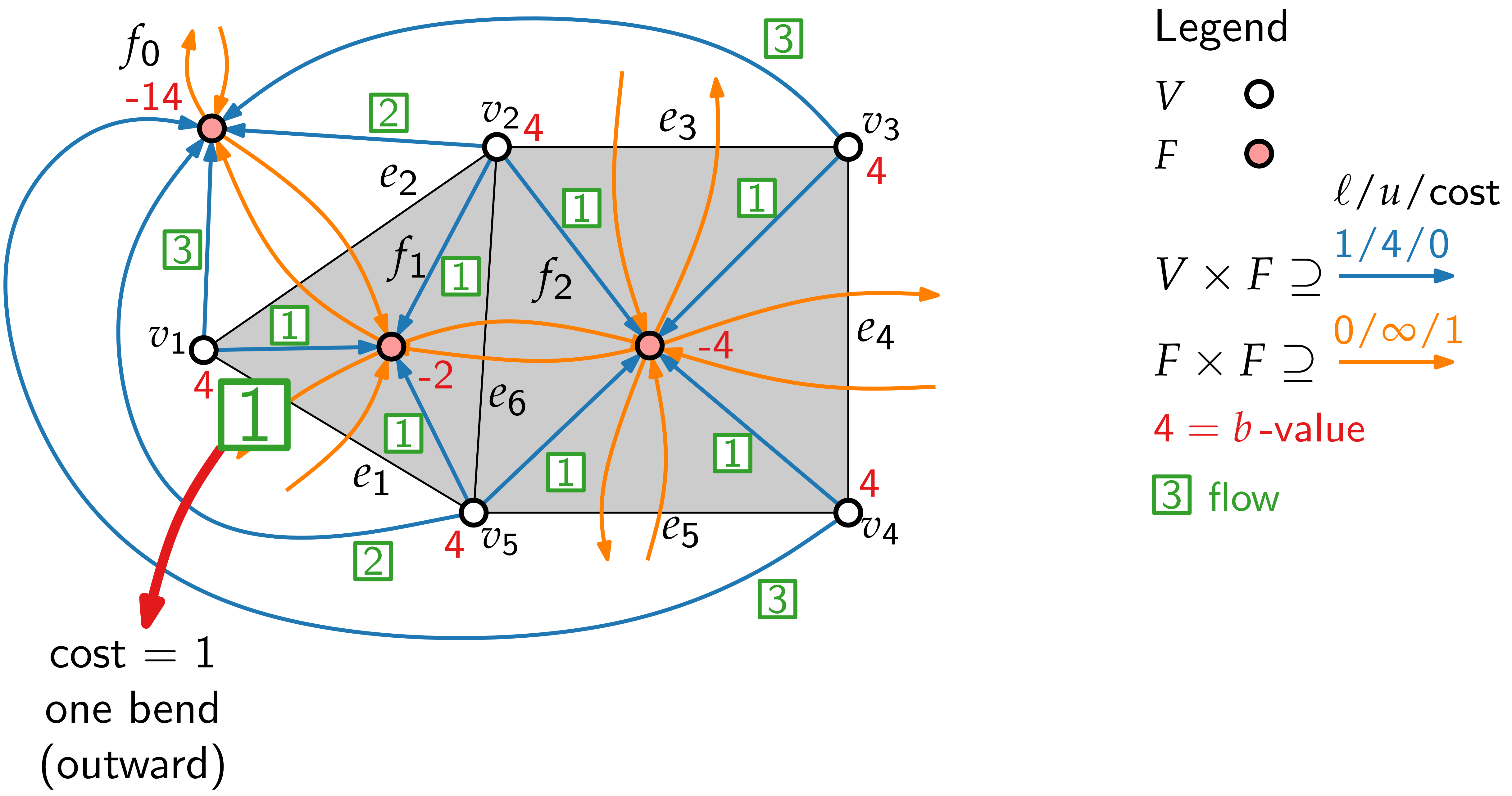
$\text{cost}(f, g) = 1$

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



→ Exercise

Flow network example



Bend minimisation – 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.
- Show properties (H1)–(H4).

(H1) H(G) matches F, f_0 ✓

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

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

(H4) Total angle at each vertex $= 2\pi$ ✓

Bend minimisation – 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 $X \colon A \to \mathbb{R}_0^+$.
- 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, δ_{fg}, x) describes $e \stackrel{*}{=} fg$ from f ✓

(N3) capacities, deficit/demand coverage ✓

(N4) cost $= k$ ✓

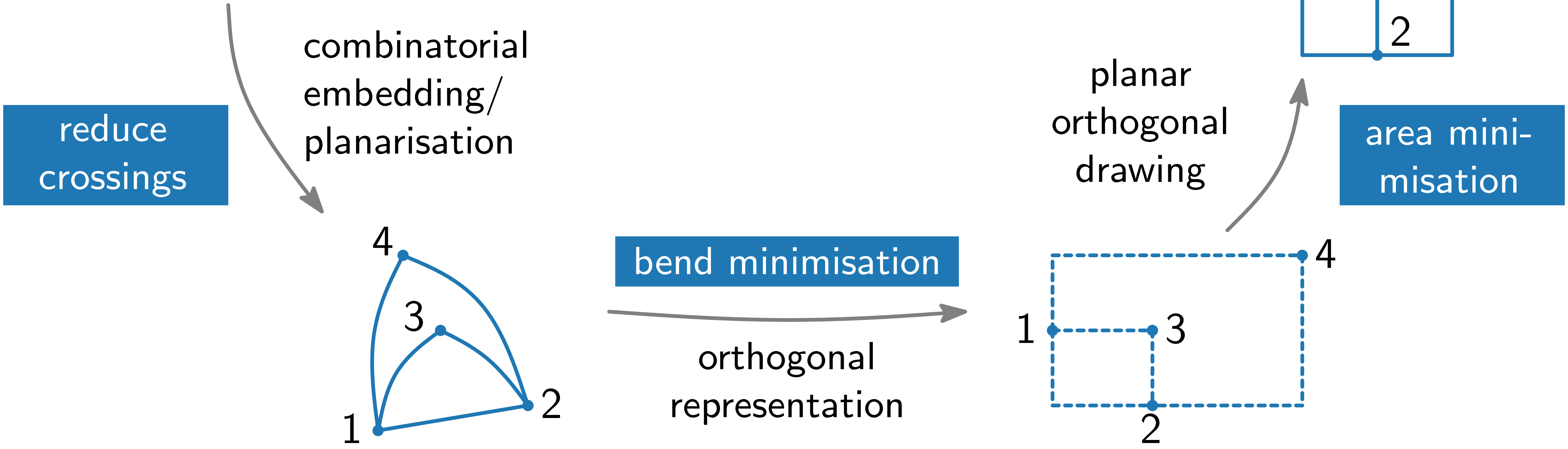
Bend minisation – remarks

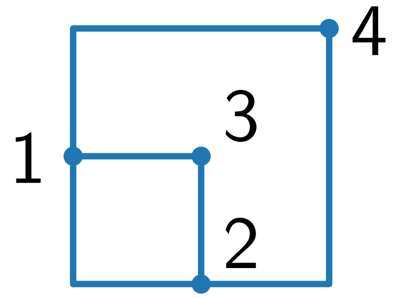
- From Theorem follows that the combinatorial orthogonal bend minimisation problem for plane graphs can be solved using an algorithm for the Min-Cost-Flow problem.
- This special flow problem for a planar network $N(G)$ can be solved in $O(n^{3/2})$ time. [Cornelsen, Karrenbauer GD 2011]
- Bend minimization without a given combinatorial embedding is an NP-hard problem. [Garg, Tamassia SIAM J. Comput. 2001]

Topology - Shape - Metrics

Three-step approach:
[Tamassia SIAM J. Comput. 1987]

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





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 possible
- minimum total edge length
- minimum area

Properties.

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

Idea.

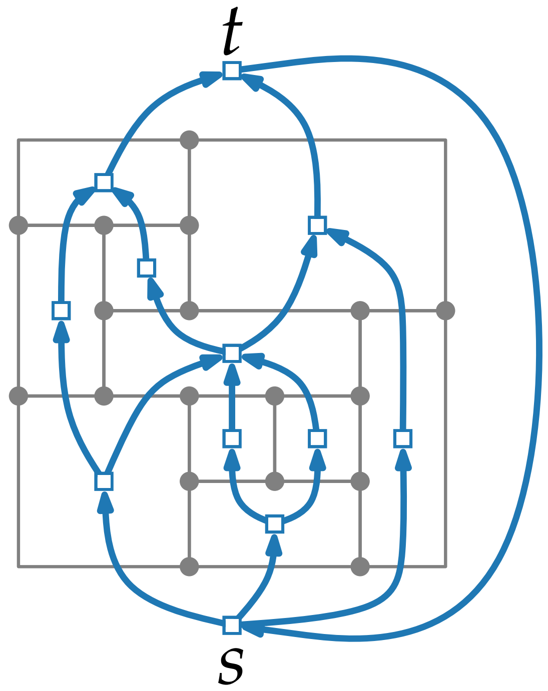
- Formulate flow network for horizontal/vertical compaction

Flow network for edge length assignment

Definition.

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

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

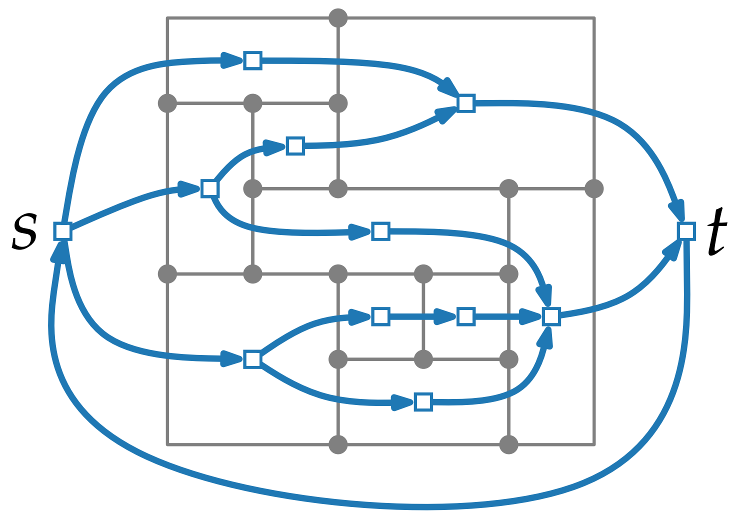


s and t represent lower and upper side of f_0

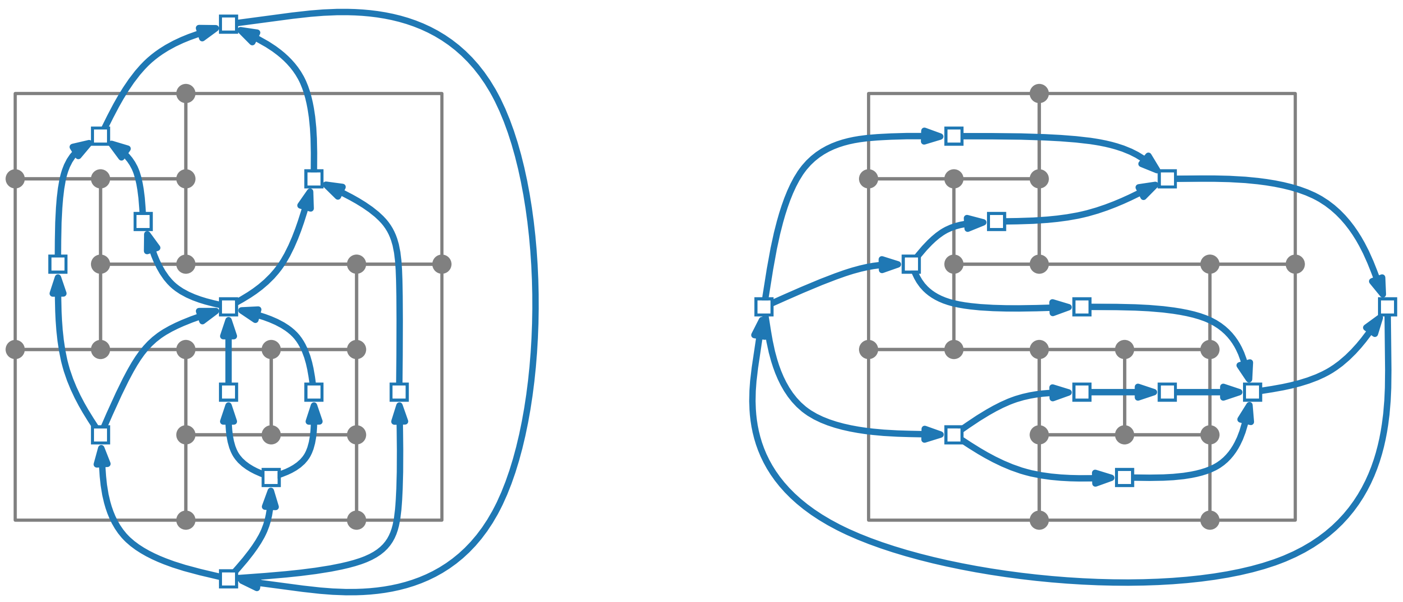
Flow network for edge length assignment

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

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



Compaction – result



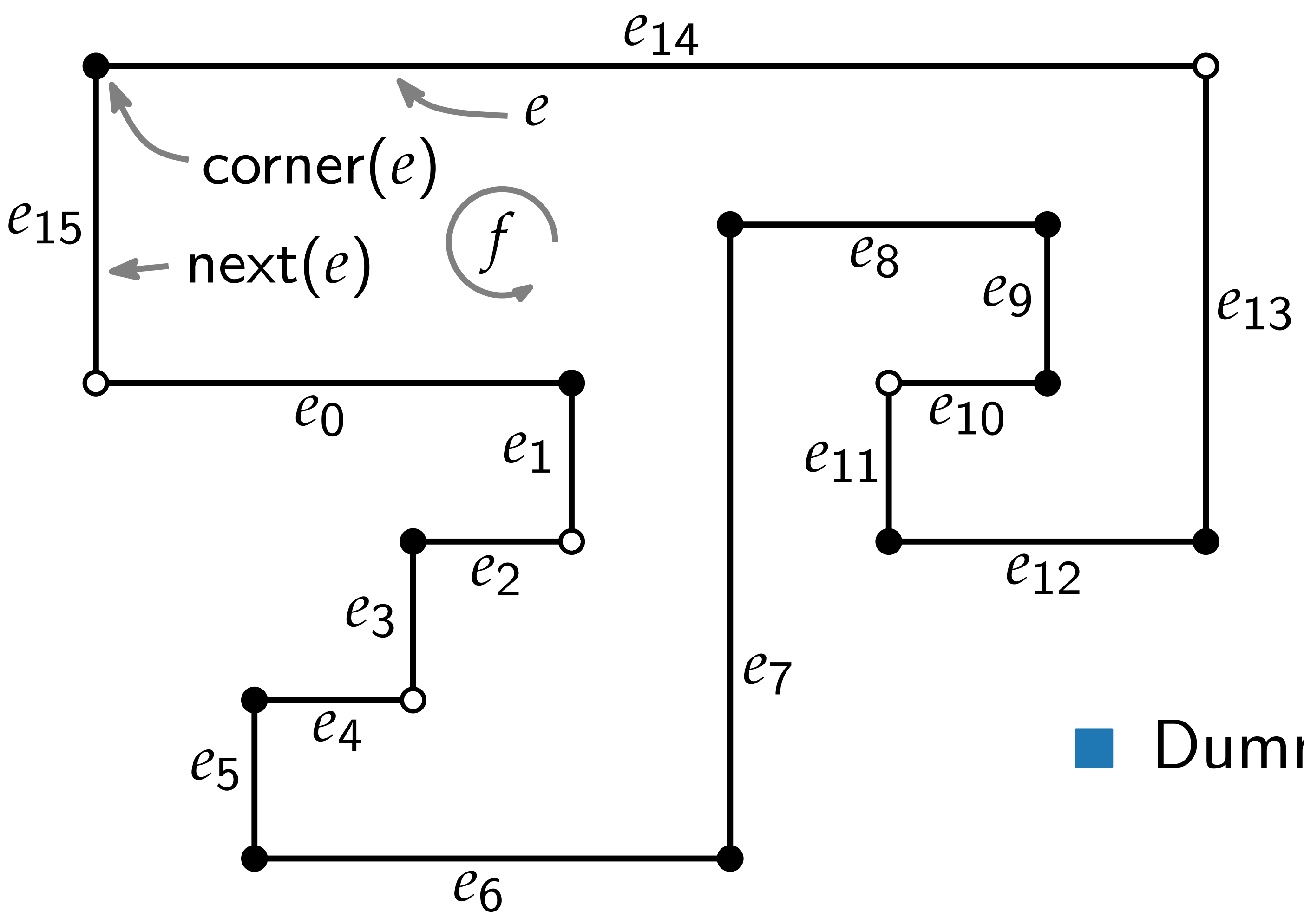
What if not all faces rectangular?

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

What values of the drawing represent the following?

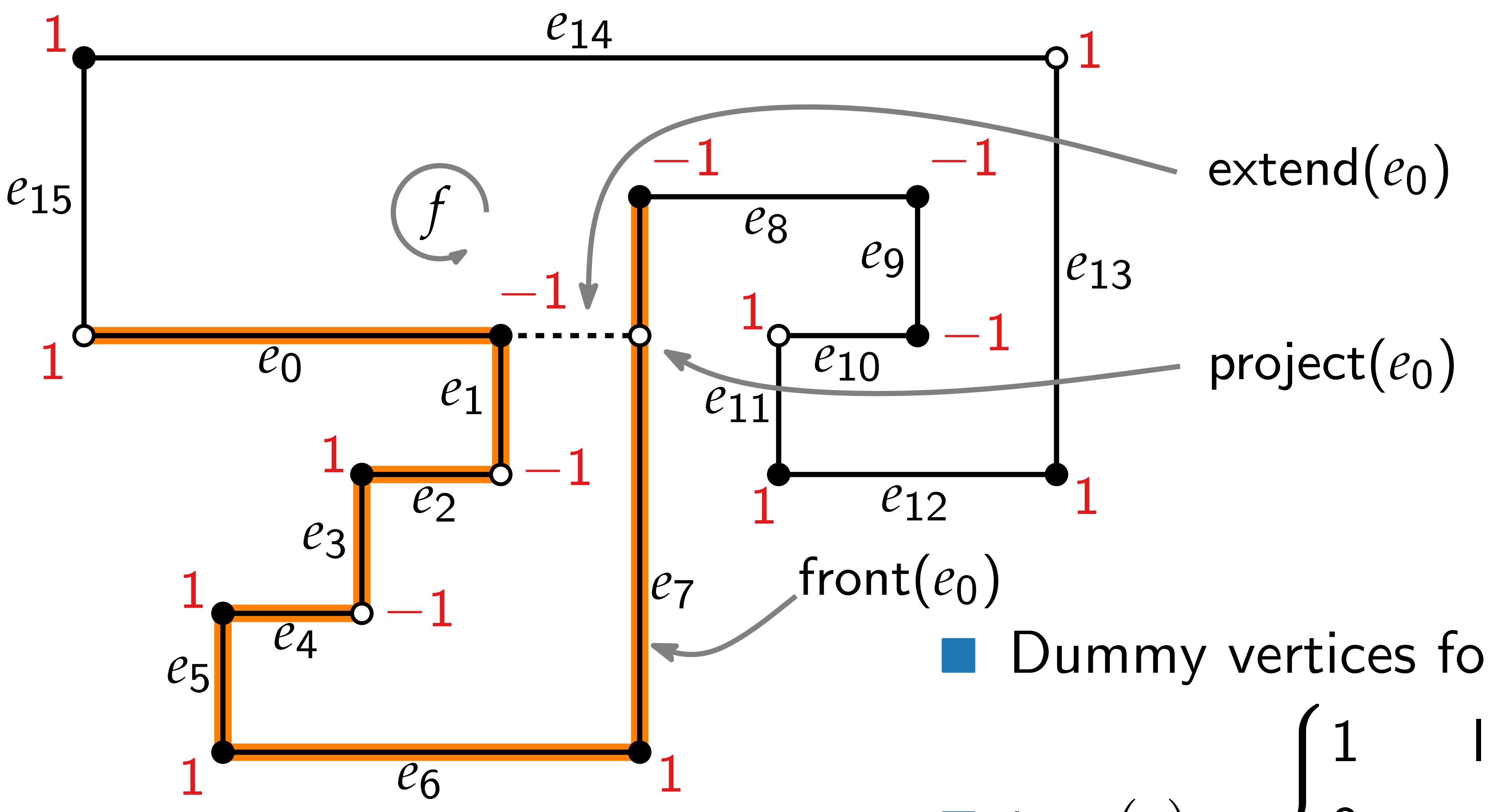
- $|X_{\text{hor}}(t,s)|$ and $|X_{\text{ver}}(t,s)|$?
- $\sum_{a \in A_{\text{hor}}} X_{\text{hor}}(a) + \sum_{a \in A_{\text{ver}}} X_{\text{ver}}(a)$

Refinement of (G, H) – inner face



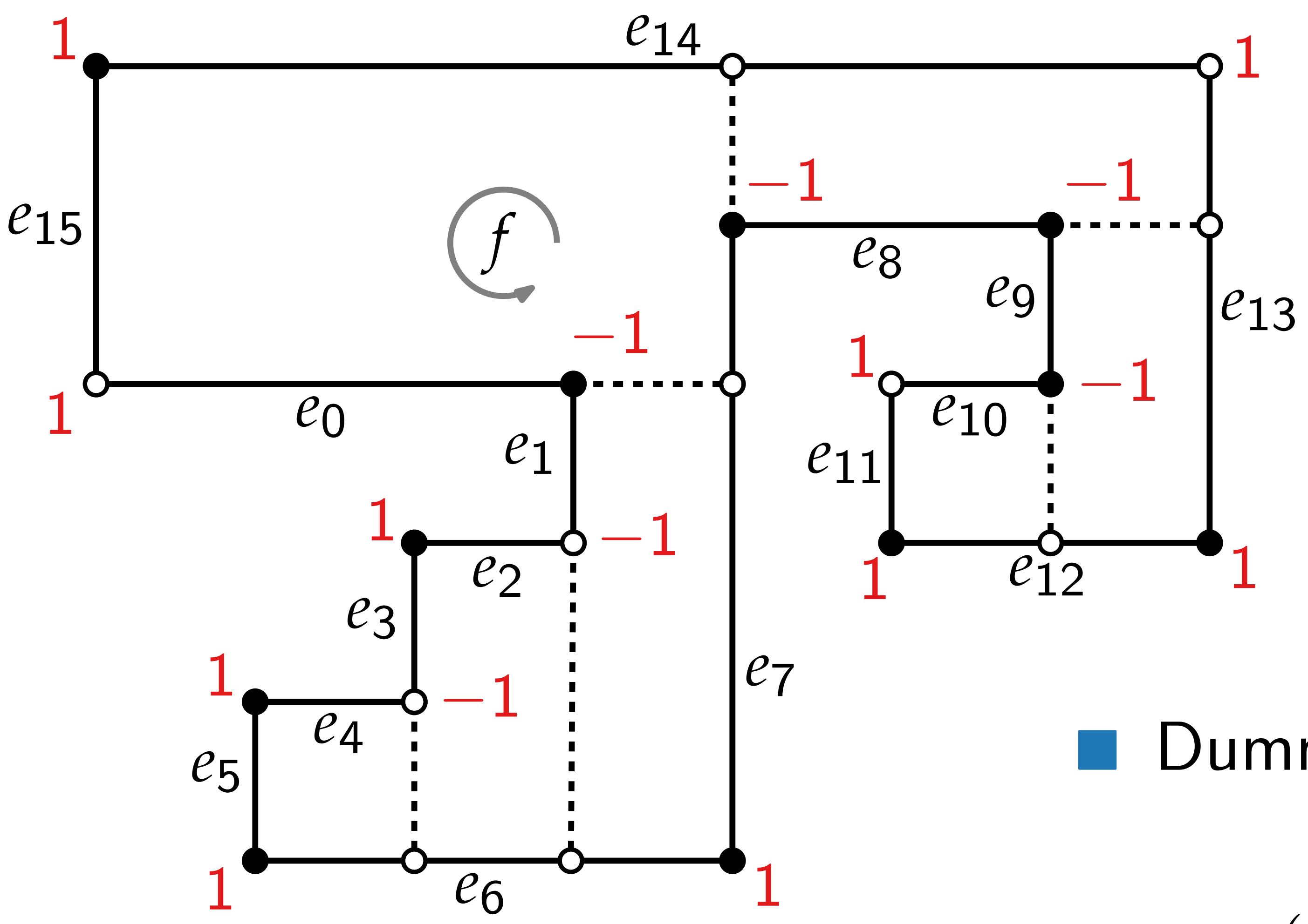
- Dummy vertices for bends

Refinement of (G, H) – inner face



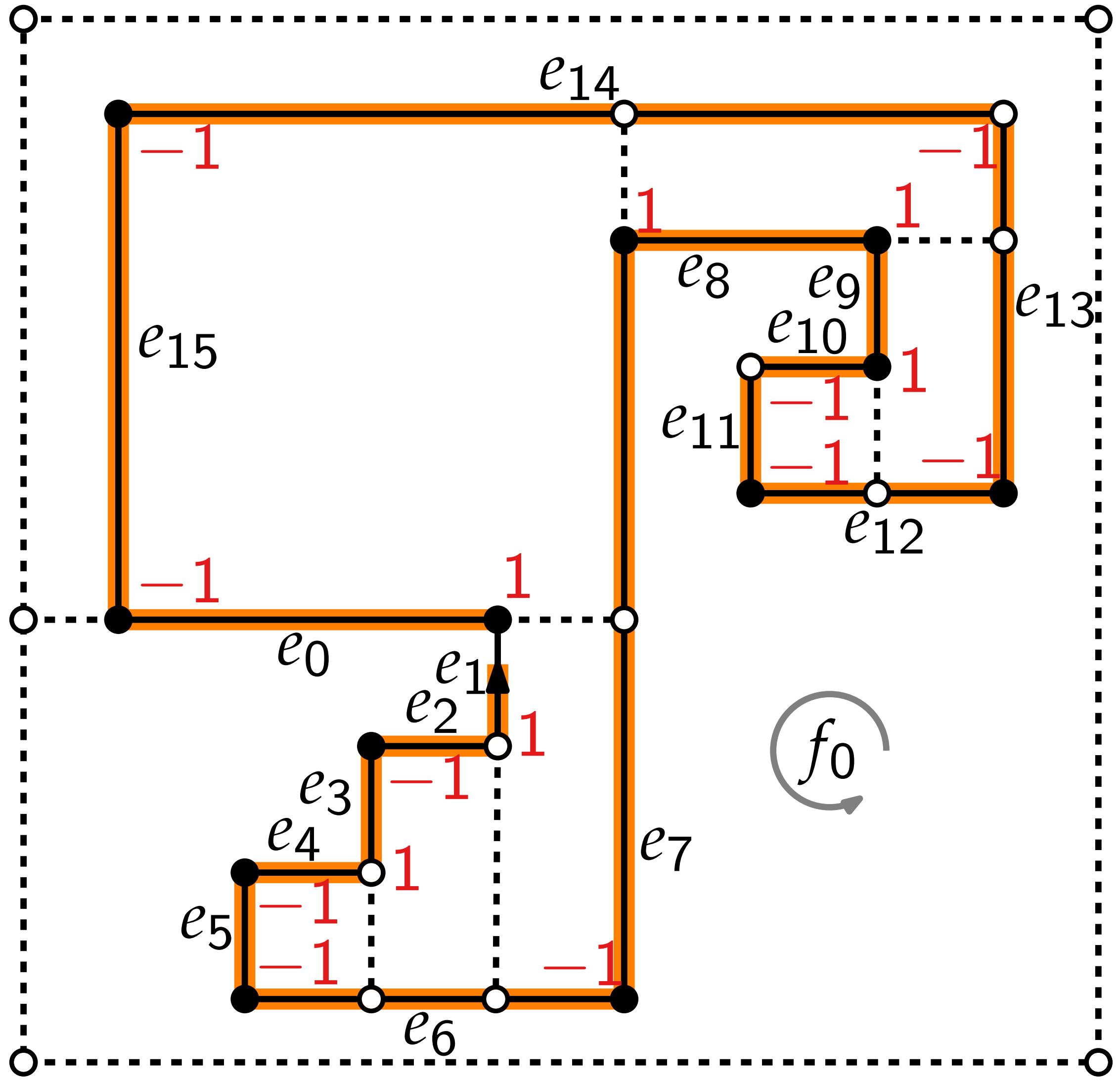
- Dummy vertices for bends
- $\text{turn}(e) = \begin{cases} 1 & \text{left turn} \\ 0 & \text{no turn} \\ -1 & \text{right turn} \end{cases}$

Refinement of (G, H) – inner face

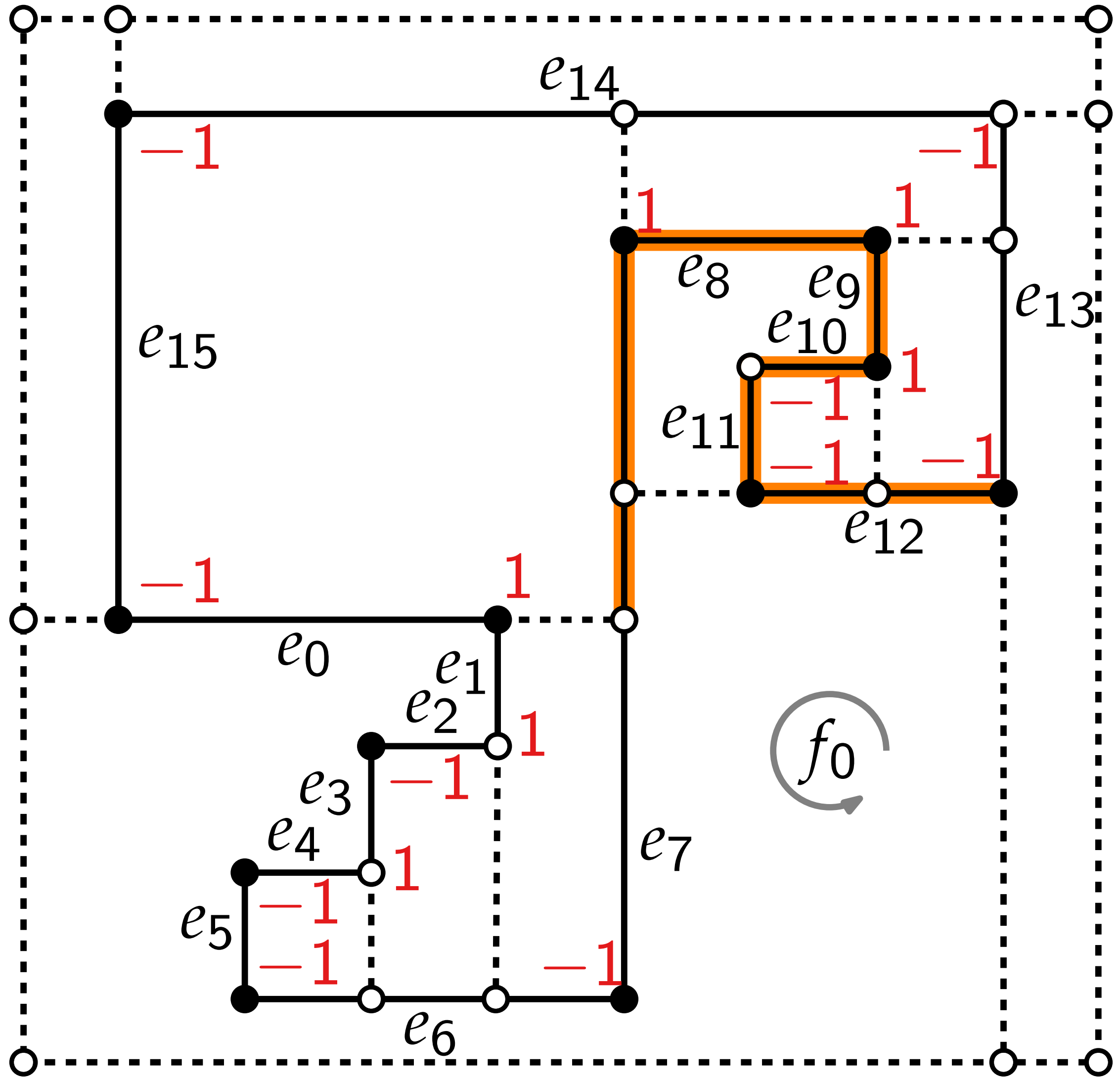


- Dummy vertices for bends
- $\text{turn}(e) = \begin{cases} 1 & \text{left turn} \\ 0 & \text{no turn} \\ -1 & \text{right turn} \end{cases}$

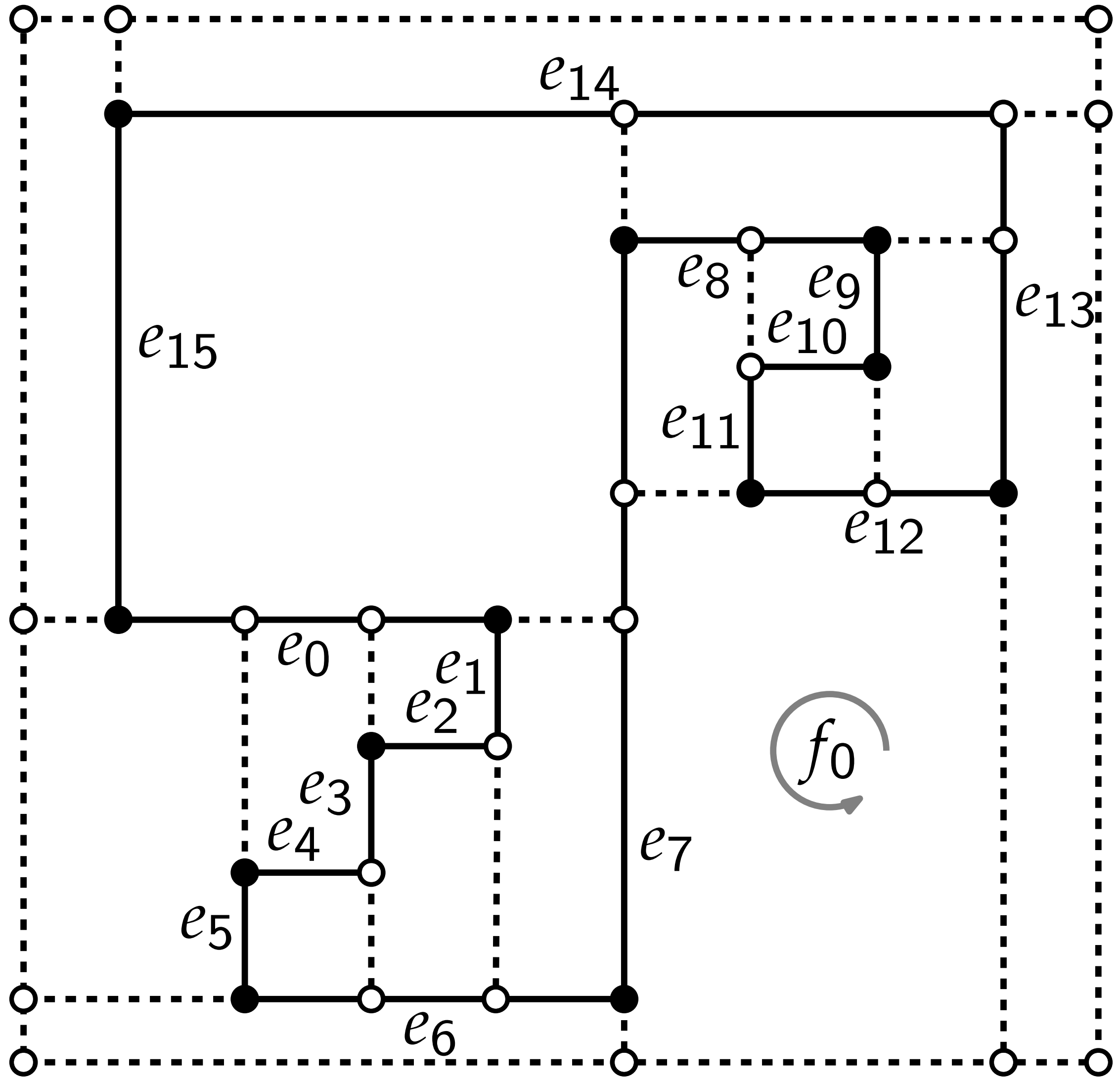
Refinement of (G, H) – outer face



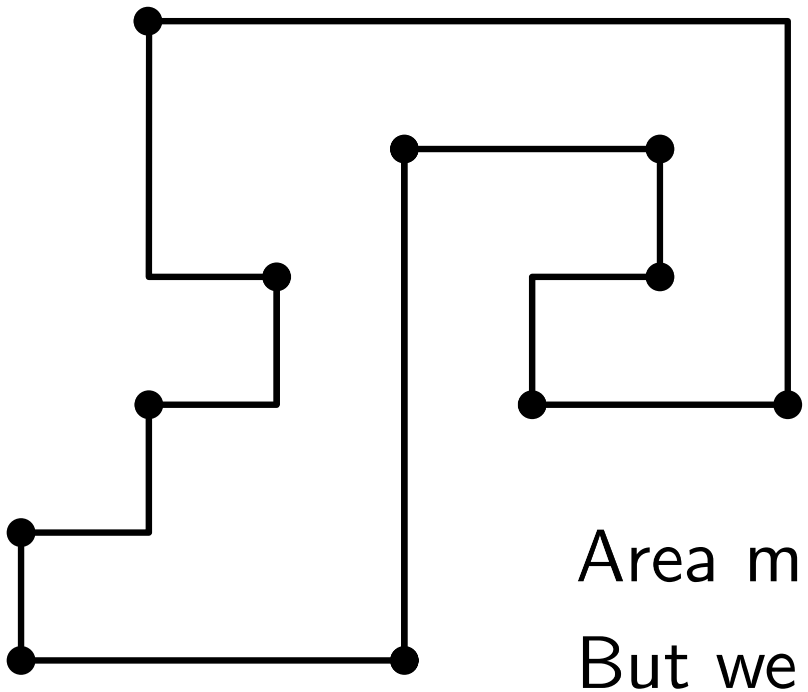
Refinement of (G, H) – outer face



Refinement of (G, H) – outer face



Refinement of (G, H) – outer face



Area minimized? No!

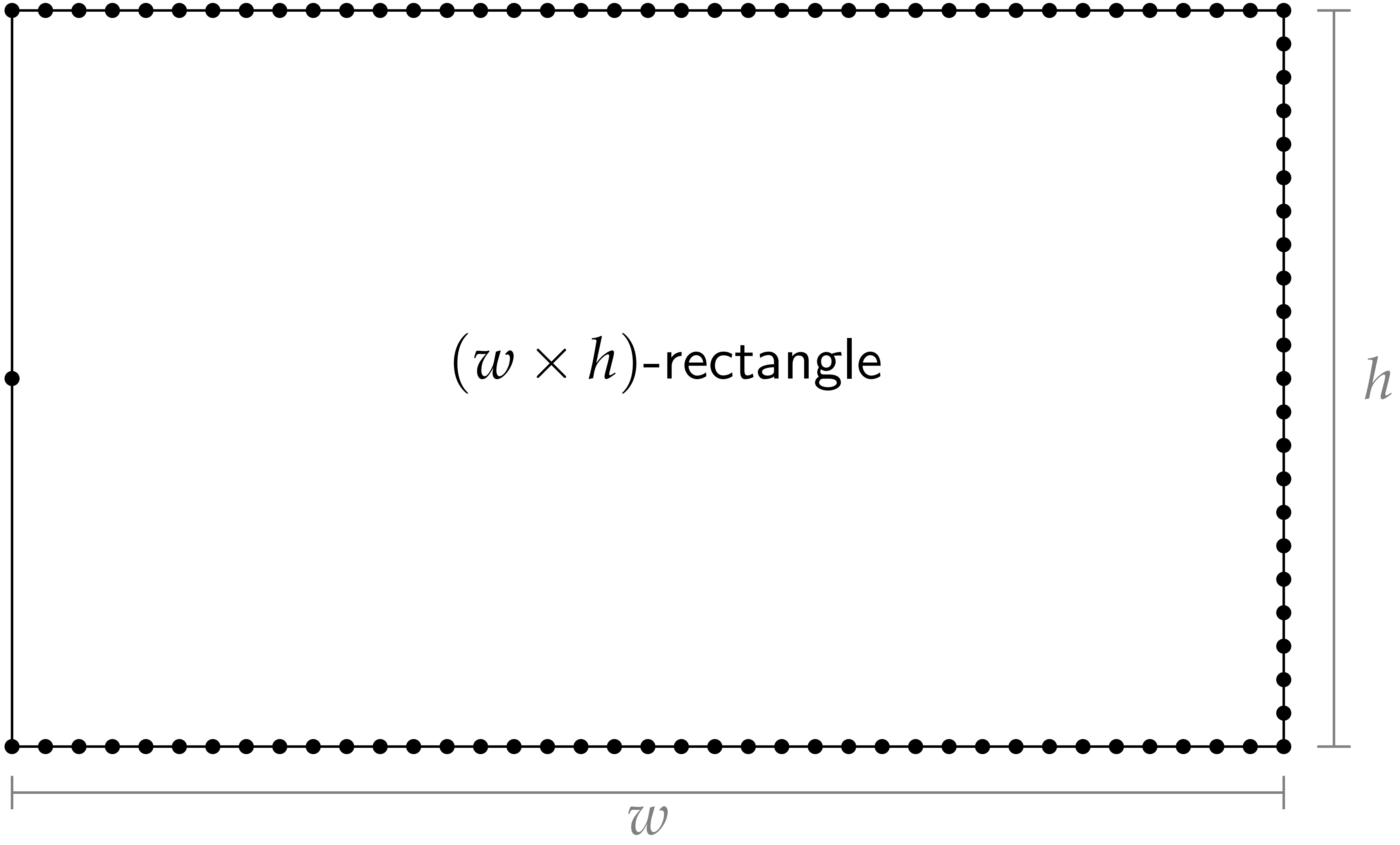
But we get bound $O((n+b)^2)$ on the area.

Compaction for given orthogonal representation is in general NP-hard.

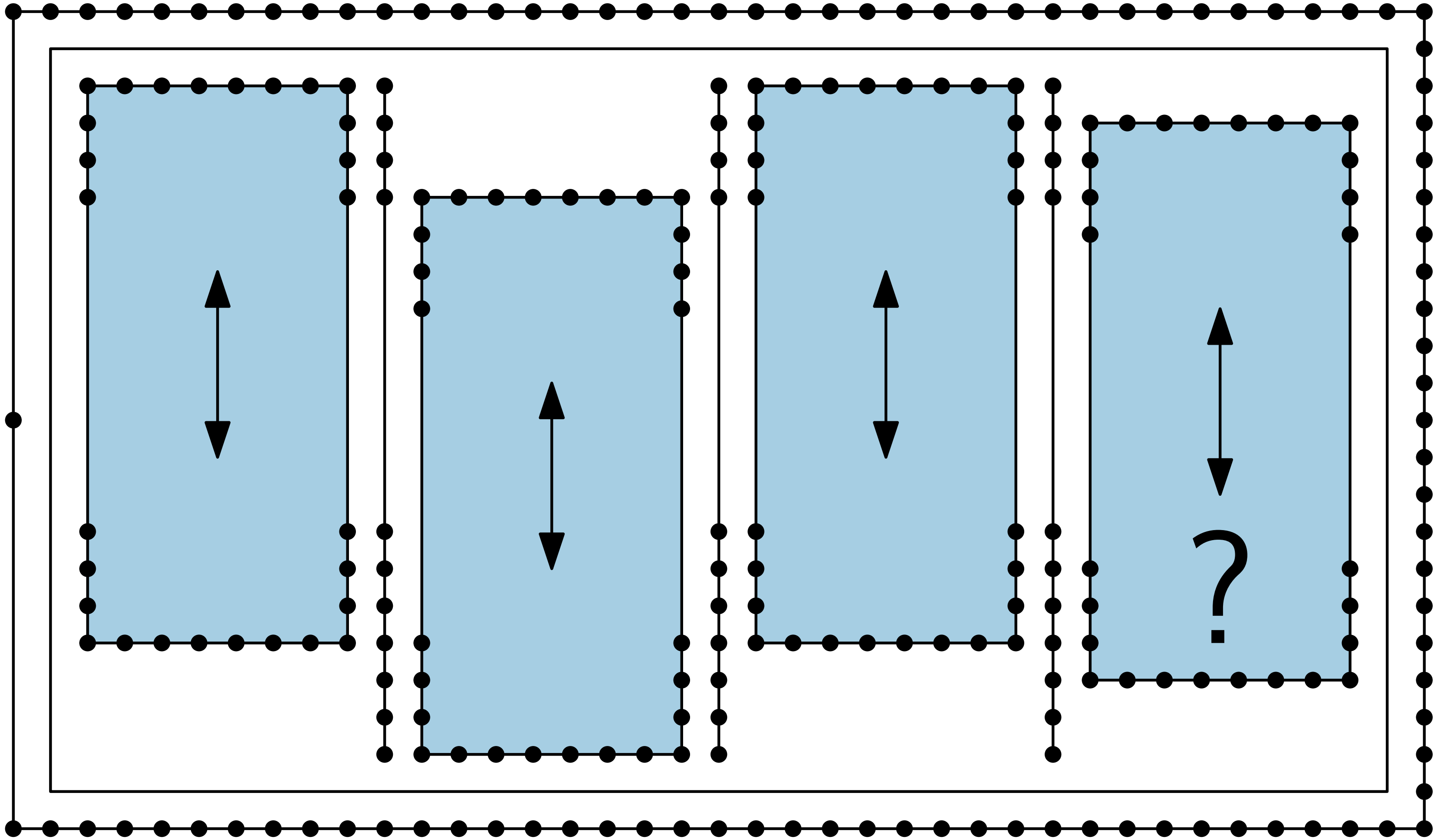
Compactifying is NP-hard [Patrignani '01]

- Reduction via **SAT**
  - n variables $x_1, \ldots, x_n$
  - m clauses $C_1, \ldots, C_m$;
  - each clause: Disjunction of literals $x_i / \overline{x_i}$
    e.g.: $C_1 = x_1 \vee \overline{x_2} \vee x_3$
  - Is $\Phi = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ satisfiable, i.e., is there an assignment to the variables satisfying every clause?
- Find an appropriate value K such that
  (G, H) can be drawn in K area $\Leftrightarrow$ Φ is satisfiable.
- High level structure of (G, H)
  - boundary
  - belts, and pistons
  - clause gadgets
  - variable gadgets

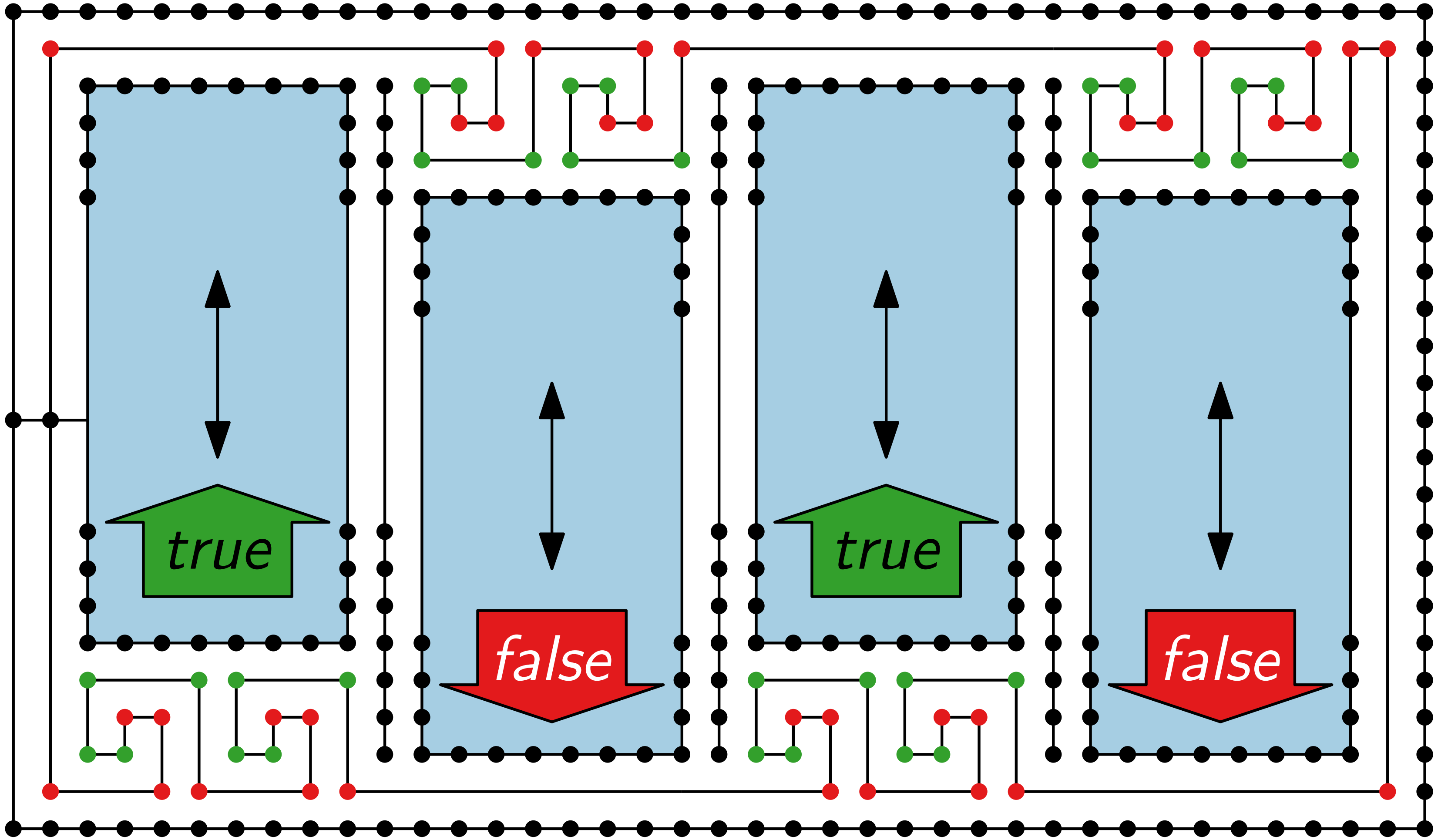
Boundary, belt, and "piston" gadget



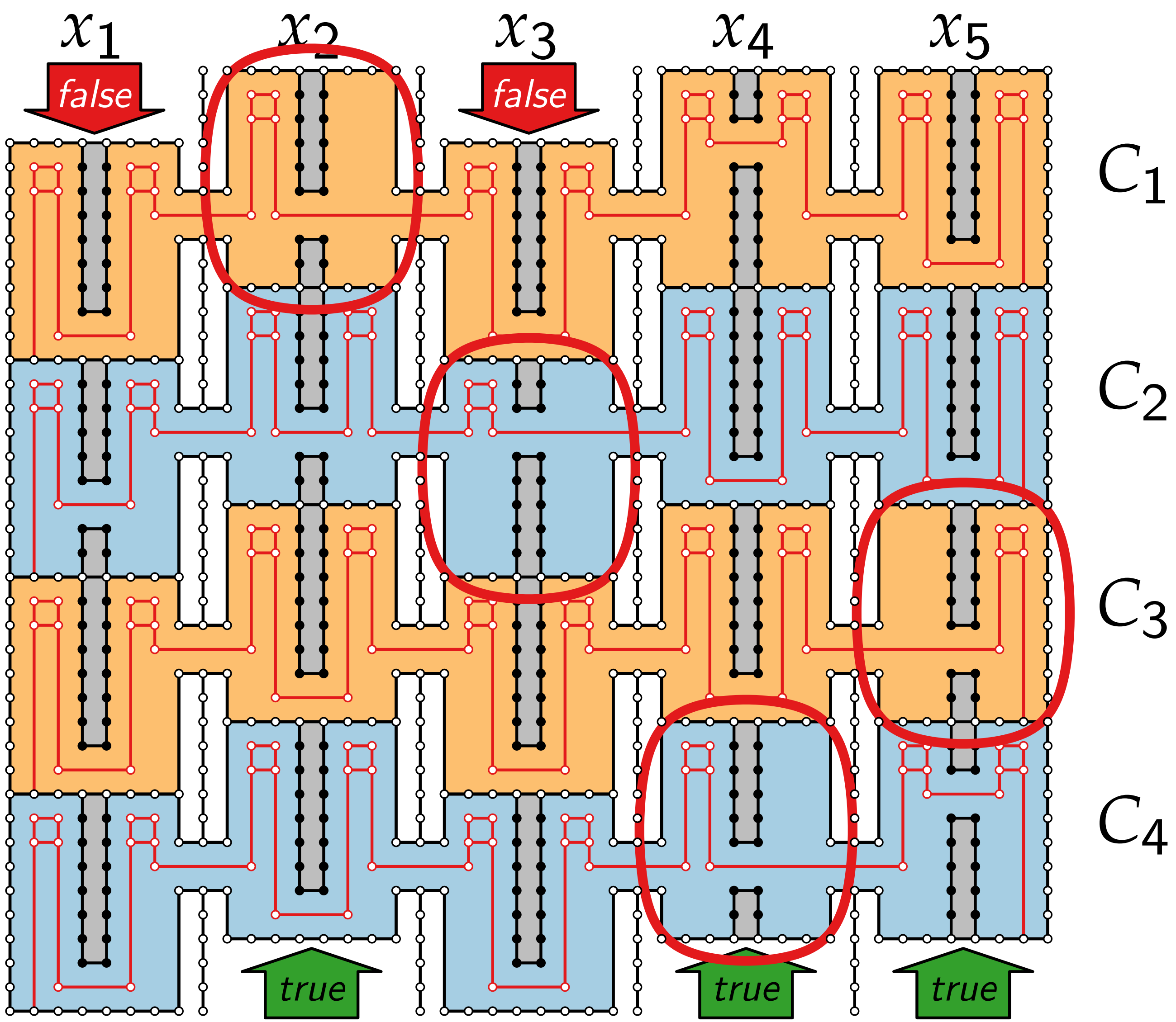
Boundary, belt, and “piston” gadget



Boundary, belt, and "piston" gadget



Clause gadgets



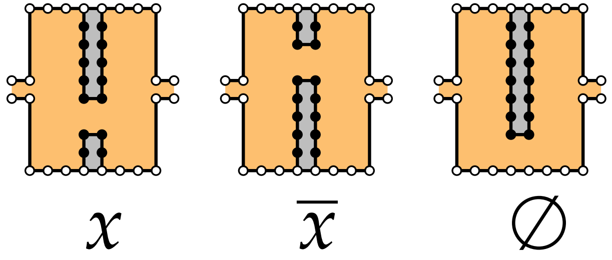
Example:

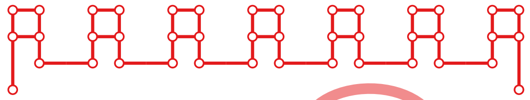
$C_1 = x_2 \vee \overline{x_4}$

$C_2 = x_1 \vee x_2 \vee \overline{x_3}$

$C_3 = x_5$

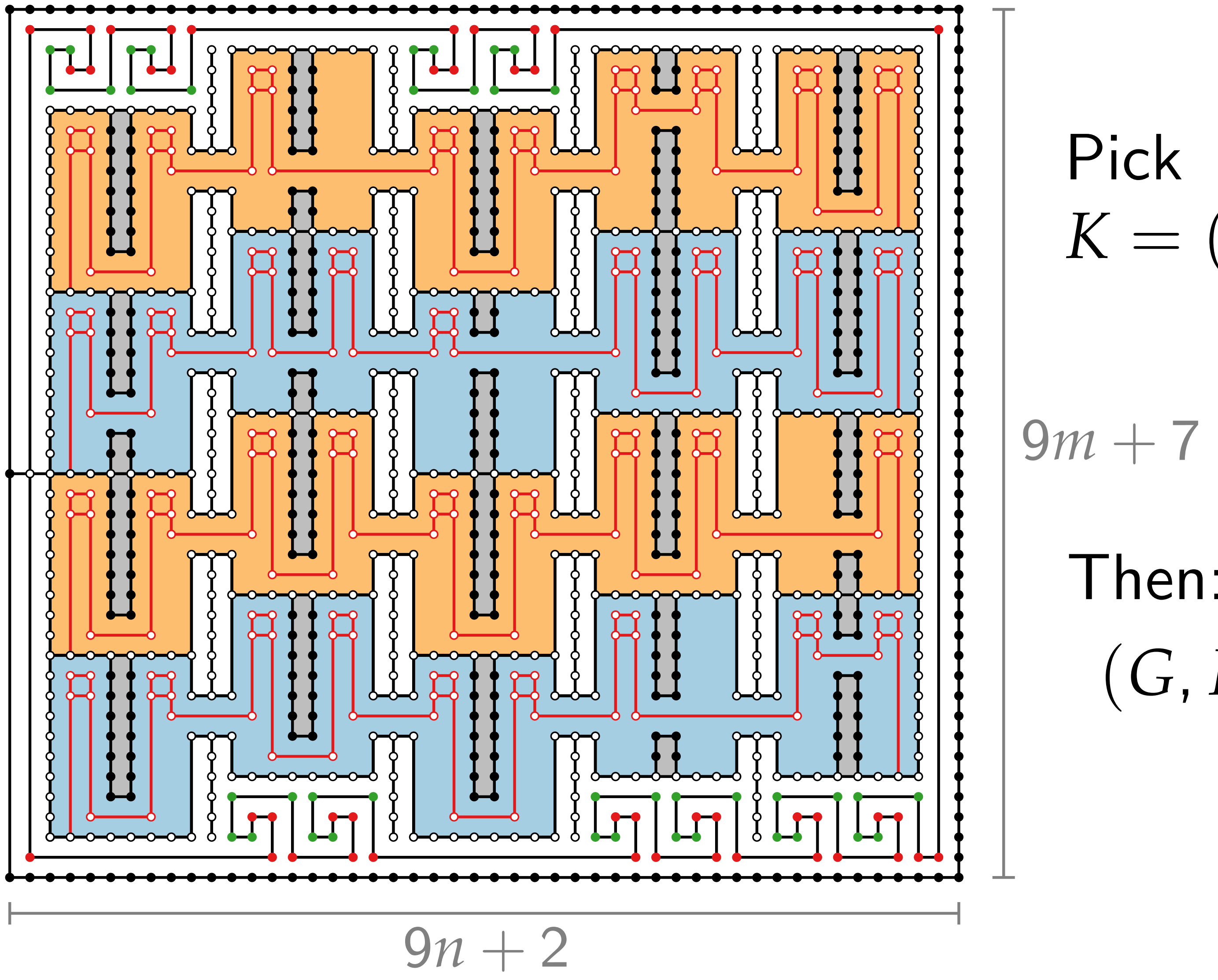
$C_4 = x_4 \vee \overline{x_5}$





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

Complete reduction



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

Then:

(G, H) has an area K drawing
$\Leftrightarrow$
Φ satisfiable $\square$

Literature

- [GD Ch. 5] for detailed explanation
- [Tam87] Tamassia “On embedding a graph in the grid with the minmum number of bends” 1987 – original paper on flow for bend minimisation
- [Pat01] Patrignani “On the complexity of orthogonal compaction”’ 2001– NP-hardness proof of compactification