Algorithmen zur Visualisierung von Graphen

Kombinatorische Optimierung mittels Flussmethoden

Vorlesung im Wintersemester 2012/2013

Tamara Mchedlidze – Martin Nöllenburg – Ignaz Rutter

30.10.2012

Institute for Theoretical Informatics
Algorithmics Group

Kombinatorische Optimierung mittels Flussmethoden

(Kapitel 4)

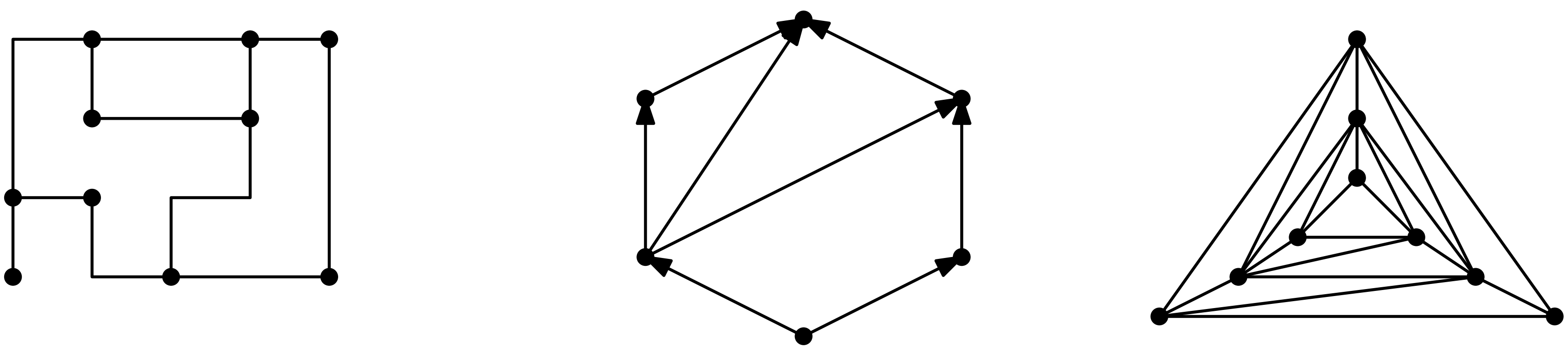
Institute for Theoretical Informatics
Algorithmics Group

Inhalt Kapitel 4

Layoutprobleme für *planare* Graphen

- orthogonale Layouts
- aufwärtsgerichtete Layouts für gerichtete azyklische Graphen
- Winkelauflösung in geradlinigen Layouts

→ Modellierung der Probleme durch Flussnetzwerke



Institute for Theoretical Informatics
Algorithmics Group

Planarität

Definition

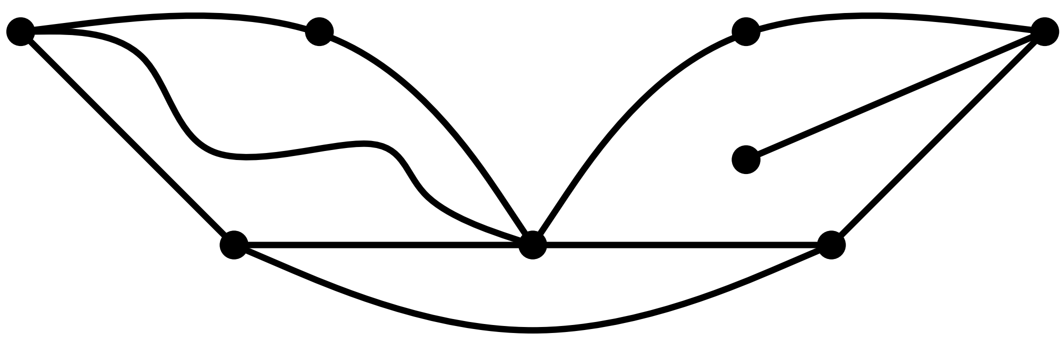
Ein Graph $G = (V, E)$ heißt planar, wenn es eine Zeichnung Γ von G (in Standardrepräsentation) gibt, in der sich keine zwei Kanten schneiden.
Eine solche Zeichnung Γ heißt planare Einbettung von G.

Institute for Theoretical Informatics
Algorithmics Group

Planarität

Definition

Ein Graph $G = (V, E)$ heißt planar, wenn es eine Zeichnung Γ von G (in Standardrepräsentation) gibt, in der sich keine zwei Kanten schneiden.
Eine solche Zeichnung Γ heißt planare Einbettung von G.

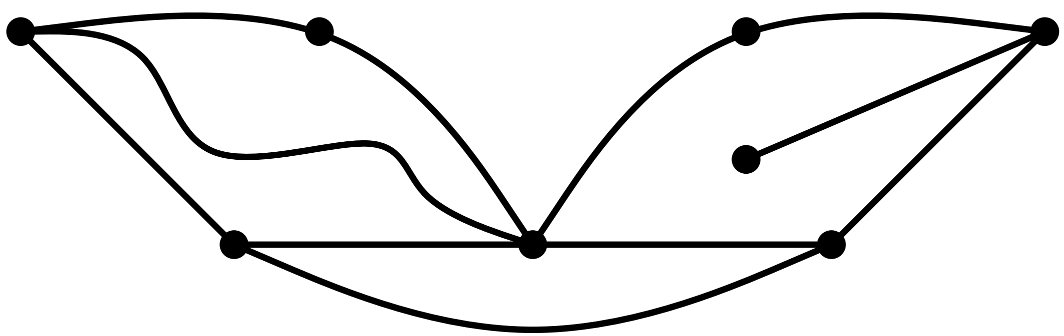


Institute for Theoretical Informatics
Algorithmics Group

Planarität

Definition

Ein Graph $G = (V, E)$ heißt planar, wenn es eine Zeichnung Γ von G (in Standardrepräsentation) gibt, in der sich keine zwei Kanten schneiden.
Eine solche Zeichnung Γ heißt planare Einbettung von G.



Zusammenhang zwischen n, m und $|\text{Facetten}|$?

s-t Flussnetzwerk

Definition

Geg: *Flussnetzwerk* $(D = (V, A); s, t; c)$ mit

- gerichtetem Graph $D = (V, A)$
- *Kantenkapazitäten* $c : A \to \mathbb{R}_0^+$
- *Quelle* $s \in V$, *Senke* $t \in V$

Eine Abbildung $X : A \to \mathbb{R}_0^+$ heißt s-t-Fluss, falls gilt:

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

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

Institute for Theoretical Informatics
Algorithmics Group

Allgemeines Flussnetzwerk

Definition

Geg: *Flussnetzwerk* $(D = (V, A); \ell; u; b)$ mit

- gerichtetem Graph $D = (V, A)$
- untere Kantenkapazitäten $\ell : A \to \mathbb{R}_0^+$
- obere Kantenkapazitäten $u : A \to \mathbb{R}_0^+$
- Knotenbewertung $b : V \to \mathbb{R}$ mit $\sum_{i \in V} b(i) = 0$

Eine Abbildung $X : A \to \mathbb{R}_0^+$ heißt Fluss, falls gilt:

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

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

Fragestellungen auf geg. Flussnetzwerk

Gültiger Fluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, d.h., der

- Kapazitäten respektiert, $l(e), u(e)$
- Bedarf genau deckt, $b(v)$

Institute for Theoretical Informatics
Algorithmics Group

Fragestellungen auf geg. Flussnetzwerk

Gültiger Fluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, d.h., der

- Kapazitäten respektiert, $l(e), u(e)$
- Bedarf genau deckt, $b(v)$

Zusätzlich gegeben: Kostenfunktion $\mathrm{cost} : A \to \mathbb{R}_0^+$
Definiere: $\mathrm{cost}(X) := \sum_{(i,j)\in A} \mathrm{cost}(i,j) \cdot X(i,j)$

Institute for Theoretical Informatics
Algorithmics Group

Fragestellungen auf geg. Flussnetzwerk

Gültiger Fluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, d.h., der

- Kapazitäten respektiert, $l(e), u(e)$
- Bedarf genau deckt, $b(v)$

Zusätzlich gegeben: Kostenfunktion $\mathsf{cost} : A \to \mathbb{R}_0^+$
Definiere: $\mathsf{cost}(X) := \sum_{(i,j)\in A} \mathsf{cost}(i,j) \cdot X(i,j)$

Minimalkostenfluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, der

- $\mathsf{cost}(X)$ minimiert (unter allen gültigen Flüssen)

Fragestellungen auf geg. Flussnetzwerk

Karlsruhe Institute of Technology

Gültiger Fluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, d.h., der

- Kapazitäten respektiert, $l(e), u(e)$
- Bedarf genau deckt, $b(v)$

Zusätzlich gegeben: Kostenfunktion $\mathsf{cost} : A \to \mathbb{R}_0^+$
Definiere: $\mathsf{cost}(X) := \sum_{(i,j)\in A} \mathsf{cost}(i,j) \cdot X(i,j)$

Minimalkostenfluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, der

- $\mathsf{cost}(X)$ minimiert (unter allen gültigen Flüssen)

Algorithmen mit Laufzeit $O(n^2 \log n)$ bzw. $O(n^{7/4} \log n)$

Fragestellungen auf geg. Flussnetzwerk

Karlsruhe Institute of Technology

Gültiger Fluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, d.h., der

- Kapazitäten respektiert, $l(e), u(e)$
- Bedarf genau deckt, $b(v)$

Zusätzlich gegeben: Kostenfunktion $\mathsf{cost} : A \to \mathbb{R}_0^+$
Definiere: $\mathsf{cost}(X) := \sum_{(i,j)\in A} \mathsf{cost}(i,j) \cdot X(i,j)$

Minimalkostenfluß

Finde einen gültigen Fluss $X : A \to \mathbb{R}_0^+$, der

- $\mathsf{cost}(X)$ minimiert (unter allen gültigen Flüssen)

Algorithmen mit Laufzeit $O(n^2 \log n)$ bzw. $O(n^{7/4} \log n)$

neu: $O(n^{3/2})$ [Cornelsen, Karrenbauer GD'11]

Institute for Theoretical Informatics
Algorithmics Group

(Planare) Orthogonale Zeichnungen

Dreistufiger Ansatz: *Topology – Shape – Metrics*

$$V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$$

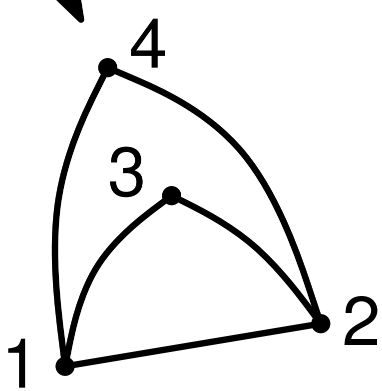
(Planare) Orthogonale Zeichnungen

Dreistufiger Ansatz: *Topology – Shape – Metrics*

$V = \{v_1, v_2, v_3, v_4\}$
$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$

Kreuzungsminimierung

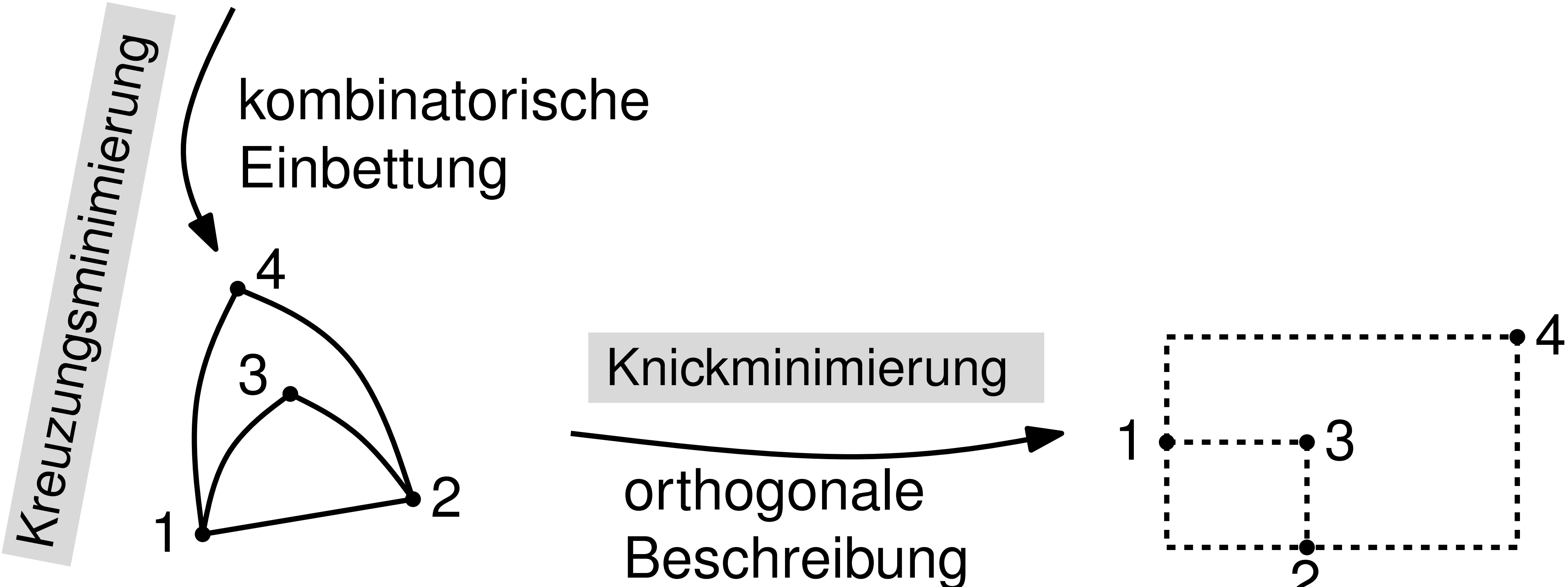
kombinatorische Einbettung



(Planare) Orthogonale Zeichnungen

Dreistufiger Ansatz: *Topology – Shape – Metrics*

$$V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$$

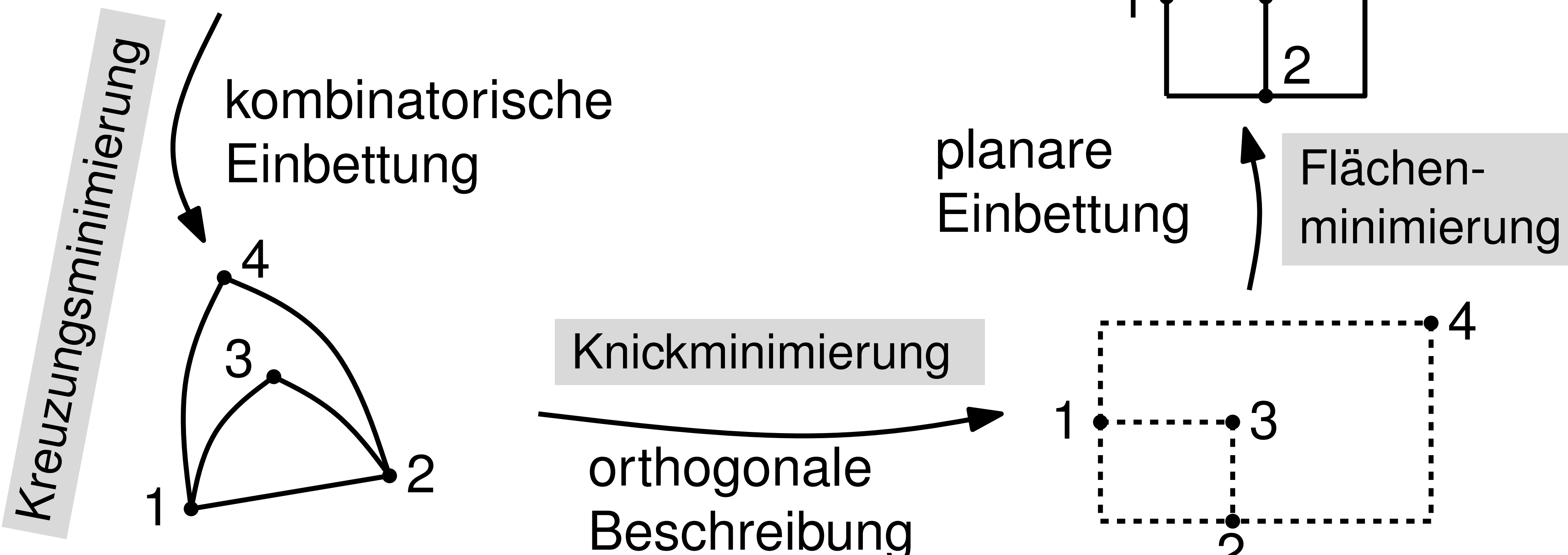


Institute for Theoretical Informatics
Algorithmics Group

(Planare) Orthogonale Zeichnungen

Dreistufiger Ansatz: *Topology – Shape – Metrics*

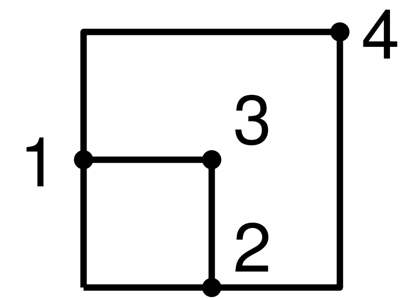
$$V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$$



(Planare) Orthogonale Zeichnungen

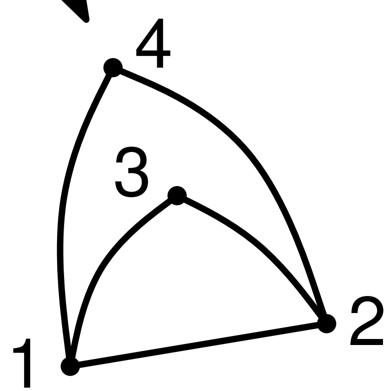
Dreistufiger Ansatz: *Topology – Shape – Metrics*

$V = \{v_1, v_2, v_3, v_4\}$
$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$



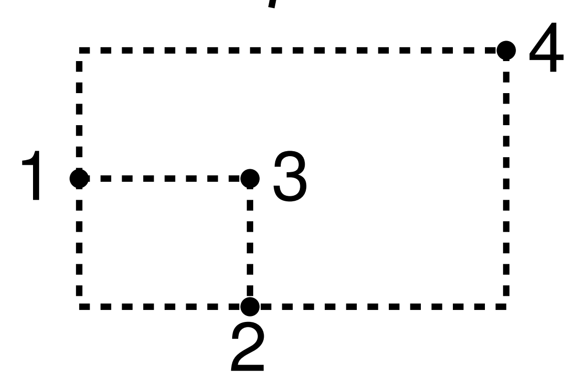
Kreuzungsminimierung

kombinatorische Einbettung



Knickminimierung

orthogonale Beschreibung



planare Einbettung

Flächen-minimierung

Problem: Orthogonale Beschreibung

Problem: Knickminimierung bei fester Einbettung

Gegeben ein Graph $G = (V, E)$ mit Maximalgrad $\mathrm{deg}_{\mathrm{max}} \leq 4$, kombinatorischer Einbettung $\mathcal{F}$ und äußerer Facette f_0, finde eine orthogonale Gitterzeichnung, die $(\mathcal{F},\ f_0)$ erhält und minimale Anzahl von Knicken hat.

Institute for Theoretical Informatics
Algorithmics Group

Problem: Orthogonale Beschreibung

Problem: Knickminimierung bei fester Einbettung

Gegeben ein Graph $G = (V, E)$ mit Maximalgrad $\deg_{\max} \leq 4$, kombinatorischer Einbettung $\mathcal{F}$ und äußerer Facette f_0, finde eine orthogonale Gitterzeichnung, die $(\mathcal{F},\ f_0)$ erhält und minimale Anzahl von Knicken hat.

Problem': Orthogonale Beschreibung

Gegeben ein Graph $G = (V, E)$ mit Maximalgrad $\deg_{\max} \leq 4$, kombinatorischer Einbettung $\mathcal{F}$ und äußerer Facette f_0, finde eine orthogonale Beschreibung $H(G)$ die $(\mathcal{F},\ f_0)$ erhält und minimale Anzahl von Knicken hat.

Institute for Theoretical Informatics
Algorithmics Group

Orthogonale Beschreibung

Eingabe: planarer Graph $G = (V, E)$, Facettenmenge $\mathcal{F}$, äußere Facette f_0

Ausgabe: orthogonale Beschreibung $H(G) = \{H(f) \mid f \in \mathcal{F}\}$

Facettenbeschreibung $H(f)$**:** im UZS um f geordnete Folge von Kantenbeschreibungen (e, δ, α) mit

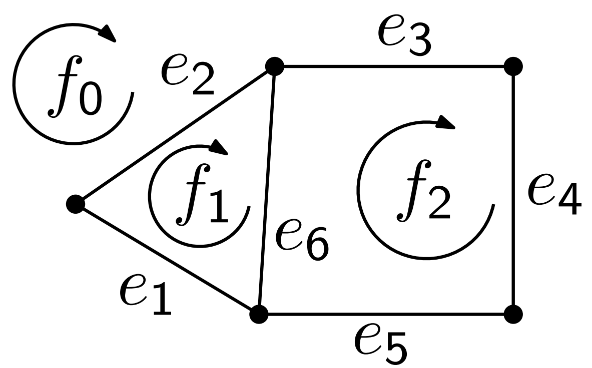
- e ist Randkante von f
- δ ist 0-1-Folge (0 = Rechtsknick, 1 = Linksknick)
- α ist Winkel $\in \{\frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$ zwischen e und Nachfolger e'

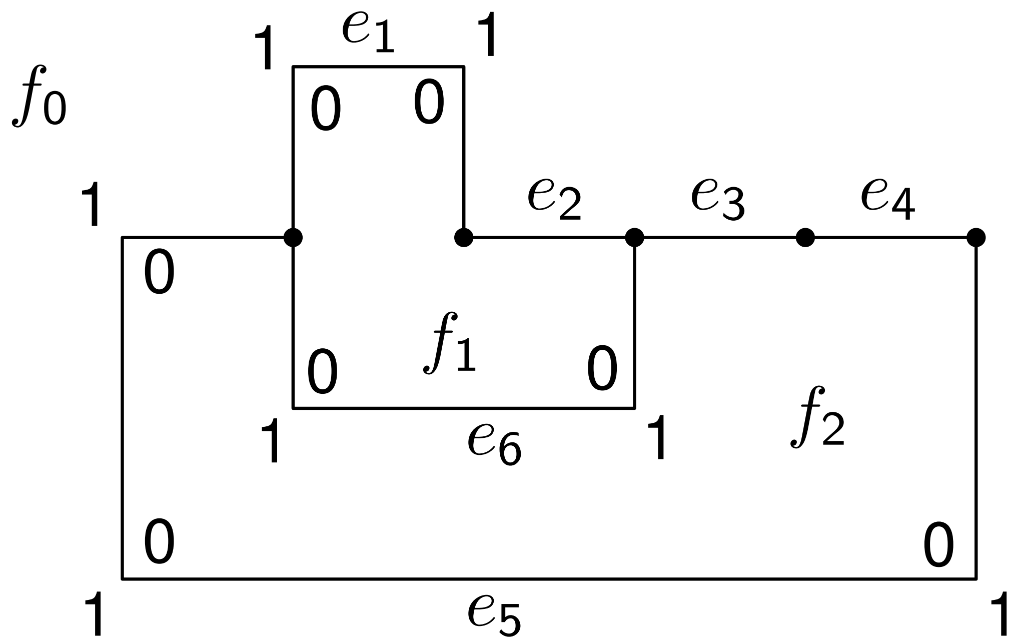
Beispiel: Orthogonale Beschreibung

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

$$H(f_1) = ((e_1, 00, \tfrac{3\pi}{2}), (e_2, \emptyset, \tfrac{\pi}{2}), (e_6, 00, \pi))$$

$$H(f_2) = ((e_5, 000, \tfrac{\pi}{2}), (e_6, 11, \tfrac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \tfrac{\pi}{2}))$$





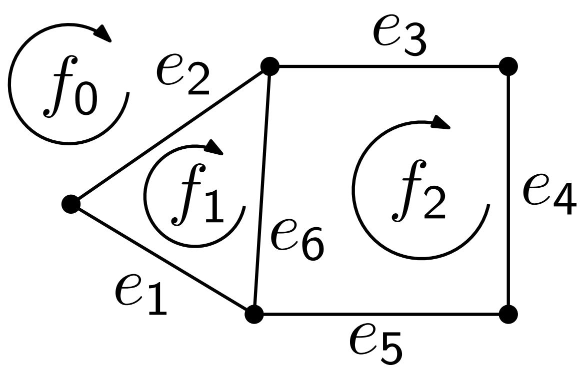
Beispiel: Orthogonale Beschreibung

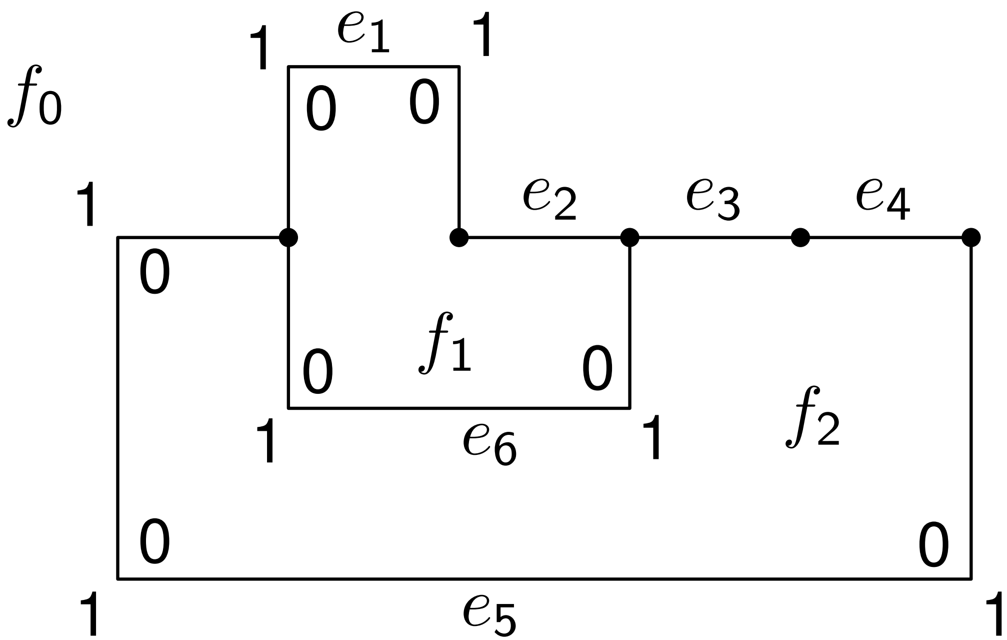
$$H(f_0) = ((e_1, 11, \tfrac{\pi}{2}), (e_5, 111, \tfrac{3\pi}{2}), (e_4, \emptyset, \pi), (e_3, \emptyset, \pi), (e_2, \emptyset, \tfrac{\pi}{2}))$$

$$H(f_1) = ((e_1, 00, \tfrac{3\pi}{2}), (e_2, \emptyset, \tfrac{\pi}{2}), (e_6, 00, \pi))$$

$$H(f_2) = ((e_5, 000, \tfrac{\pi}{2}), (e_6, 11, \tfrac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \tfrac{\pi}{2}))$$

f_0 falsch rum!?





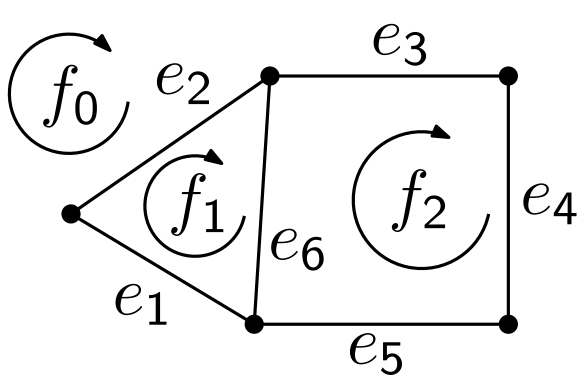
Institute for Theoretical Informatics
Algorithmics Group

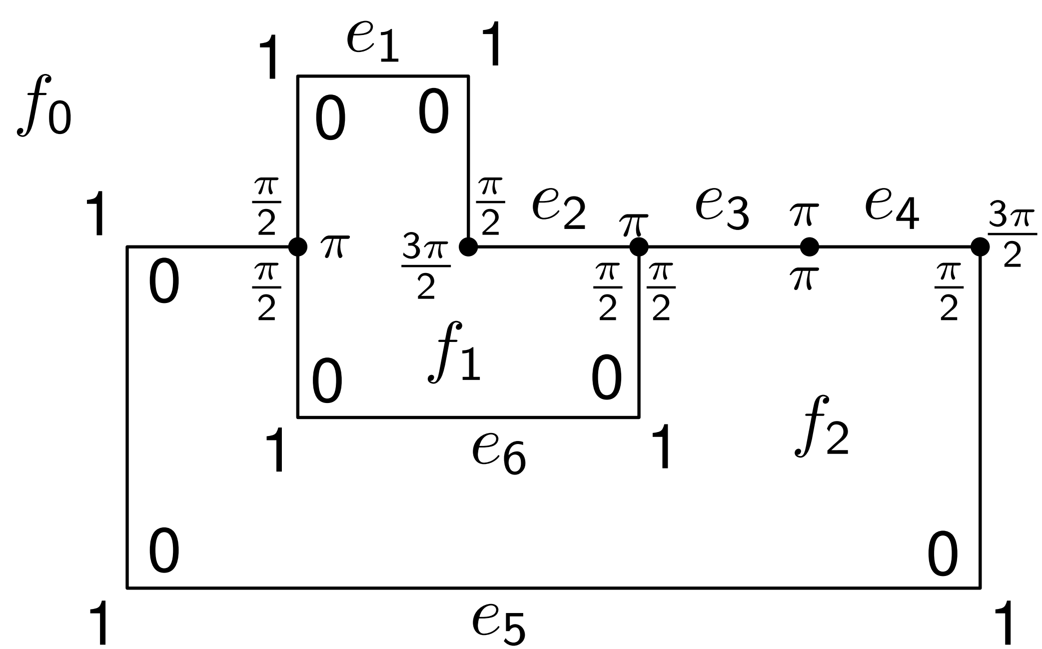
Beispiel: Orthogonale Beschreibung

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

$$H(f_1) = ((e_1, 00, \tfrac{3\pi}{2}), (e_2, \emptyset, \tfrac{\pi}{2}), (e_6, 00, \pi))$$

$$H(f_2) = ((e_5, 000, \tfrac{\pi}{2}), (e_6, 11, \tfrac{\pi}{2}), (e_3, \emptyset, \pi), (e_4, \emptyset, \tfrac{\pi}{2}))$$





Korrektheit einer orthogonalen Beschreibung

(H1) $H(G)$ entspricht $\mathcal{F}, f_0$

Institute for Theoretical Informatics
Algorithmics Group

Korrektheit einer orthogonalen Beschreibung

(H1) $H(G)$ entspricht $\mathcal{F}, f_0$

(H2) für gemeinsame Randkante $\{u, v\}$ zweier Facetten f und g mit $((u, v), \delta_1, \alpha_1) \in H(f)$ und $((v, u), \delta_2, \alpha_2) \in H(g)$ gilt δ_1 ist invertierte und umgedrehte Folge δ_2

Institute for Theoretical Informatics
Algorithmics Group

Korrektheit einer orthogonalen Beschreibung

(H1) $H(G)$ entspricht $\mathcal{F}, f_0$

(H2) für gemeinsame Randkante $\{u, v\}$ zweier Facetten f und g mit $((u, v), \delta_1, \alpha_1) \in H(f)$ und $((v, u), \delta_2, \alpha_2) \in H(g)$ gilt δ_1 ist invertierte und umgedrehte Folge δ_2

(H3) Sei $|\delta|_0$ (bzw. $|\delta|_1$) die Anzahl Nullen (bzw. Einsen) in δ und $r = (e, \delta, \alpha)$. Für $C(r) := |\delta|_0 - |\delta|_1 + 2 - 2\alpha/\pi$ gilt:
$\sum_{r \in H(f)} C(r) = 4$ für $f \neq f_0$ und $\sum_{r \in H(f_0)} C(r) = -4$

Korrektheit einer orthogonalen Beschreibung

(H1) $H(G)$ entspricht $\mathcal{F}$, f_0

(H2) für gemeinsame Randkante $\{u, v\}$ zweier Facetten f und g mit $((u, v), \delta_1, \alpha_1) \in H(f)$ und $((v, u), \delta_2, \alpha_2) \in H(g)$ gilt δ_1 ist invertierte und umgedrehte Folge δ_2

(H3) Sei $|\delta|_0$ (bzw. $|\delta|_1$) die Anzahl Nullen (bzw. Einsen) in δ und $r = (e, \delta, \alpha)$. Für $C(r) := |\delta|_0 - |\delta|_1 + 2 - 2\alpha/\pi$ gilt:
$\sum_{r \in H(f)} C(r) = 4$ für $f \neq f_0$ und $\sum_{r \in H(f_0)} C(r) = -4$

(H4) Für jeden Knoten v ist die Summe der anliegenden Winkel gleich 2π

Institute for Theoretical Informatics
Algorithmics Group

Erinnerung: Problem

Problem 2': Orthogonale Beschreibung

Gegeben ein Graph $G = (V, E)$ mit Maximalgrad $\mathsf{deg}_{\mathsf{max}} \leq 4$, kombinatorischer Einbettung $\mathcal{F}$ und äußerer Facette f_0, finde eine gültige orthogonale Beschreibung $H(G)$, die $(\mathcal{F},\ f_0)$ erhält und die Knickanzahl minimiert.

Erinnerung: Problem

Problem 2': Orthogonale Beschreibung

Gegeben ein Graph $G = (V, E)$ mit Maximalgrad $\deg_{\max} \leq 4$, kombinatorischer Einbettung $\mathcal{F}$ und äußerer Facette f_0, finde eine gültige orthogonale Beschreibung $H(G)$, die $(\mathcal{F},\ f_0)$ erhält und die Knickanzahl minimiert.

Ansatz: Baue Flussnetzwerk!

- Währung = $\measuredangle\ \frac{\pi}{2}$

Erinnerung: Problem

Problem 2': Orthogonale Beschreibung

Gegeben ein Graph $G = (V, E)$ mit Maximalgrad $\mathrm{deg}_{\mathrm{max}} \leq 4$, kombinatorischer Einbettung $\mathcal{F}$ und äußerer Facette f_0, finde eine gültige orthogonale Beschreibung $H(G)$, die $(\mathcal{F},\ f_0)$ erhält und die Knickanzahl minimiert.

Ansatz: Baue Flussnetzwerk!

- Währung = $\measuredangle\ \frac{\pi}{2}$
- Knoten $\xrightarrow{\measuredangle}$ Facetten (# $\frac{\pi}{2}$ zur Facette)
- Facetten $\xrightarrow{\measuredangle}$ Nachbar-Facetten (# Knicke zum Nachbar)

Institute for Theoretical Informatics
Algorithmics Group

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

Institute for Theoretical Informatics
Algorithmics Group

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

- $A = \{(v, f) \in V \times \mathcal{F} \mid v \text{ inzident zu } f\} \cup$
  $\{(f, g) \in \mathcal{F} \times \mathcal{F} \mid f, g \text{ adjazent via Kante } e\}$

Institute for Theoretical Informatics
Algorithmics Group

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

- $A = \{(v, f) \in V \times \mathcal{F} \mid v \text{ inzident zu } f\} \cup$
  $\{(f, g) \in \mathcal{F} \times \mathcal{F} \mid f, g \text{ adjazent via Kante } e\}$
- $b(v) = 4 \quad \forall v \in V$

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

- $A = \{(v, f) \in V \times \mathcal{F} \mid v \text{ inzident zu } f\} \cup$
  $\{(f, g) \in \mathcal{F} \times \mathcal{F} \mid f, g \text{ adjazent via Kante } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = -2(d_G(f) - 2) \quad \forall f \in \mathcal{F} \setminus \{f_0\}$
- $b(f_0) = -2(d_G(f) + 2)$

Institute for Theoretical Informatics
Algorithmics Group

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

$$\left.\begin{array}{l} \bullet\ A = \{(v, f) \in V \times \mathcal{F} \mid v \text{ inzident zu } f\} \cup \\ \qquad \{(f, g) \in \mathcal{F} \times \mathcal{F} \mid f, g \text{ adjazent via Kante } e\} \\ \bullet\ b(v) = 4 \quad \forall v \in V \\ \bullet\ b(f) = -2(d_G(f) - 2) \quad \forall f \in \mathcal{F} \setminus \{f_0\} \\ \bullet\ b(f_0) = -2(d_G(f) + 2) \end{array}\right\} \Rightarrow \sum b \overset{?}{=} 0$$

Institute for Theoretical Informatics
Algorithmics Group

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

- $A = \{(v, f) \in V \times \mathcal{F} \mid v \text{ inzident zu } f\} \cup$
  $\{(f, g) \in \mathcal{F} \times \mathcal{F} \mid f, g \text{ adjazent via Kante } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = -2(d_G(f) - 2) \quad \forall f \in \mathcal{F} \setminus \{f_0\}$
- $b(f_0) = -2(d_G(f) + 2)$

$\Rightarrow \sum b = 0$
(Euler)

Unser Flussnetzwerk $N(G)$

Definition Flussnetzwerk $N(G) = ((V \cup \mathcal{F}, A); \ell; u; b; \text{cost})$

- $A = \{(v, f) \in V \times \mathcal{F} \mid v \text{ inzident zu } f\} \cup \{(f, g) \in \mathcal{F} \times \mathcal{F} \mid f, g \text{ adjazent via Kante } e\}$
- $b(v) = 4 \quad \forall v \in V$
- $b(f) = -2(d_G(f) - 2) \quad \forall f \in \mathcal{F} \setminus \{f_0\}$
- $b(f_0) = -2(d_G(f) + 2)$

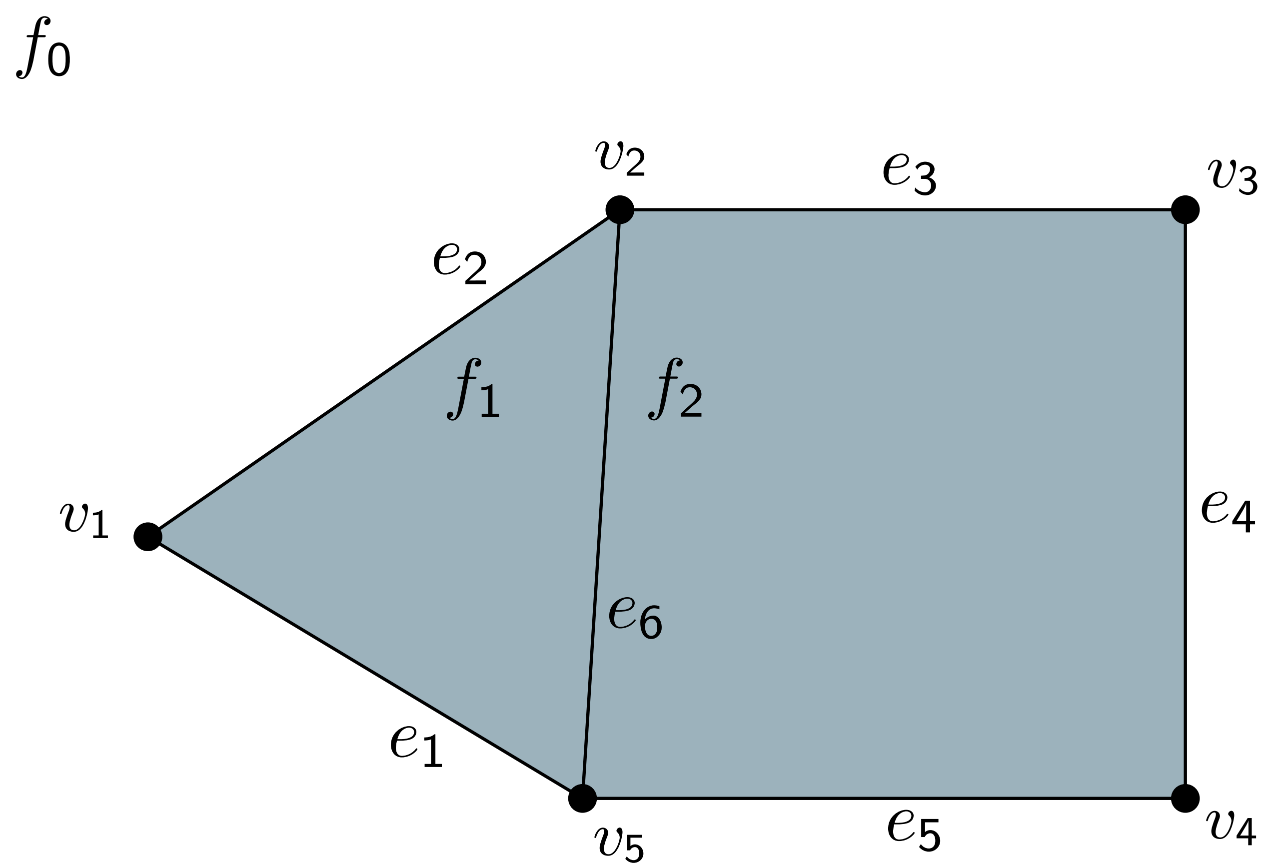
$\Rightarrow \sum b = 0$ (Euler)

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

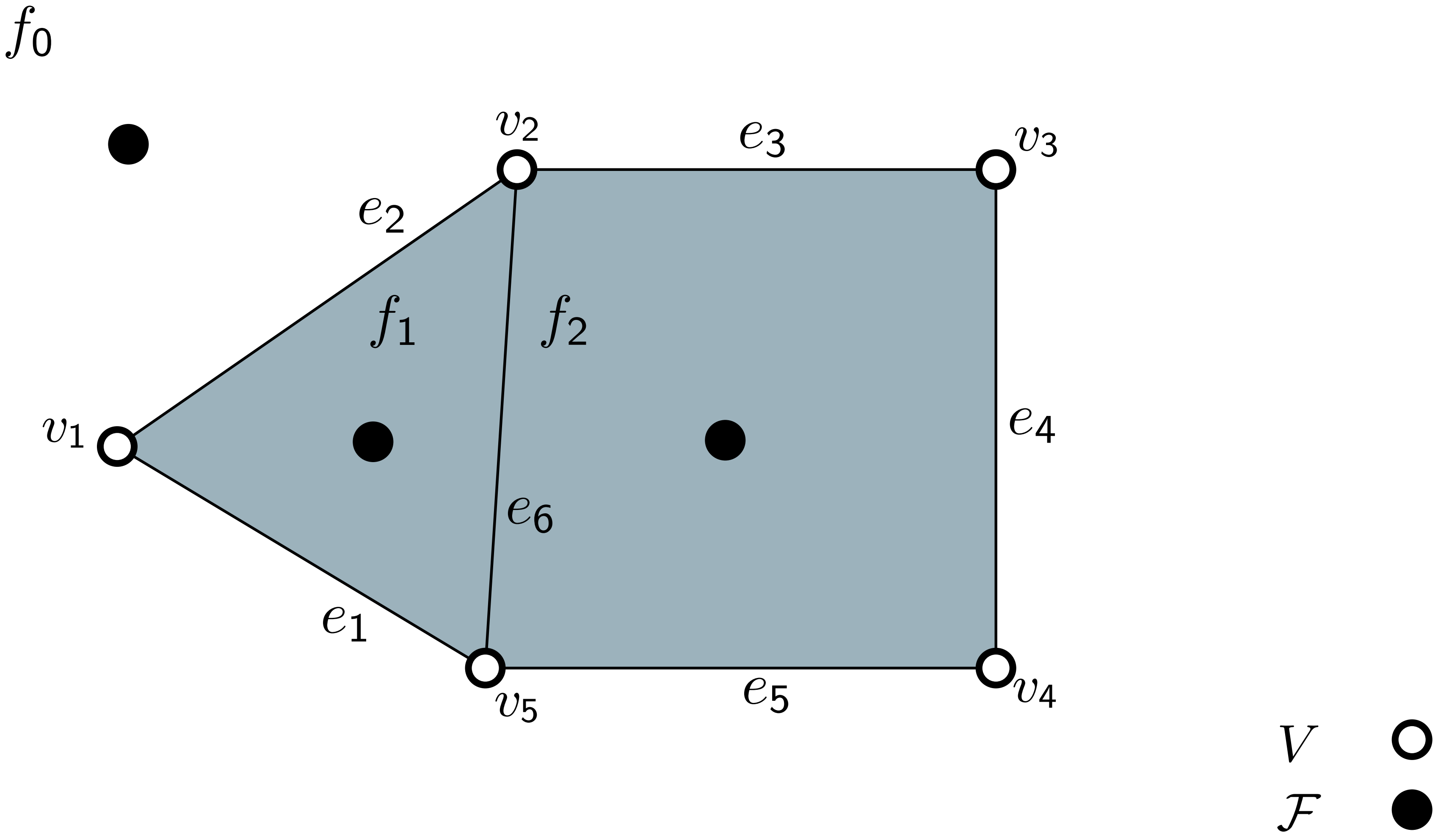
$$\forall (v, f) \in A, v \in V, f \in \mathcal{F} \qquad \ell(v, f) := 1 \leq X(v, f) \leq 4 =: u(v, f)$$

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

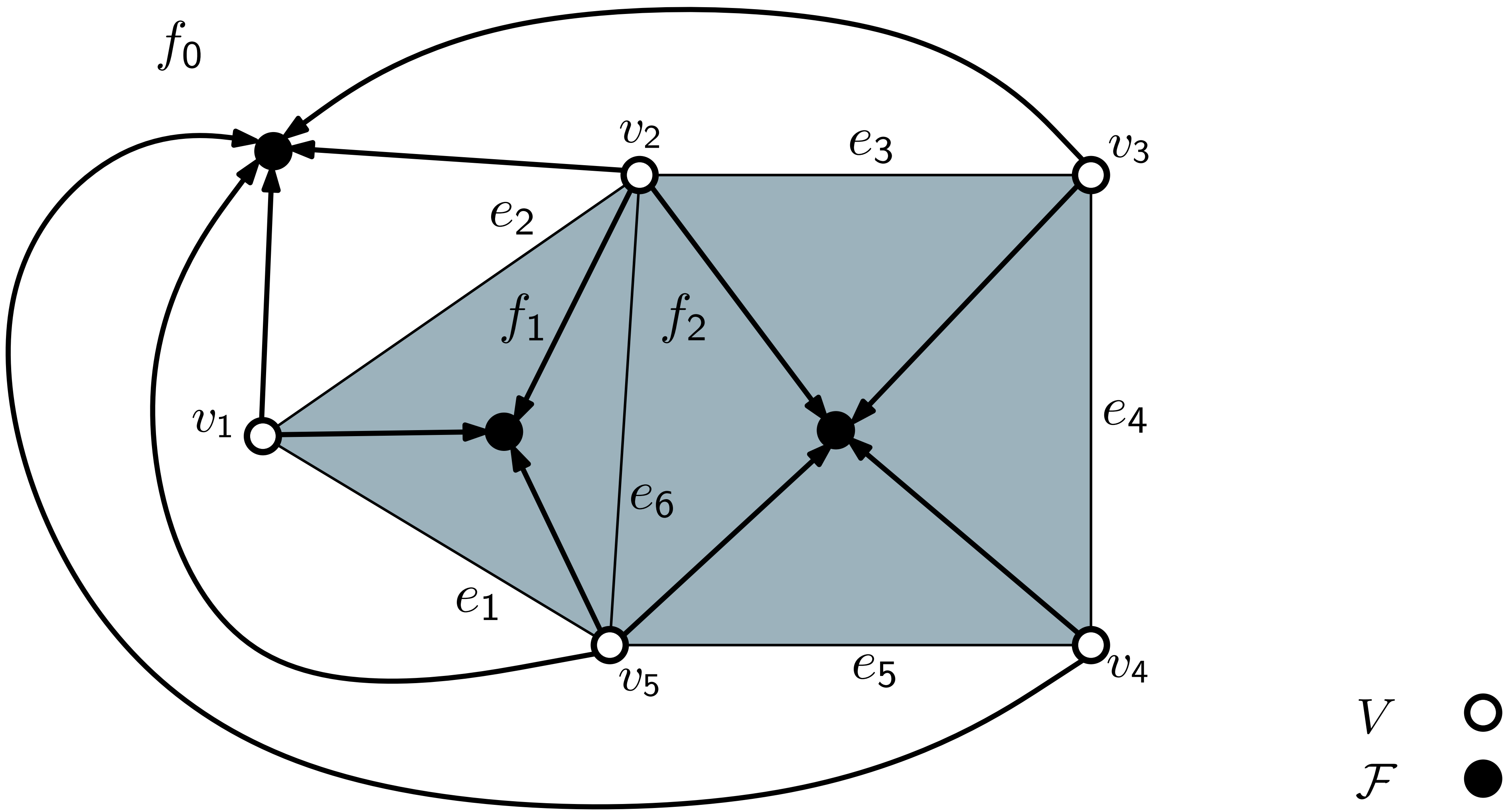
Beispiel Flussnetzwerk



Beispiel Flussnetzwerk

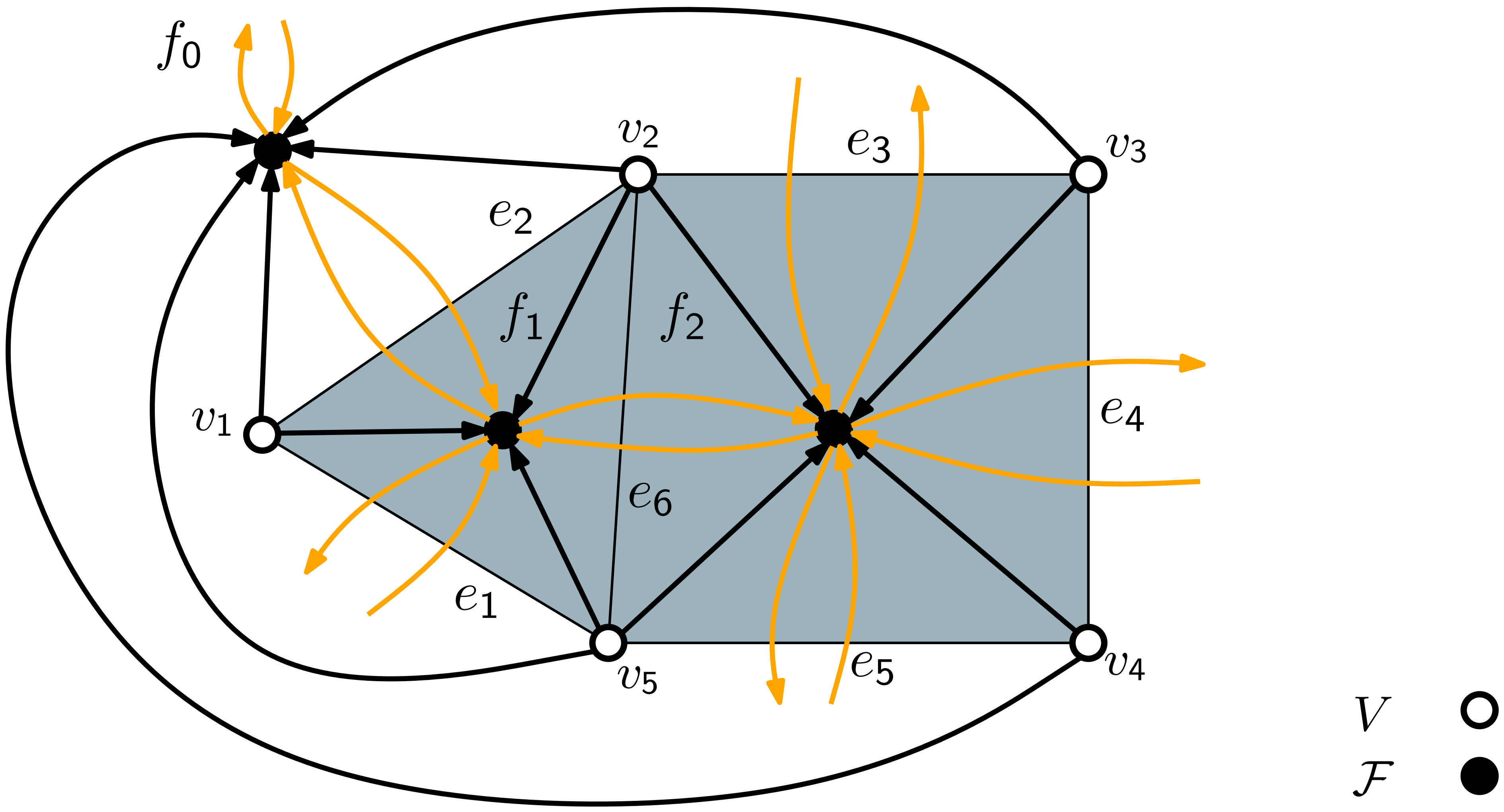


Beispiel Flussnetzwerk

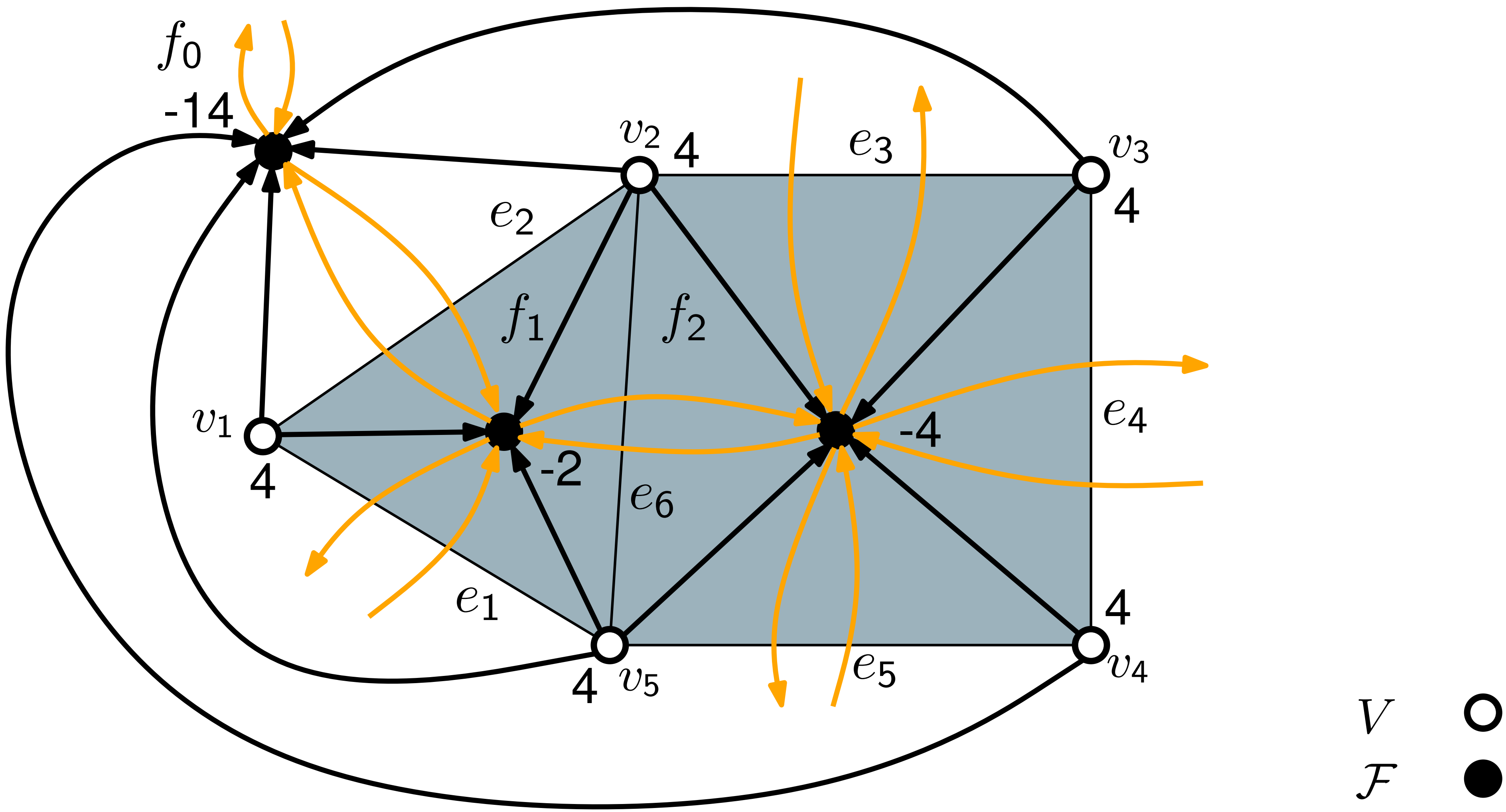


Institute for Theoretical Informatics
Algorithmics Group

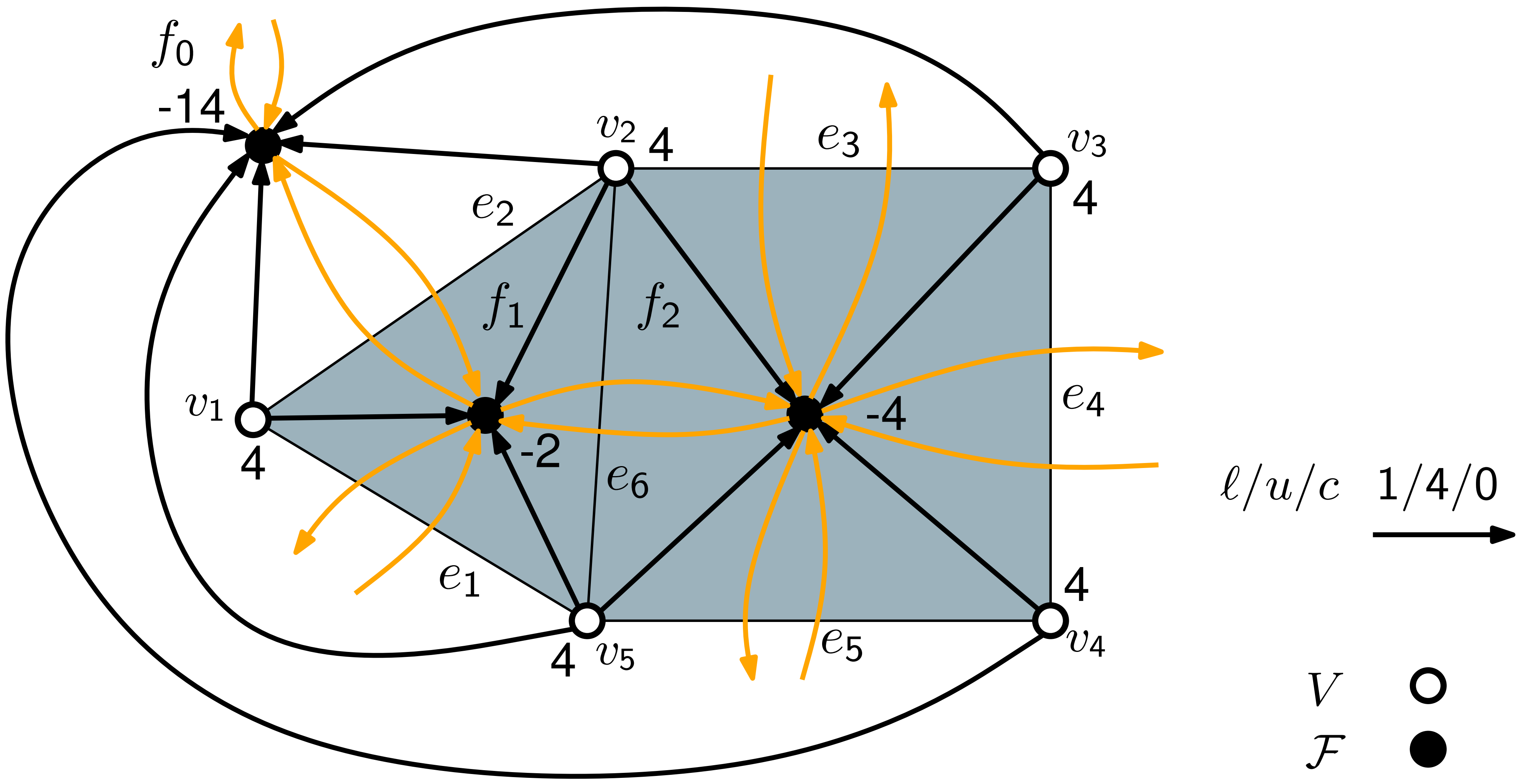
Beispiel Flussnetzwerk



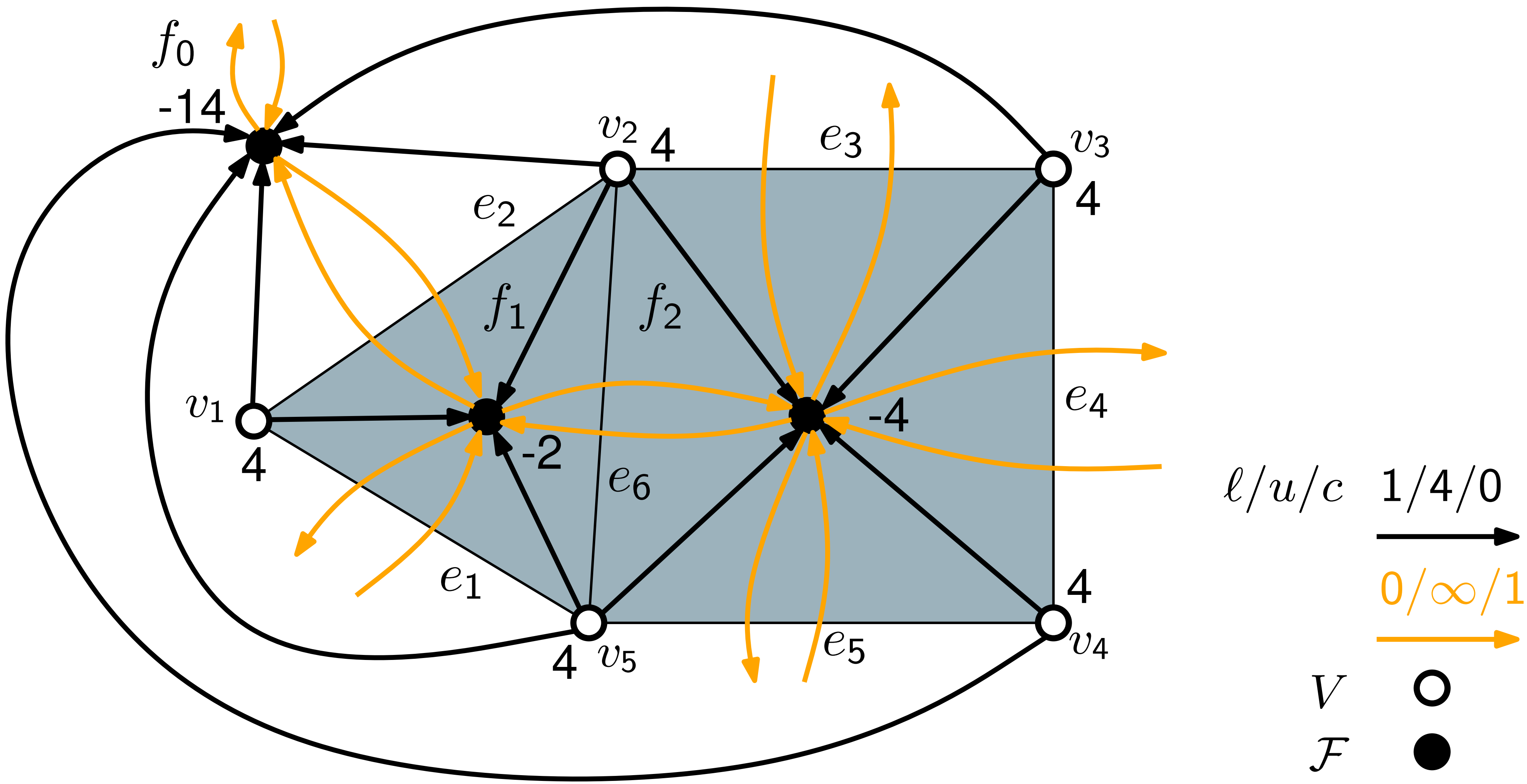
Beispiel Flussnetzwerk



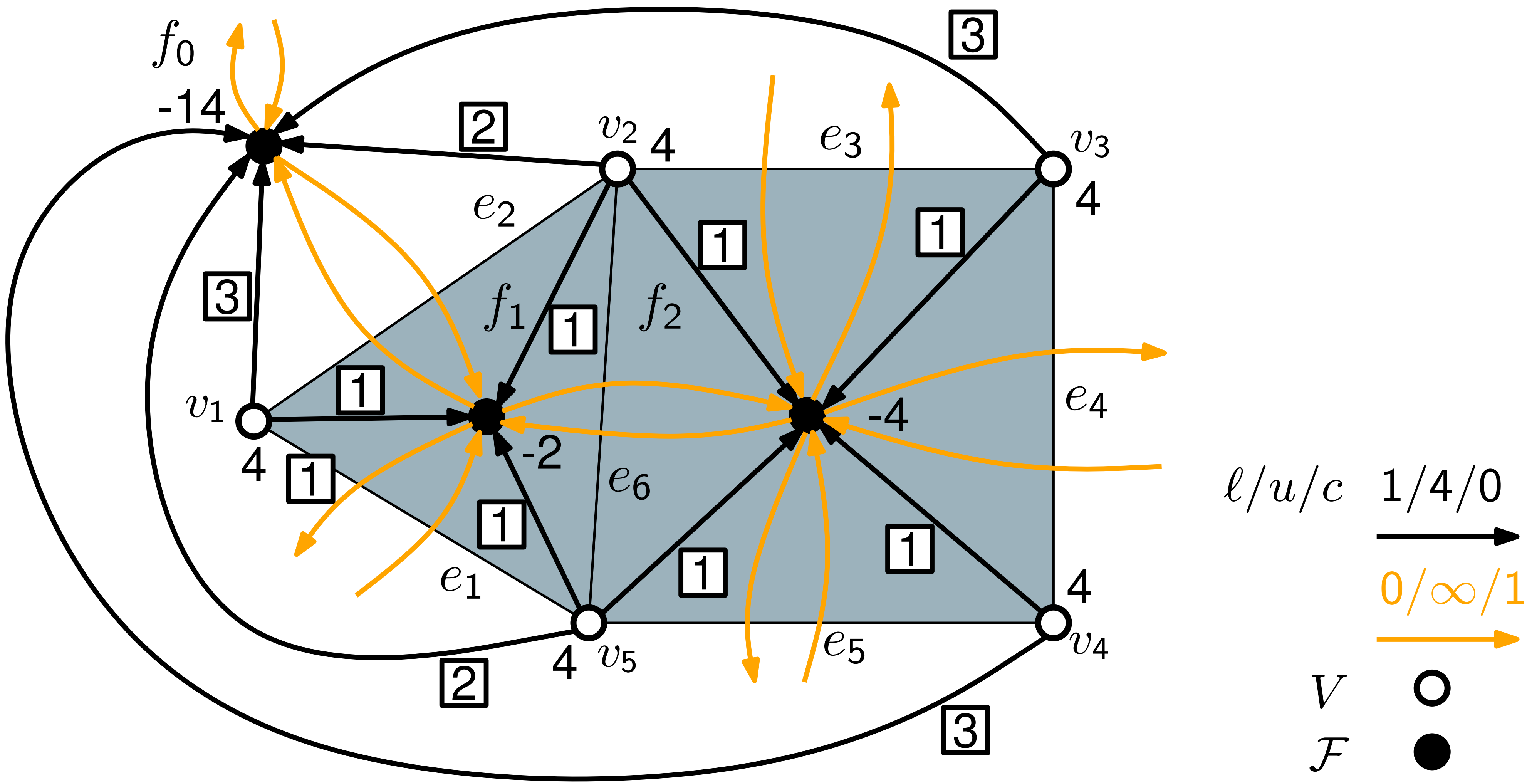
Beispiel Flussnetzwerk



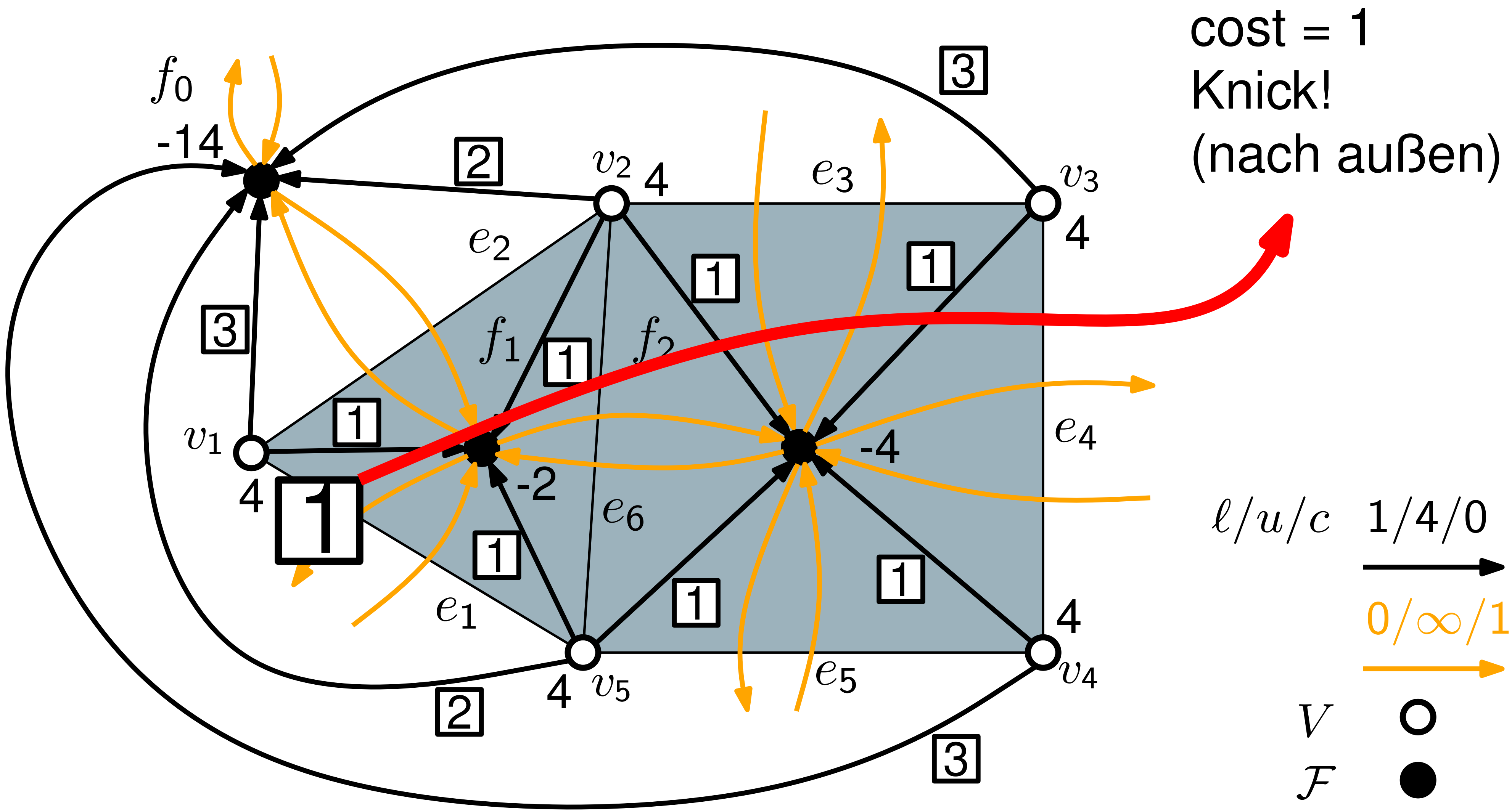
Beispiel Flussnetzwerk



Beispiel Flussnetzwerk



Beispiel Flussnetzwerk



Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Leftarrow$: Geg.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k
Ges.: orthogonale Beschreibung

Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Leftarrow$: Geg.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k
Ges.: orthogonale Beschreibung

(H4) Summe Winkel an Knoten 2π $\surd$

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Leftarrow$: Geg.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k
Ges.: orthogonale Beschreibung

(H2) Kantenbeschr. von anderer Seite invertiert + umgedreht $\surd$

(H4) Summe Winkel an Knoten 2π $\surd$

Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Leftarrow$: Geg.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k
Ges.: orthogonale Beschreibung

(H1) H(G) entspricht $\mathcal{F}, f_0$ $\surd$

(H2) Kantenbeschr. von anderer Seite invertiert + umgedreht $\surd$

(H4) Summe Winkel an Knoten 2π $\surd$

Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Leftarrow$: Geg.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k
Ges.: orthogonale Beschreibung

(H1) H(G) entspricht $\mathcal{F}, f_0$ $\surd$

(H2) Kantenbeschr. von anderer Seite invertiert + umgedreht $\surd$

(H3) Winkelsumme an f: $b(f)$ ohne gestreckte Winkel $= 4$ $\surd$

(H4) Summe Winkel an Knoten 2π $\surd$

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Rightarrow$: Geg.: orthogonale Beschreibung
Ges.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Rightarrow$: Geg.: orthogonale Beschreibung
Ges.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k

(N1) $x(v, f) = 1/2/3/4,$

Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Rightarrow$: Geg.: orthogonale Beschreibung
Ges.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k

(N1) $x(v, f) = 1/2/3/4$,

(N2) $x(f, g) := |\delta_{(f,g)}|_0$, $(e, \delta_{(f,g)}, x)$ Beschr. v. $e \stackrel{*}{=} (f, g)$ für f

Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Rightarrow$: Geg.: orthogonale Beschreibung
Ges.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k

(N1) $x(v, f) = 1/2/3/4$,

(N2) $x(f, g) := |\delta_{(f,g)}|_0$, $(e, \delta_{(f,g)}, x)$ Beschr. v. $e \stackrel{*}{=} (f, g)$ für f

(N3) Kapazitäten $\surd$, Flusserhaltung $\surd$

Institute for Theoretical Informatics
Algorithmics Group

Kernaussage + Korrektheit

Satz

Zu einem planar eingebetteten Graphen $(G, \mathcal{F}, f_0)$ existiert genau dann eine zulässige orthogonale Beschreibung $H(G)$ mit k Knicken, wenn es im Flussnetzwerk $N(G)$ einen Fluss x mit Kosten k gibt.

$\Rightarrow$: Geg.: orthogonale Beschreibung
Ges.: Flussnetzwerk $N(G)$, Fluss x mit Kosten k

(N1) $x(v, f) = 1/2/3/4$,

(N2) $x(f, g) := |\delta_{(f,g)}|_0$, $(e, \delta_{(f,g)}, x)$ Beschr. v. $e \stackrel{*}{=} (f, g)$ für f

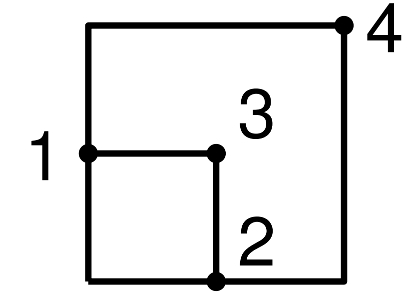
(N3) Kapazitäten $\surd$, Flusserhaltung $\surd$

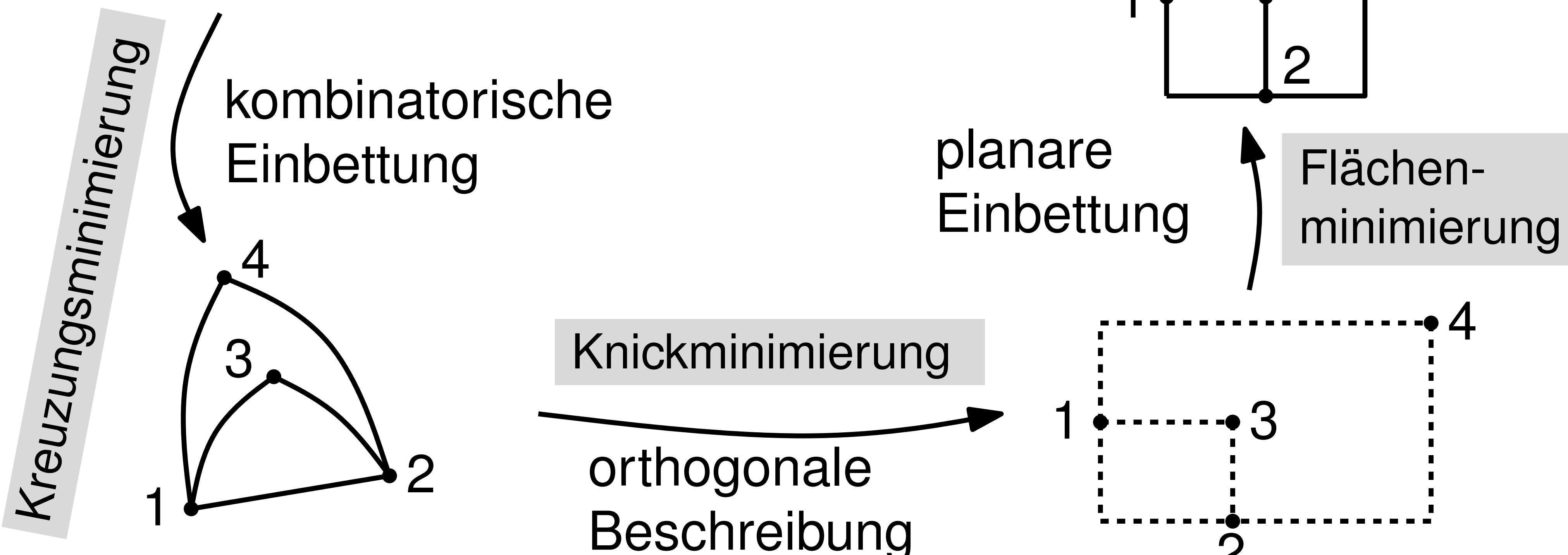
(N4) Kosten = k $\surd$

(Planare) Orthogonale Zeichnungen

Dreistufiger Ansatz: *Topology – Shape – Metrics*

$$E = \{ \ V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$$

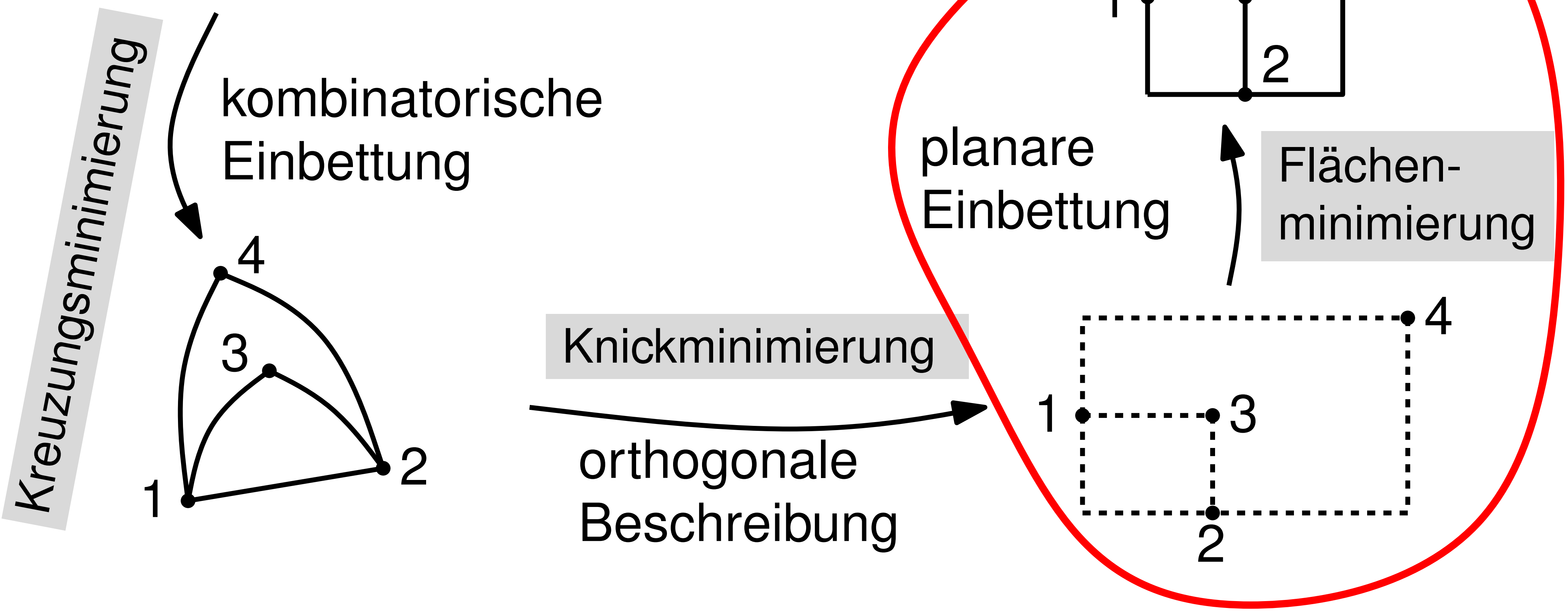




(Planare) Orthogonale Zeichnungen

Dreistufiger Ansatz: *Topology – Shape – Metrics*

$$E = \{\ V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}\}$$



Kompaktierung

Problem: Kompaktes Layout zu orthogonaler Beschreibung

Gegeben ein planerer Graph $G = (V, E)$ mit $\deg_{\max} \leq 4$ und eine orthogonale Beschreibung $H(G)$. Finde ein orthogonales Layout von G, das $H(G)$ realisiert.

Institute for Theoretical Informatics
Algorithmics Group

Kompaktierung

Problem: Kompaktes Layout zu orthogonaler Beschreibung

Gegeben ein planerer Graph $G = (V, E)$ mit $\deg_{\max} \leq 4$ und eine orthogonale Beschreibung $H(G)$. Finde ein orthogonales Layout von G, das $H(G)$ realisiert.

Spezialfall: alle Facetten sind Rechtecke

$\Rightarrow$: erlaubt Garantien:
- minimale Gesamtkantenlänge
- minimale Fläche

Institute for Theoretical Informatics
Algorithmics Group

Kompaktierung

Problem: Kompaktes Layout zu orthogonaler Beschreibung

Gegeben ein planerer Graph $G = (V, E)$ mit $\mathrm{deg}_{\mathrm{max}} \leq 4$ und eine orthogonale Beschreibung $H(G)$. Finde ein orthogonales Layout von G, das $H(G)$ realisiert.

Spezialfall: alle Facetten sind Rechtecke

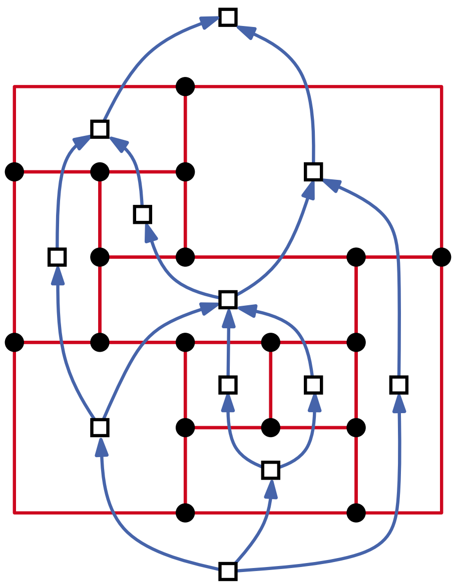
$\Rightarrow$: erlaubt Garantien:
- minimale Gesamtkantenlänge
- minimale Fläche

- Knicke sind außen
- gegenüberliegende Seiten gleich lang $\Rightarrow$ Layout ok

Flussnetzwerk Längenzuweisung

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

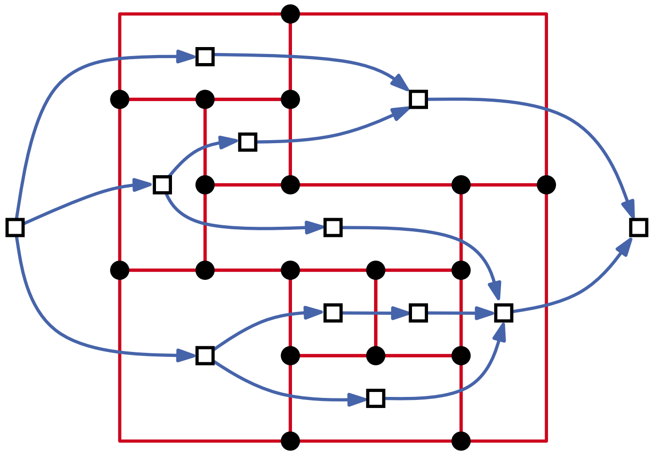
- $W_{\text{hor}} = \mathcal{F}$
- $A_{\text{hor}} = \{(f, g) \mid f, g$ besitzen gemeinsames horizontales Kantensegment und f liegt unterhalb von $g\}$
- $\ell(a) = 1 \quad \forall a \in A_{\text{hor}}$
- $u(a) = \infty \quad \forall a \in A_{\text{hor}}$
- $\text{cost}(a) = 1 \quad \forall a \in A_{\text{hor}}$
- $b(f) = 0 \quad \forall f \in W_{\text{hor}}$



Flussnetzwerk Längenzuweisung

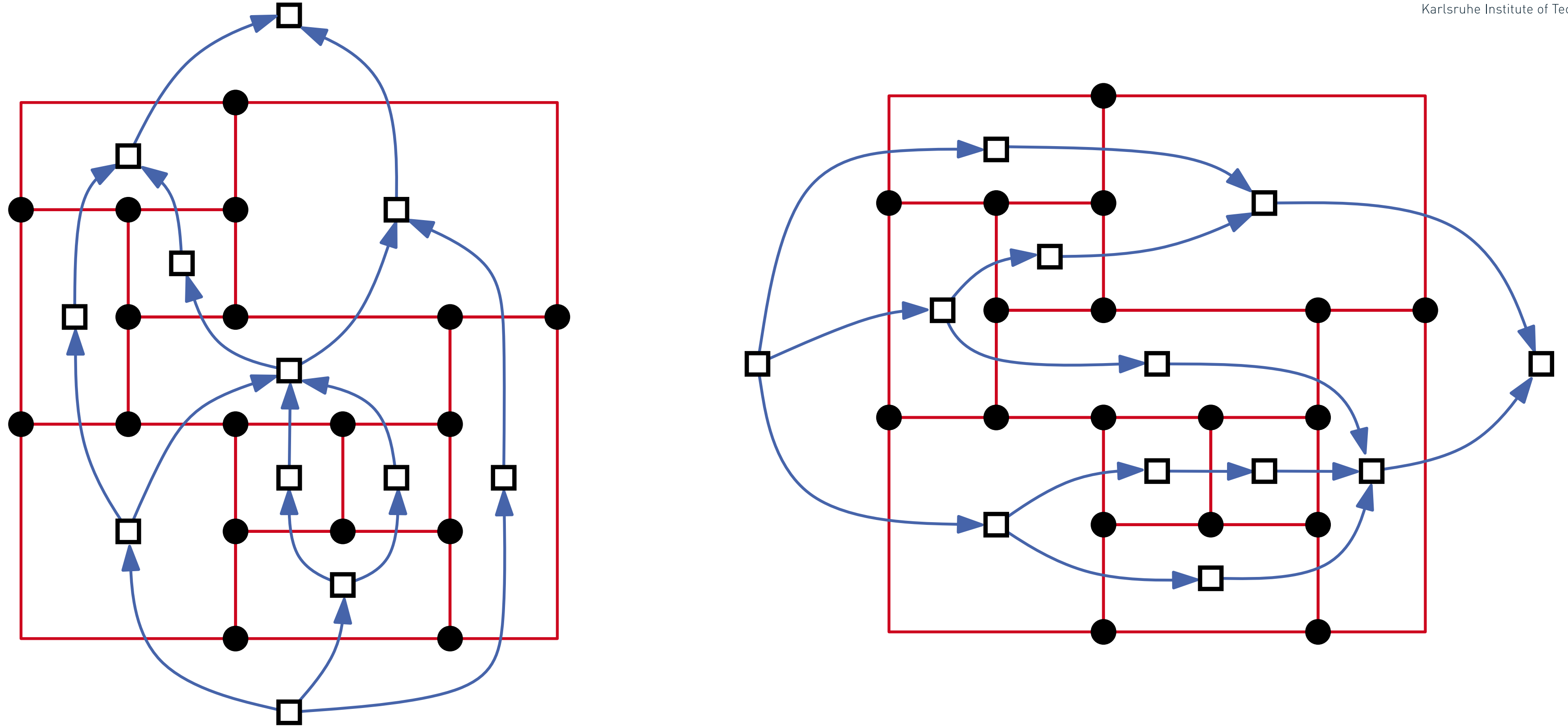
Definition Flussnetzwerk $N_{\text{ver}} = ((W_{\text{ver}}, A_{\text{ver}}); \ell; u; b; \text{cost})$

- $W_{\text{ver}} = \mathcal{F}$
- $A_{\text{ver}} = \{(f, g) \mid f, g$ besitzen gemeinsames vertikales Kantensegment und f liegt links von $g\}$
- $\ell(a) = 1 \quad \forall a \in A_{\text{ver}}$
- $u(a) = \infty \quad \forall a \in A_{\text{ver}}$
- $\text{cost}(a) = 1 \quad \forall a \in A_{\text{ver}}$
- $b(f) = 0 \quad \forall f \in W_{\text{ver}}$



Institute for Theoretical Informatics
Algorithmics Group

Optimal bei Rechtecken

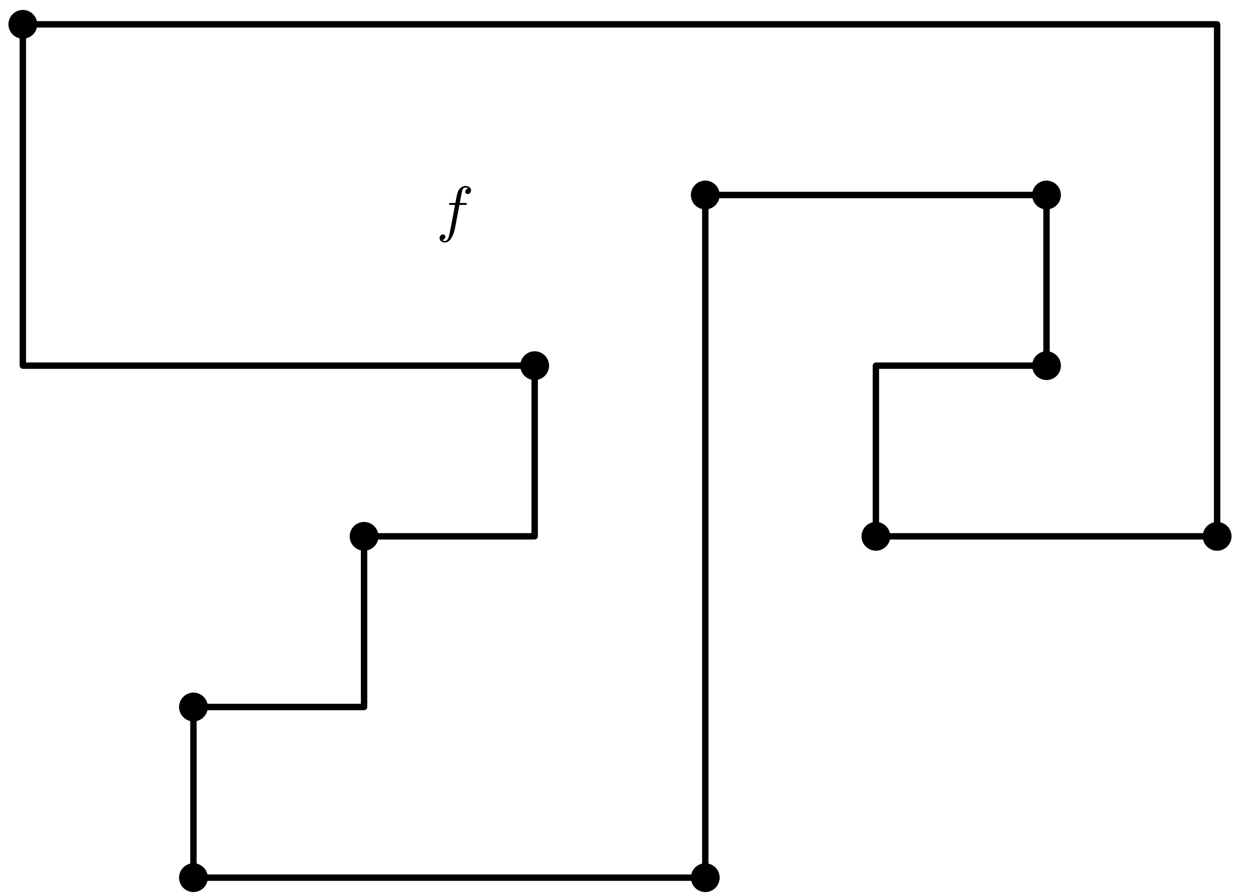


Satz

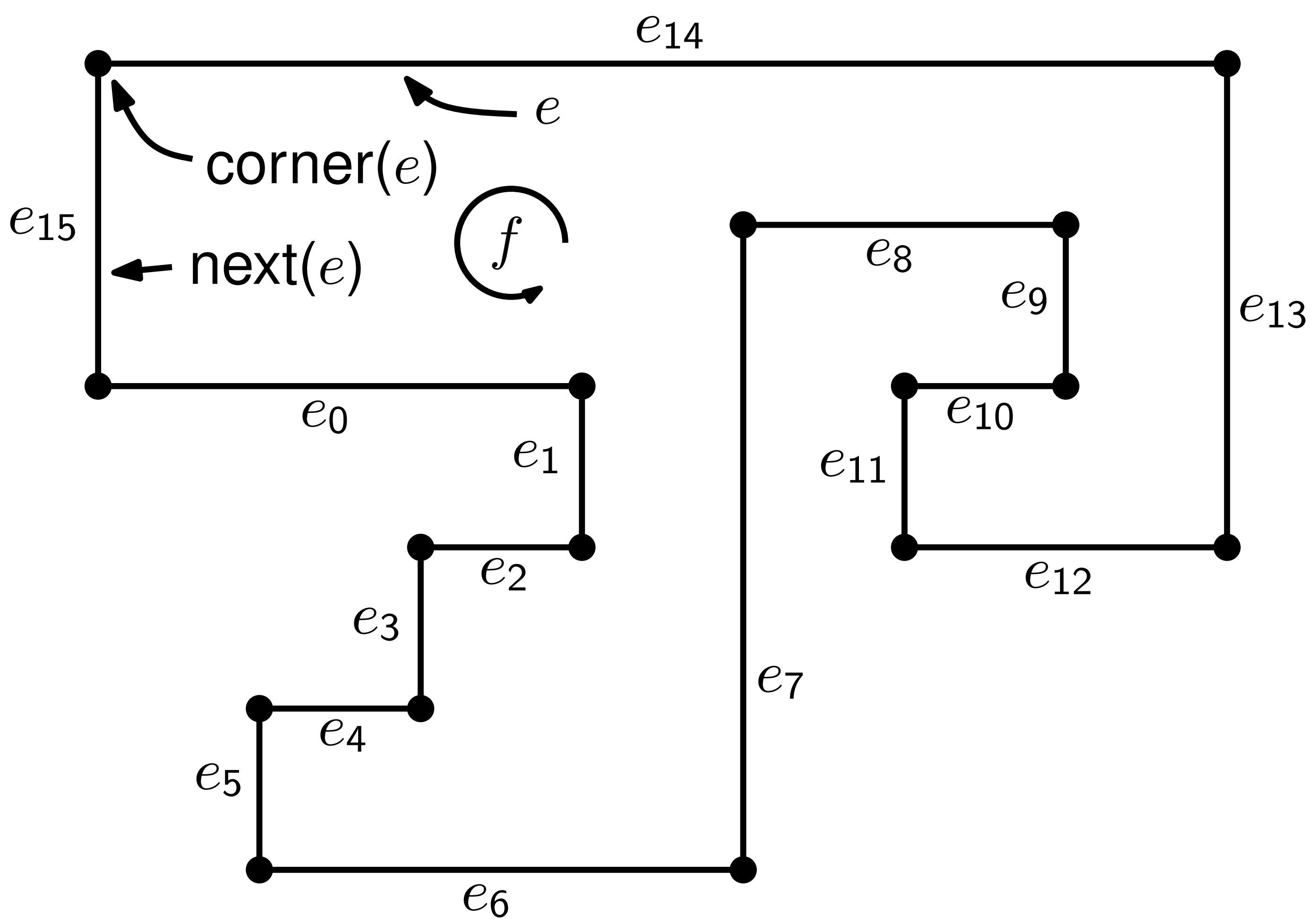
Lösung für Min-Cost Flüsse für N_{hor} und N_{vert} liefert:

1. x Fluss $\Leftrightarrow$ entspr. Kantenlängen induzieren Layout
2. $|x(N_{\text{hor}})| =$Höhe, $|x(N_{\text{ver}})| =$Breite
3. $\text{cost}(x(N_{\text{hor}})) + \text{cost}(x(N_{\text{ver}})) =$ Gesamtkantenlänge

Verfeinerung von (G, H) – innere Facette

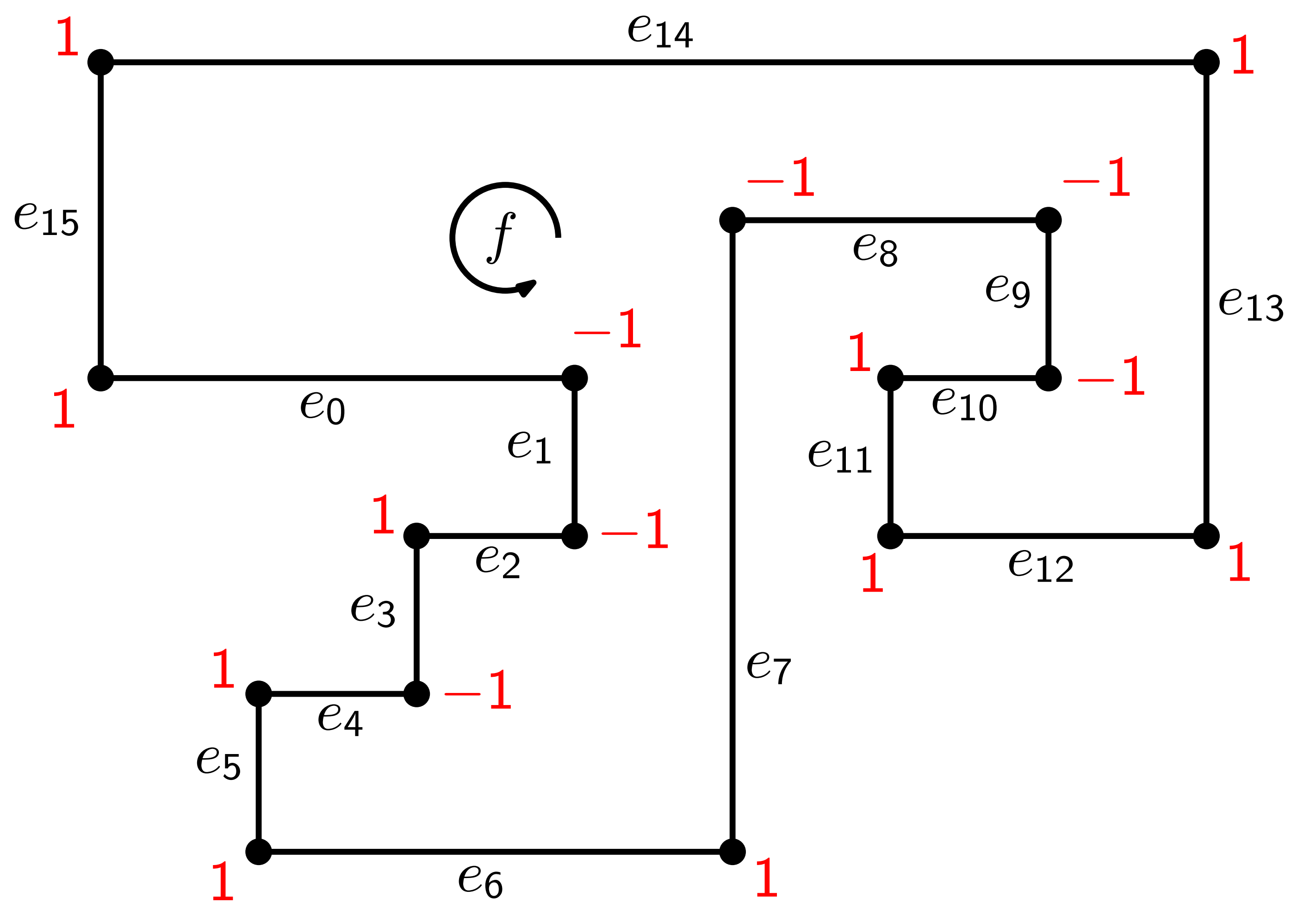


Verfeinerung von (G, H) – innere Facette



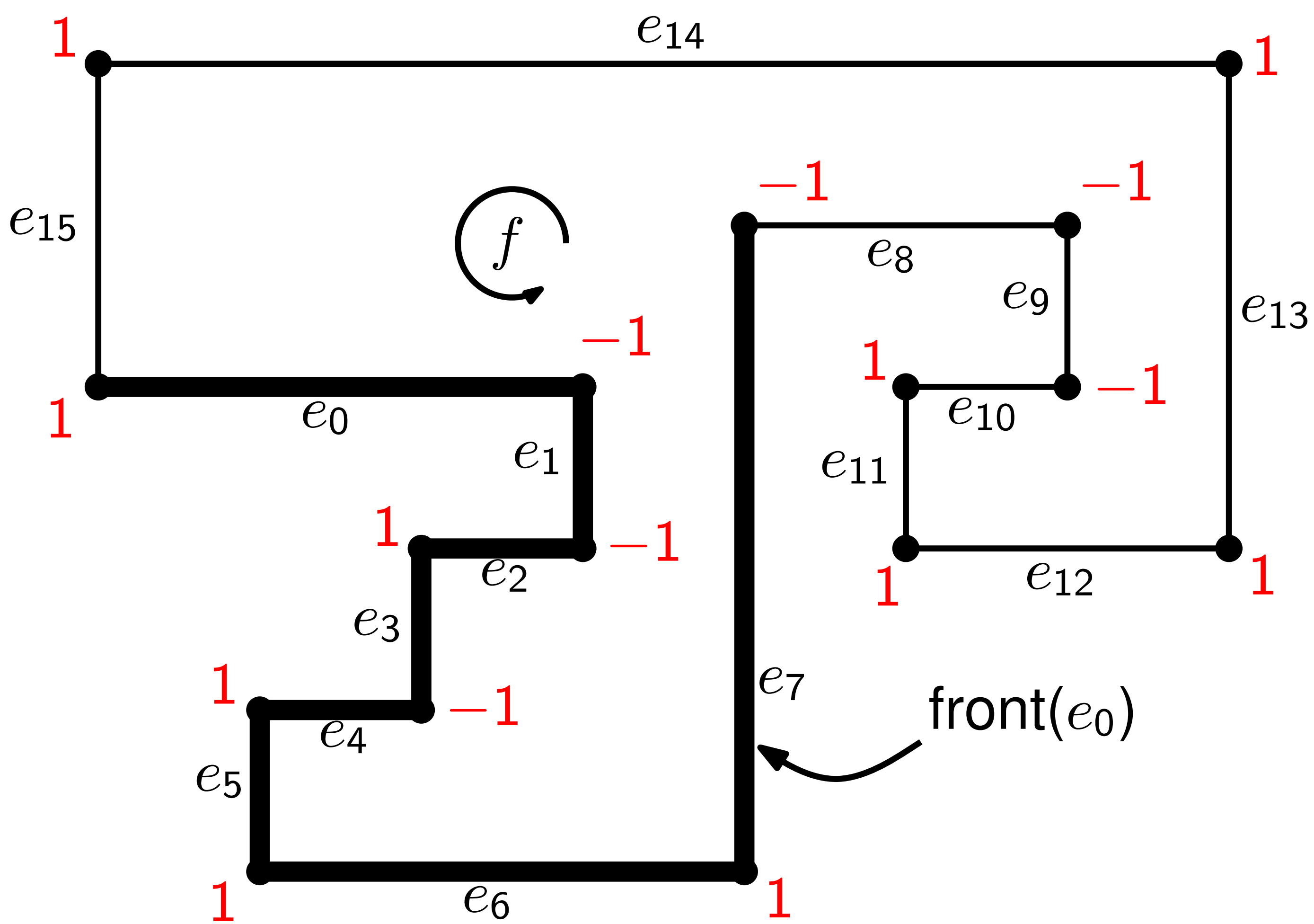
Institute for Theoretical Informatics
Algorithmics Group

Verfeinerung von (G, H) – innere Facette

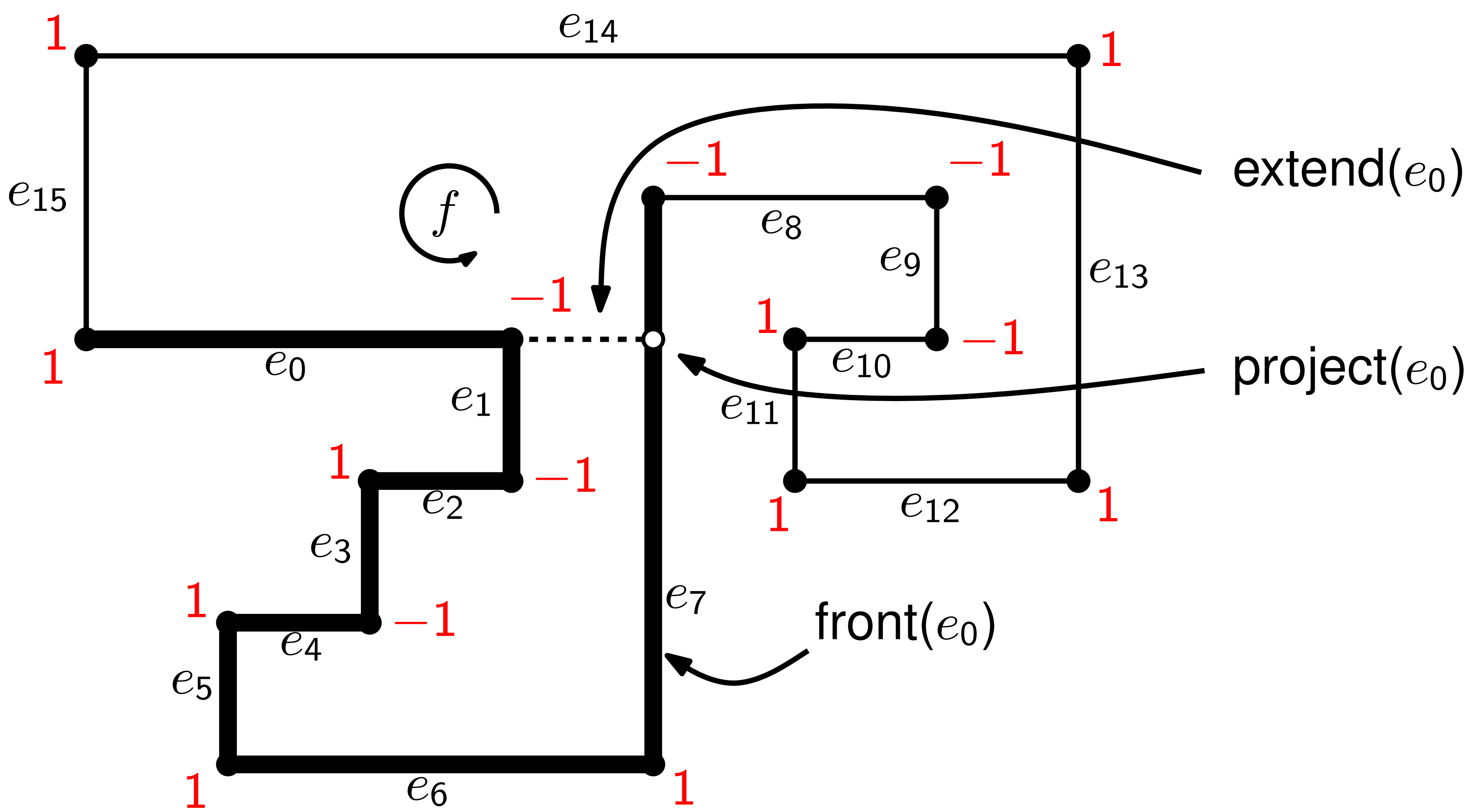


Institute for Theoretical Informatics
Algorithmics Group

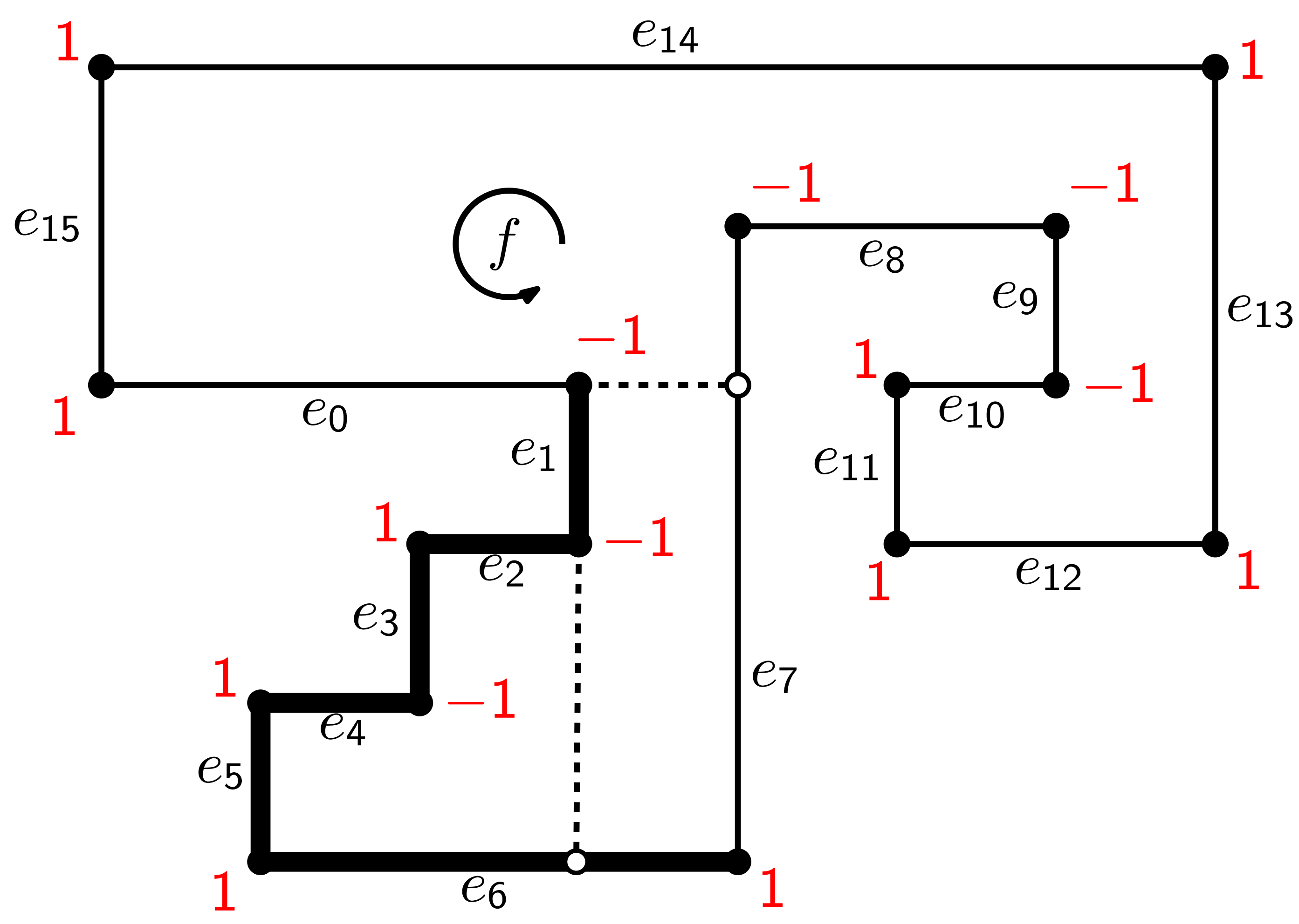
Verfeinerung von (G, H) – innere Facette



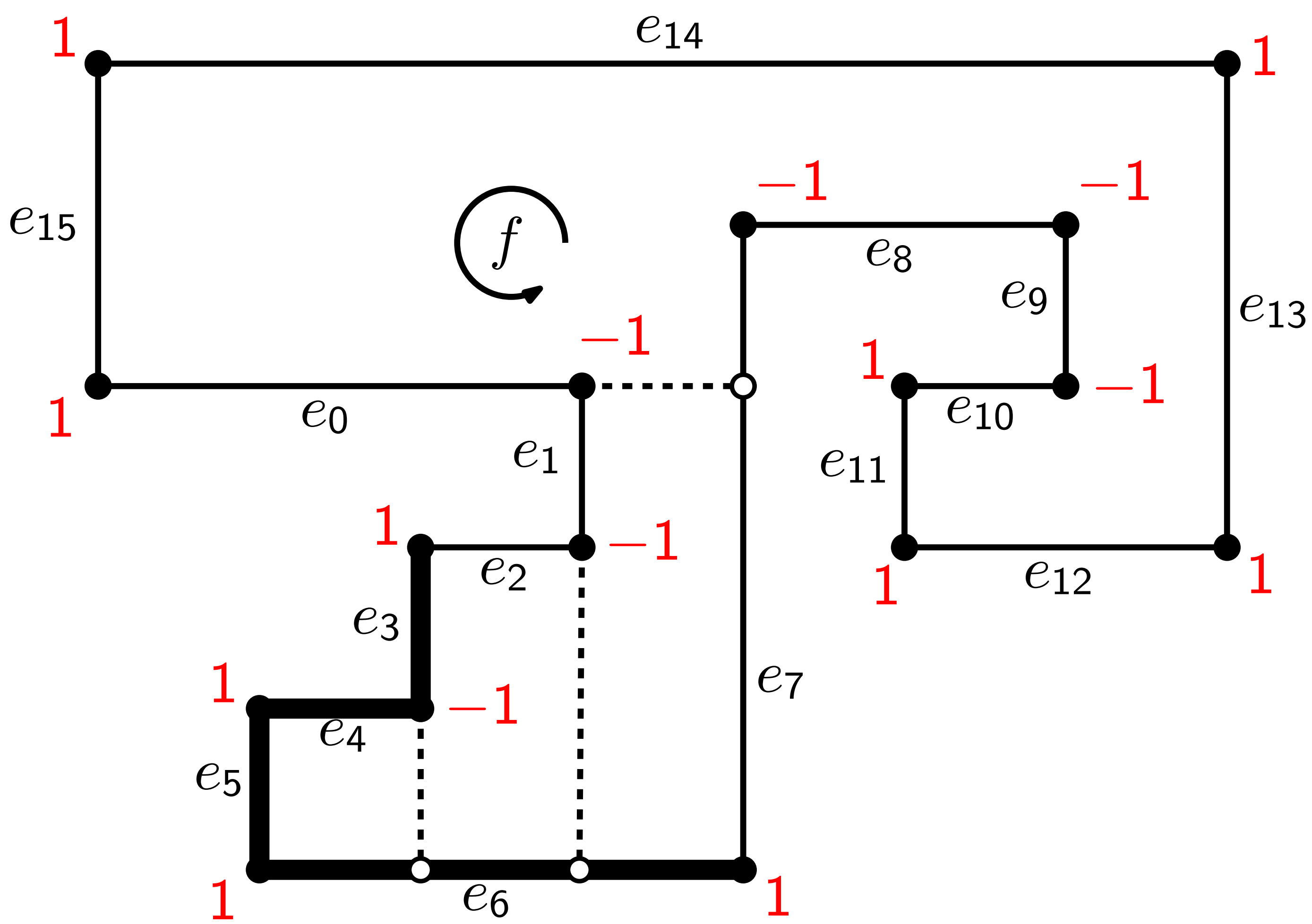
Verfeinerung von (G, H) – innere Facette



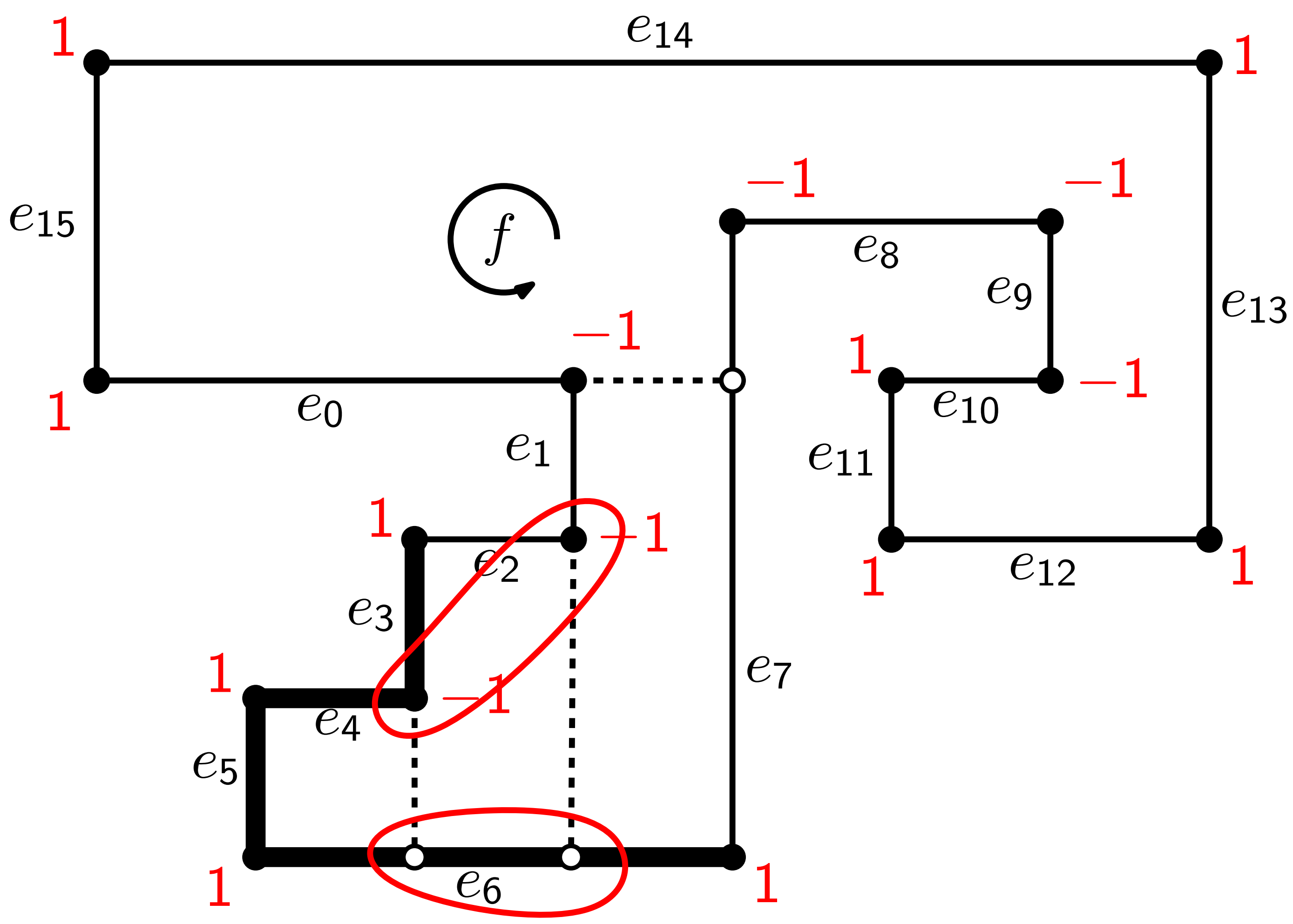
Verfeinerung von (G, H) – innere Facette



Verfeinerung von (G, H) – innere Facette

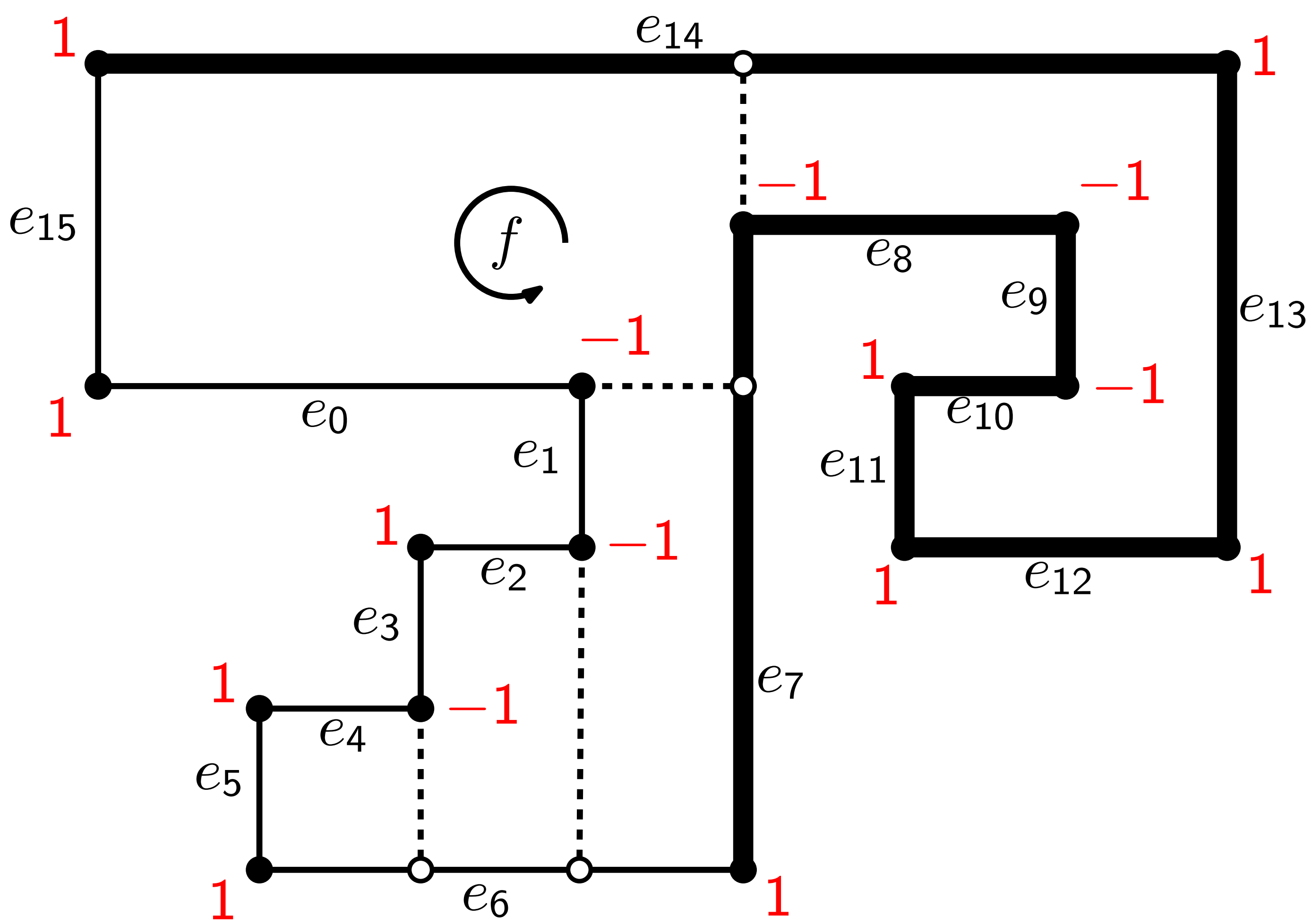


Verfeinerung von (G, H) – innere Facette

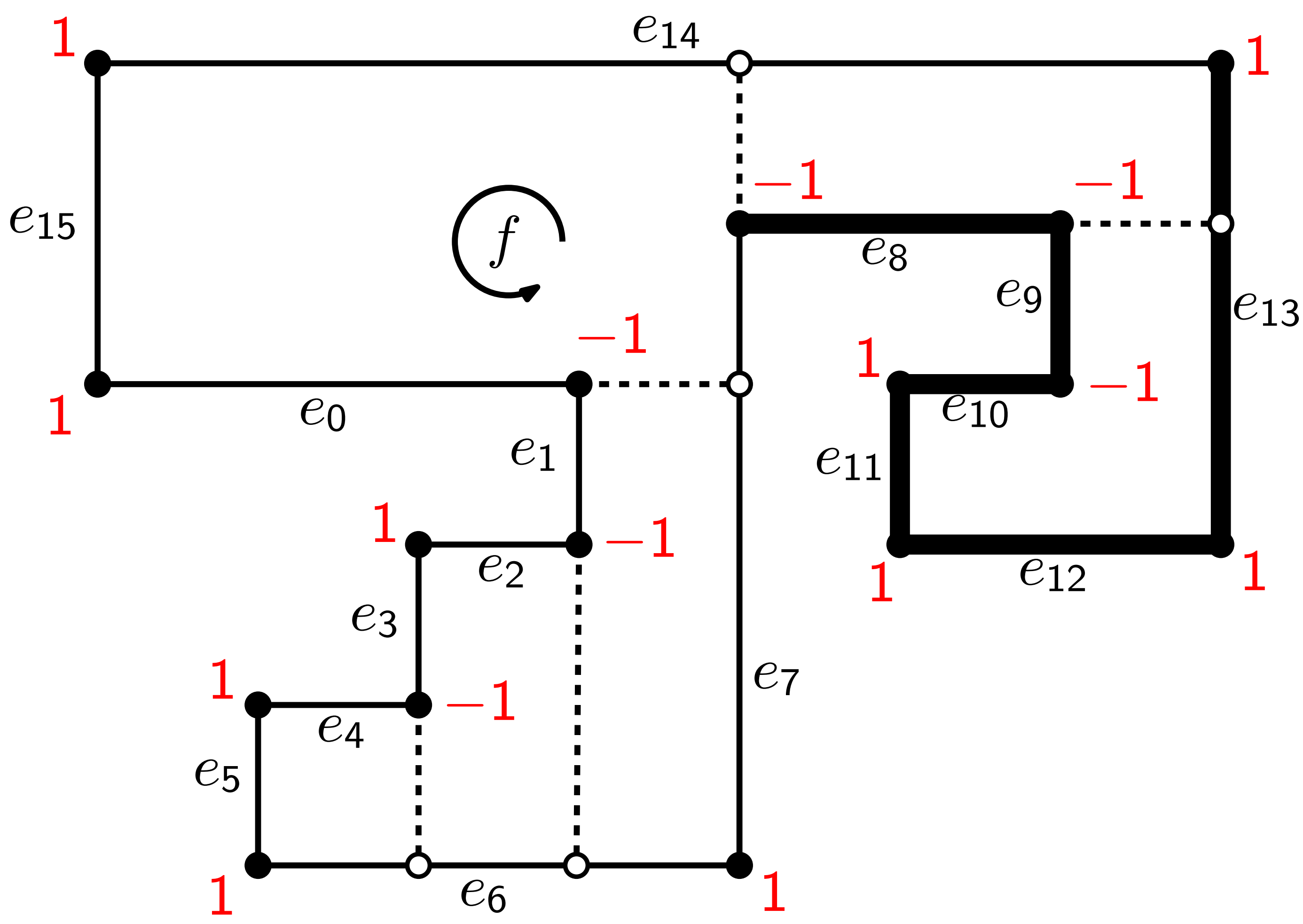


Institute for Theoretical Informatics
Algorithmics Group

Verfeinerung von (G, H) – innere Facette

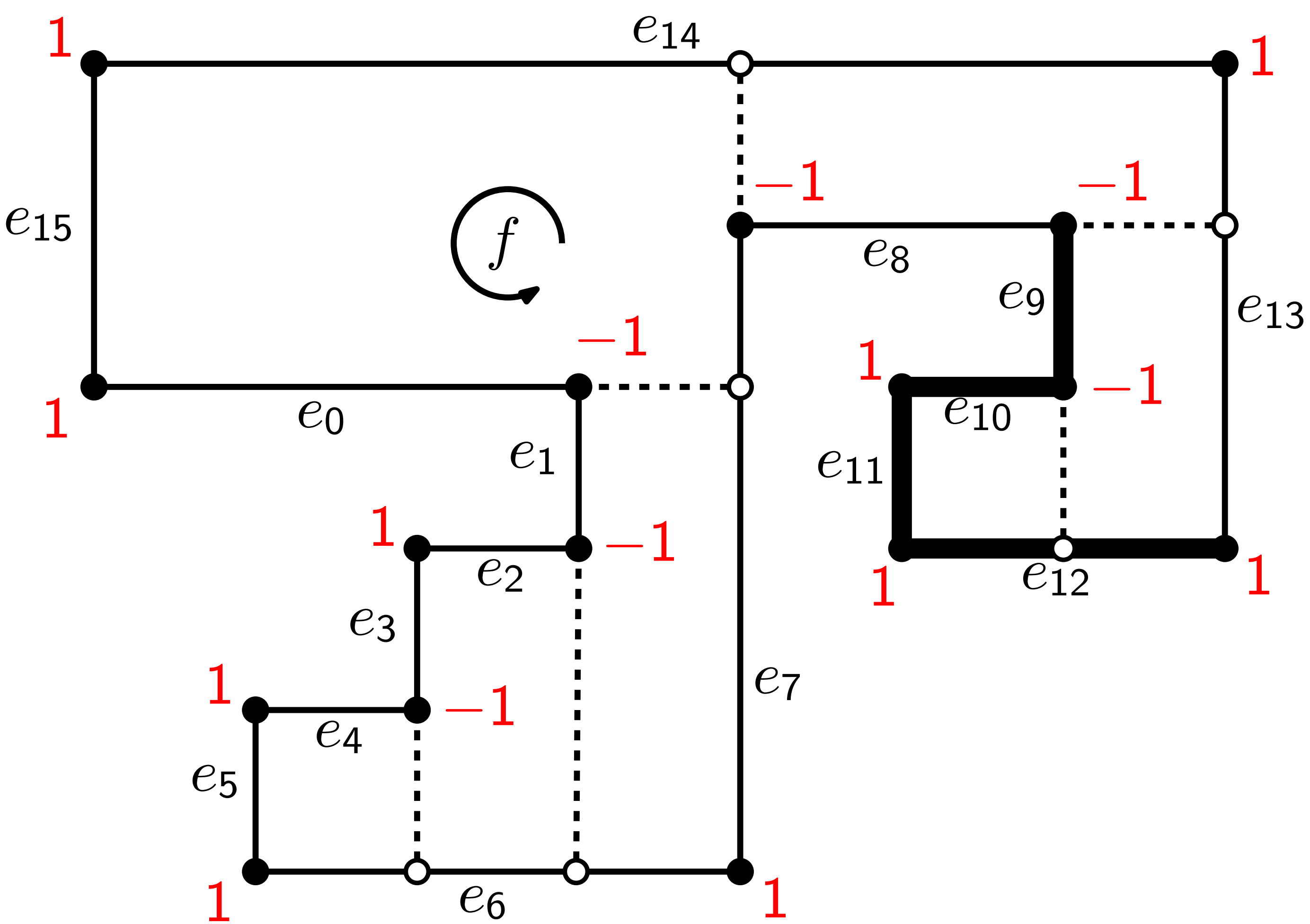


Verfeinerung von (G, H) – innere Facette

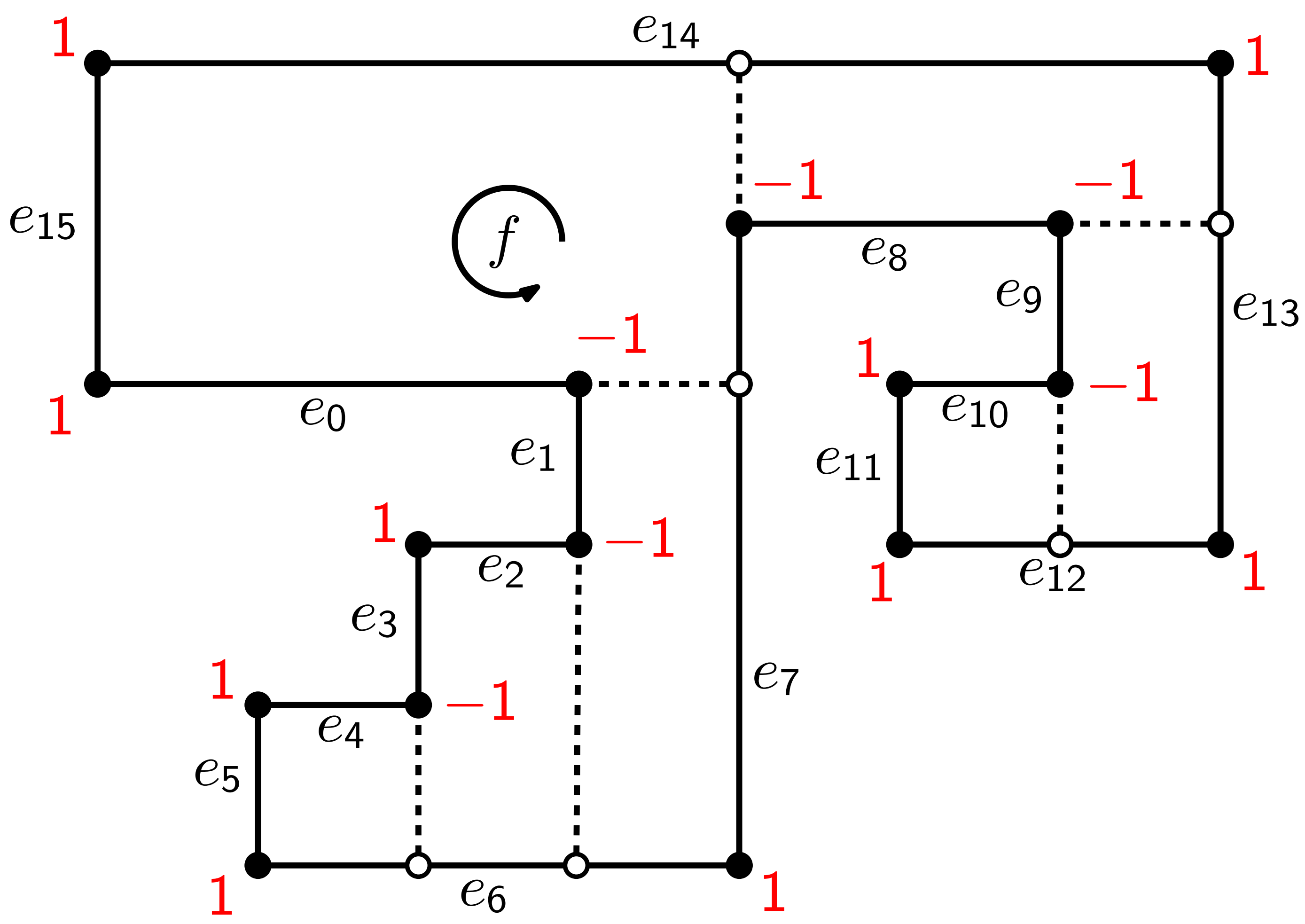


Institute for Theoretical Informatics
Algorithmics Group

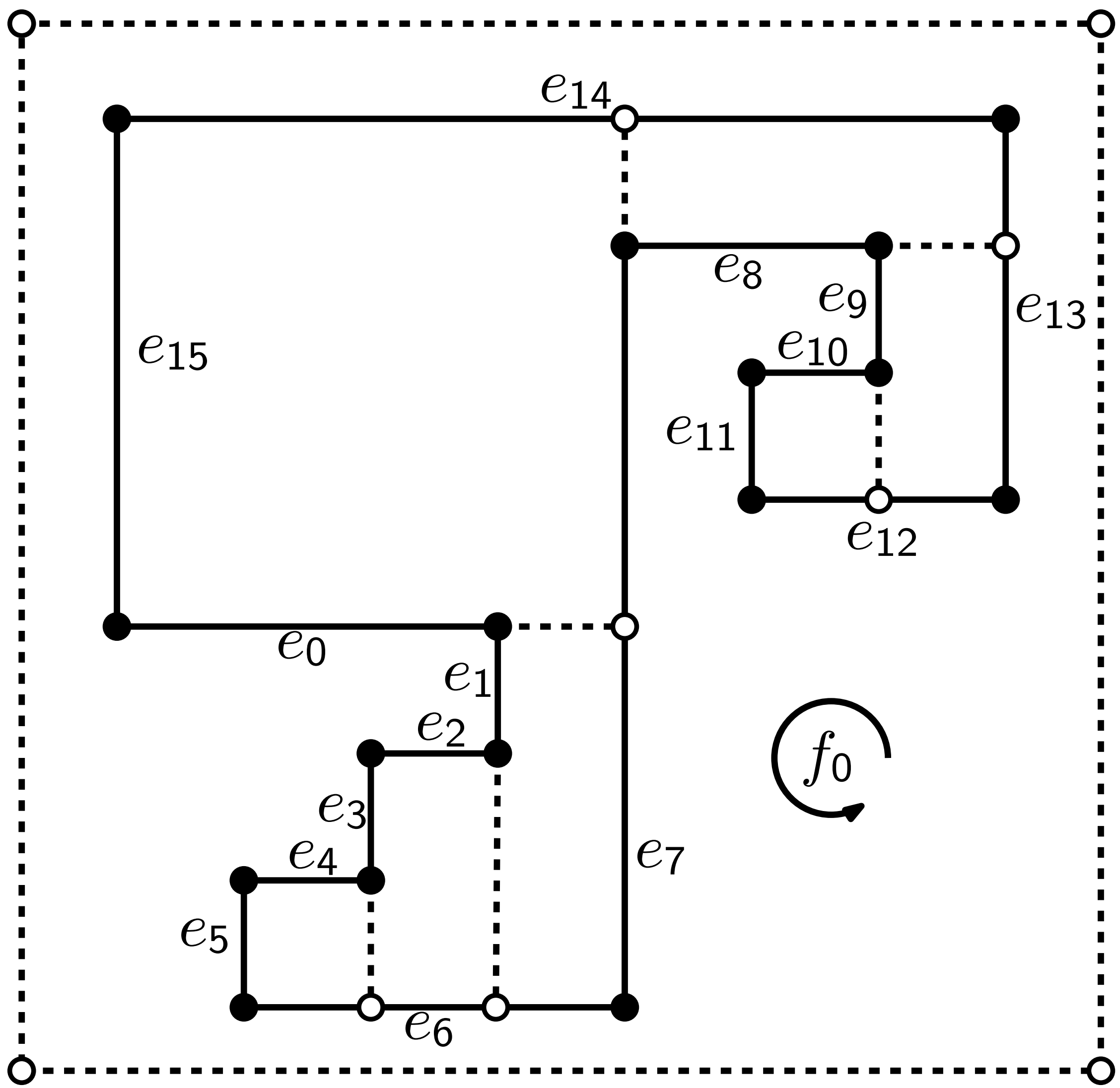
Verfeinerung von (G, H) – innere Facette



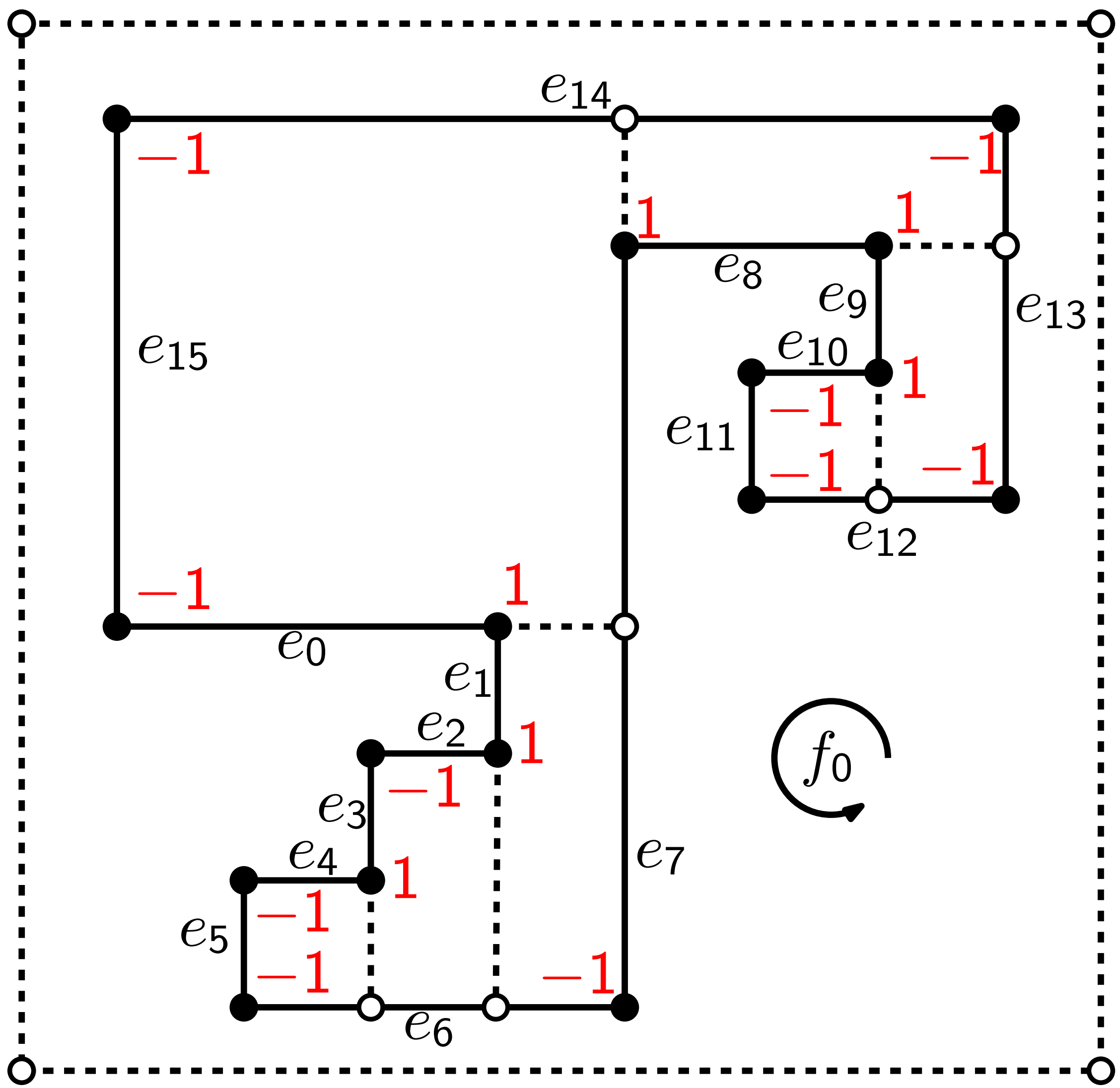
Verfeinerung von (G, H) – innere Facette



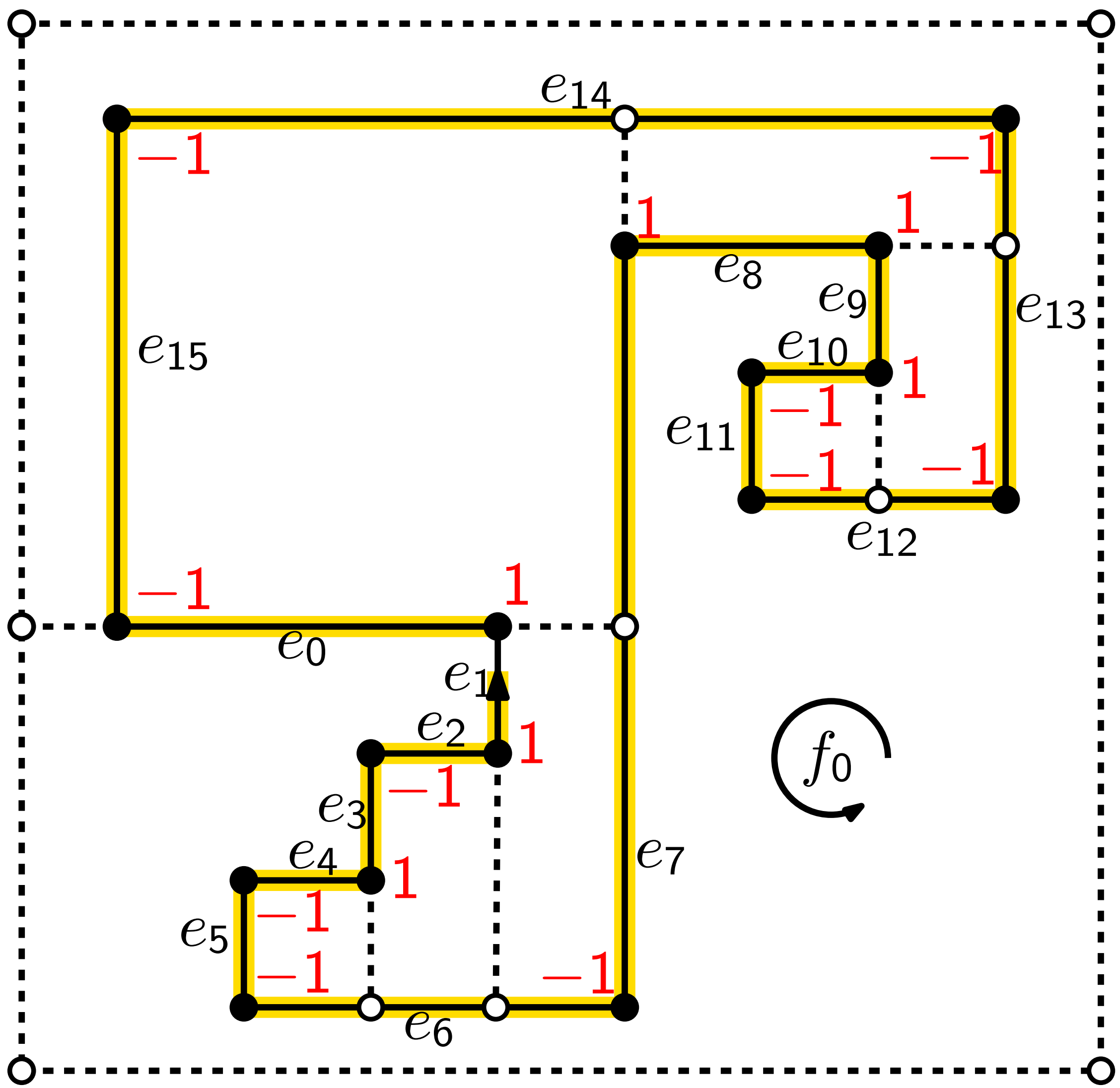
Verfeinerung von (G, H) – äußere Facette



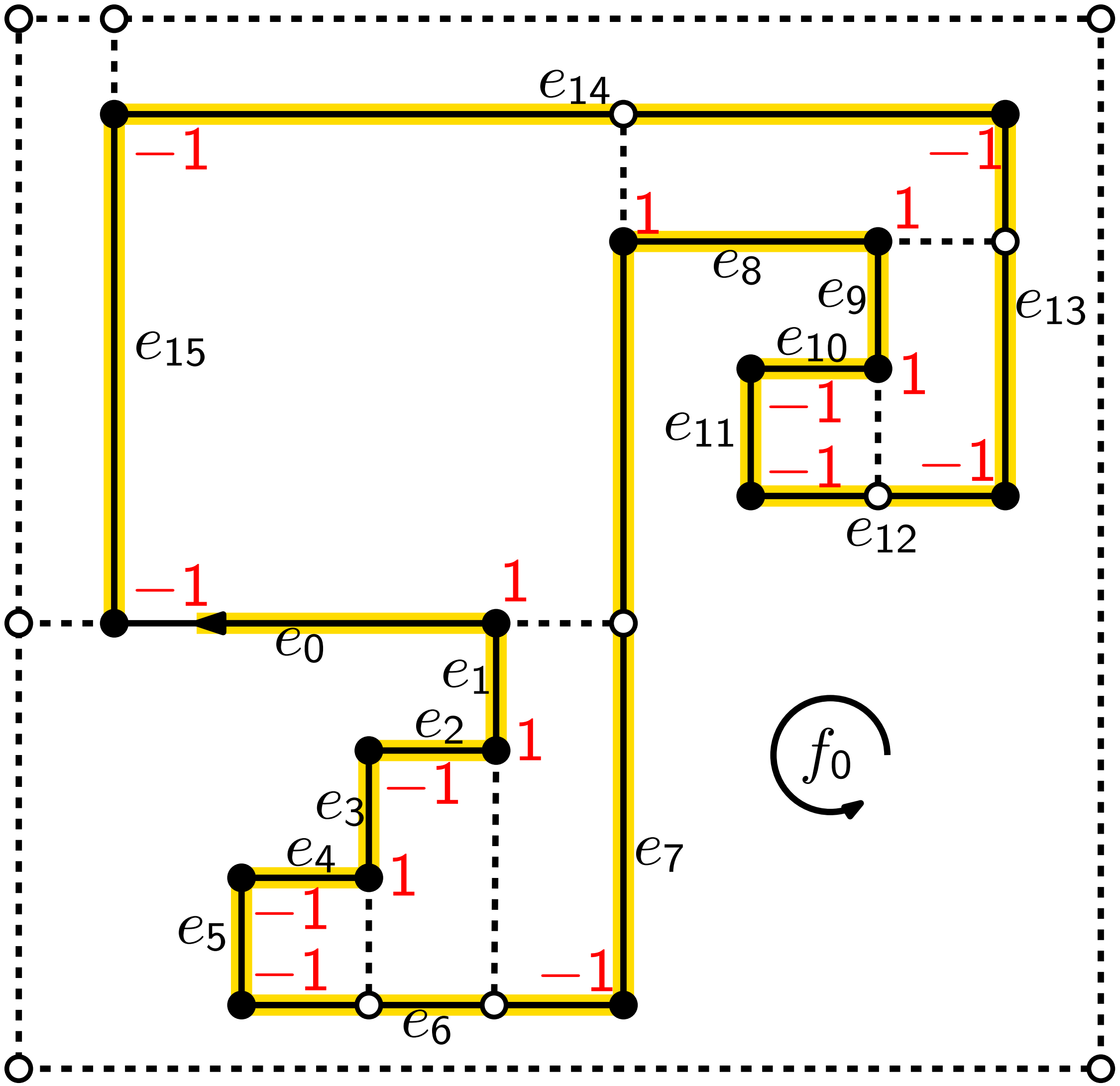
Verfeinerung von (G, H) – äußere Facette



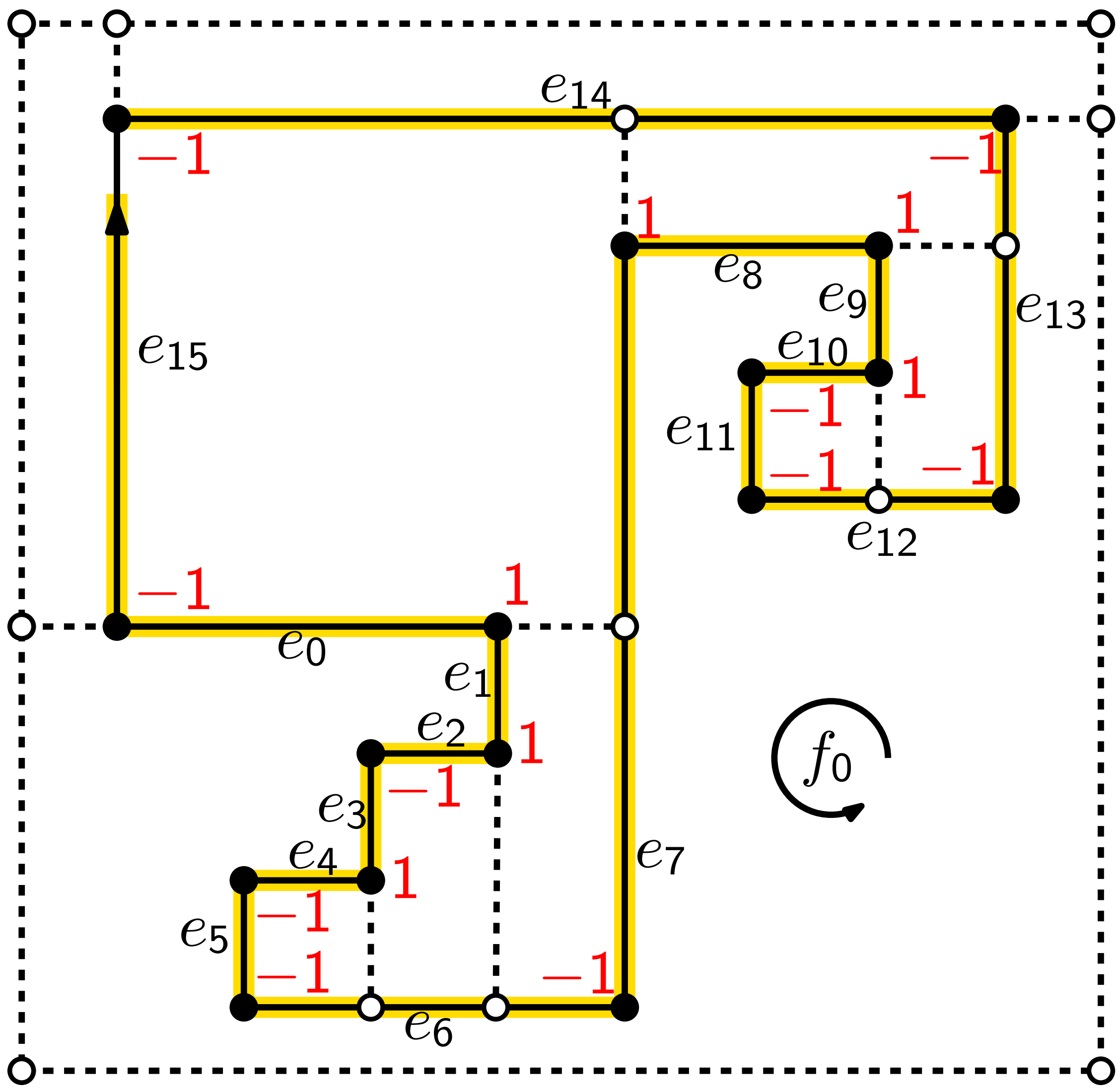
Verfeinerung von (G, H) – äußere Facette



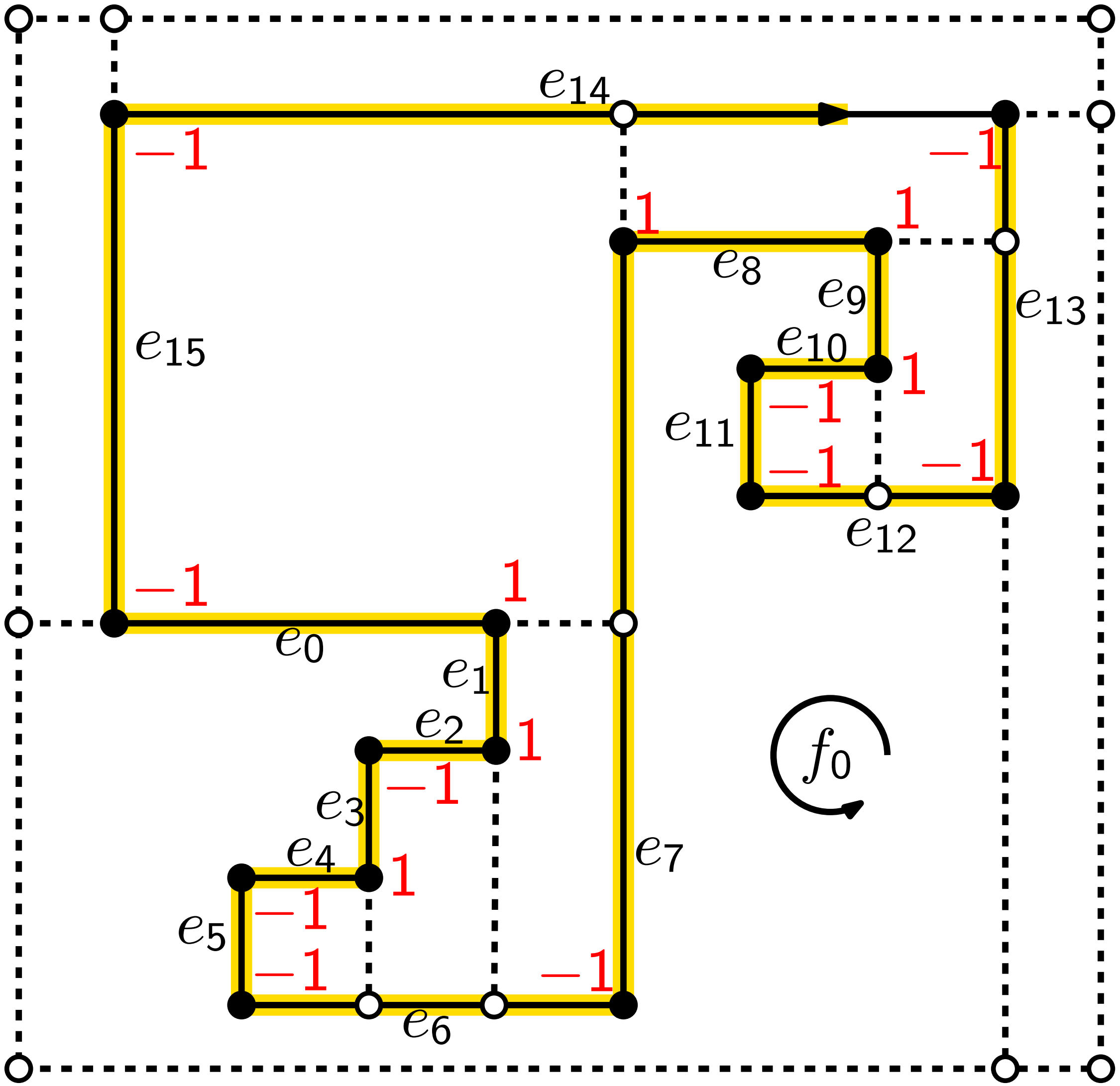
Verfeinerung von (G, H) – äußere Facette



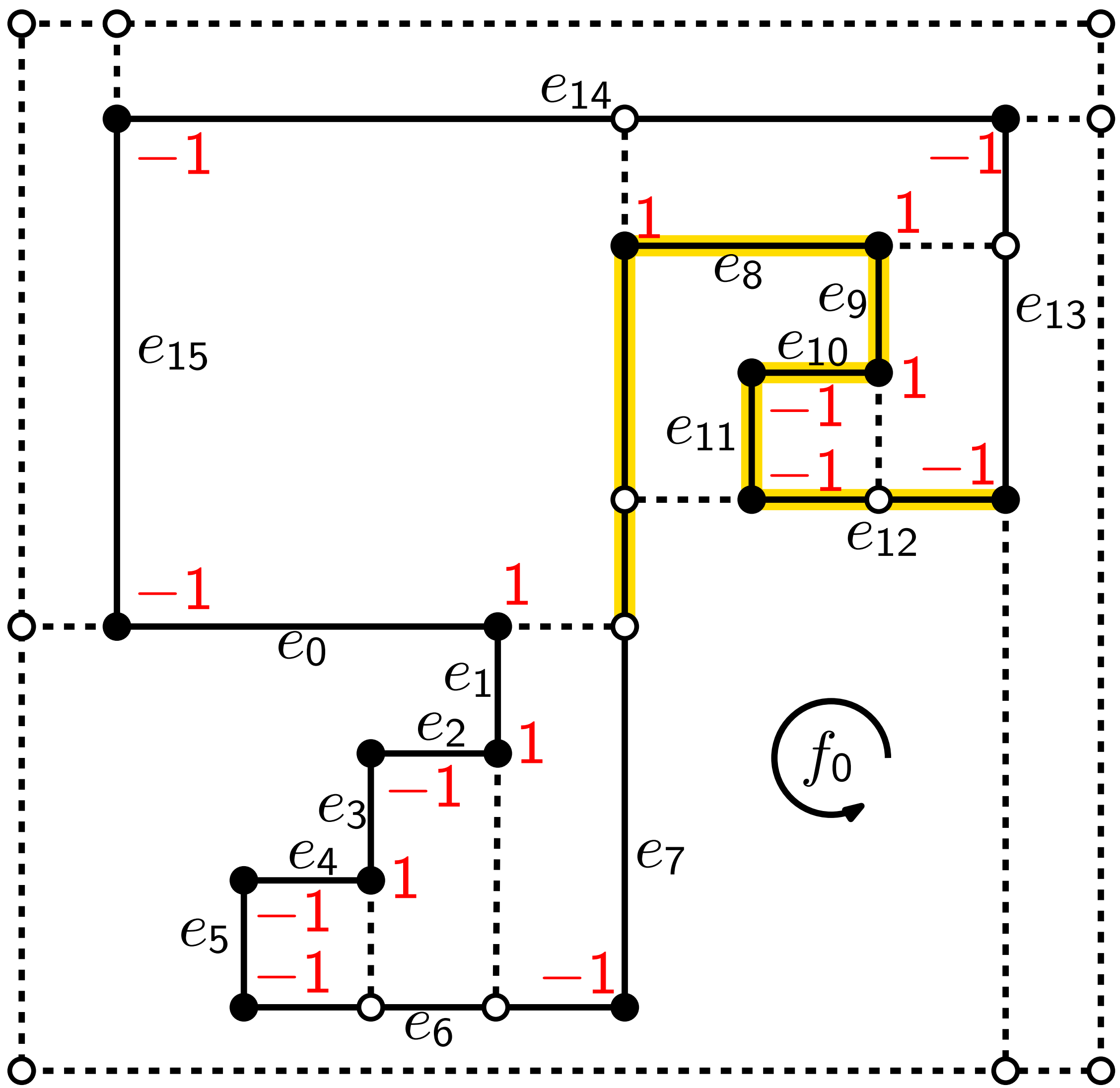
Verfeinerung von (G, H) – äußere Facette



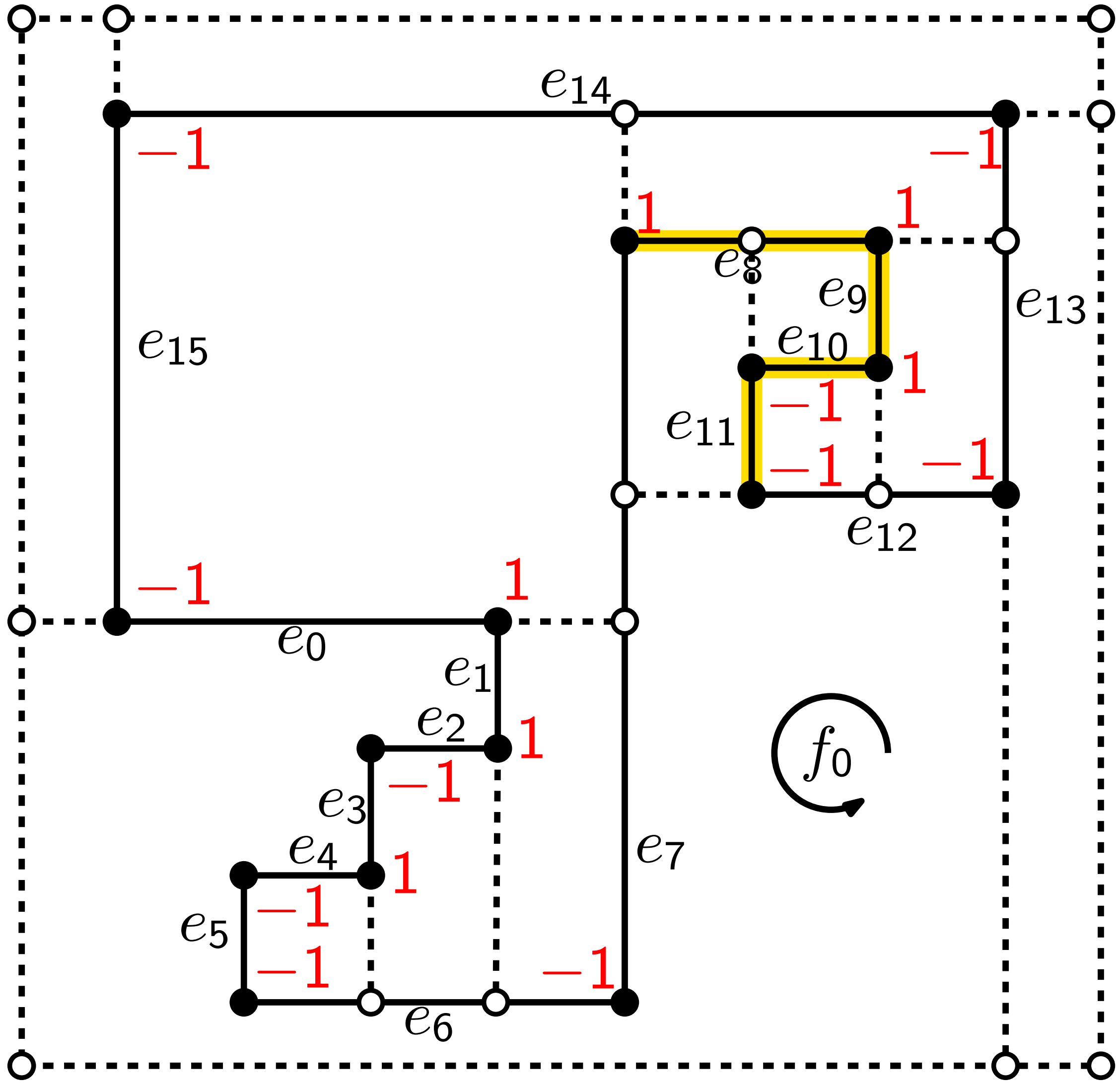
Verfeinerung von (G, H) – äußere Facette



Verfeinerung von (G, H) – äußere Facette



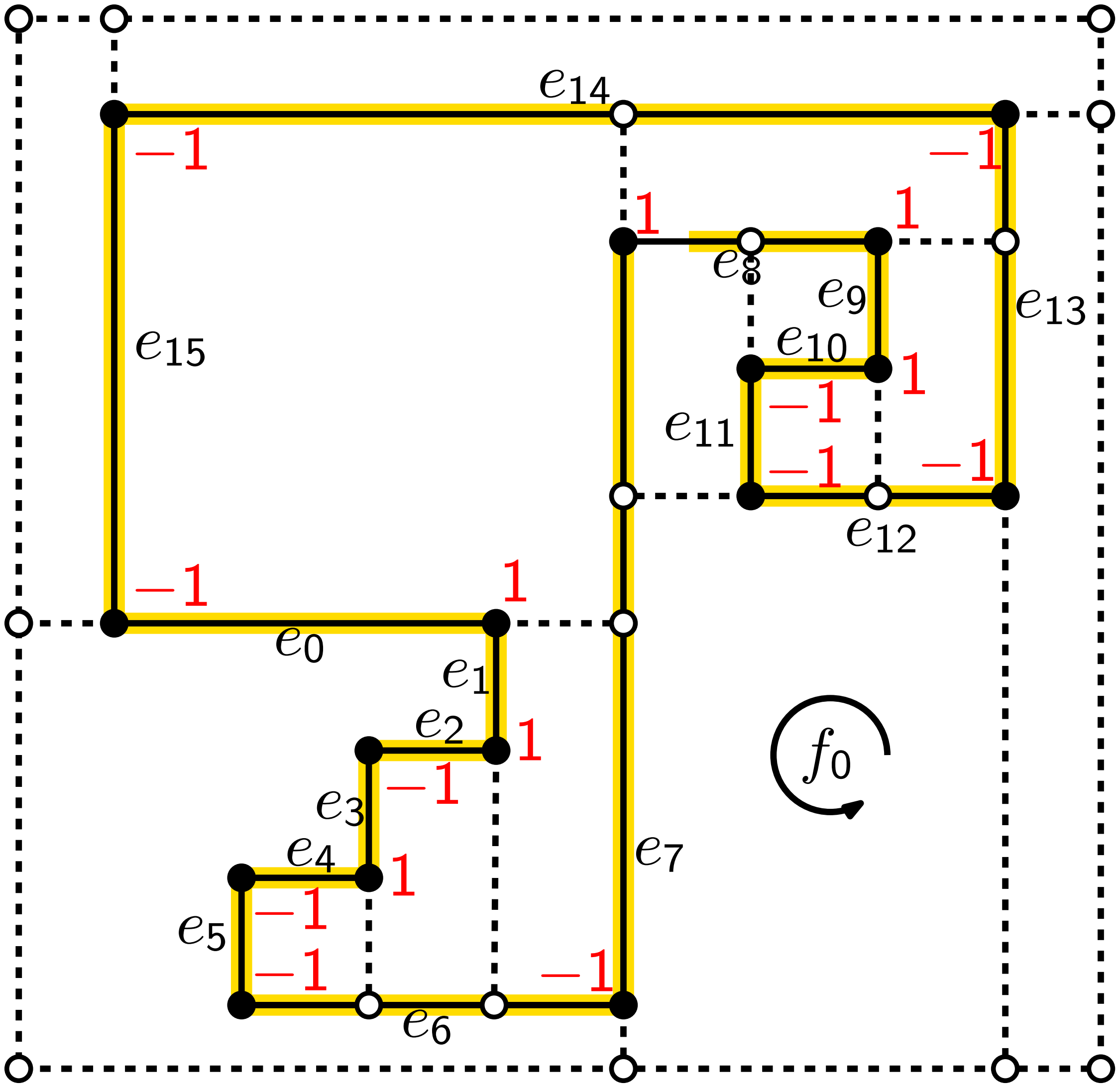
Verfeinerung von (G, H) – äußere Facette



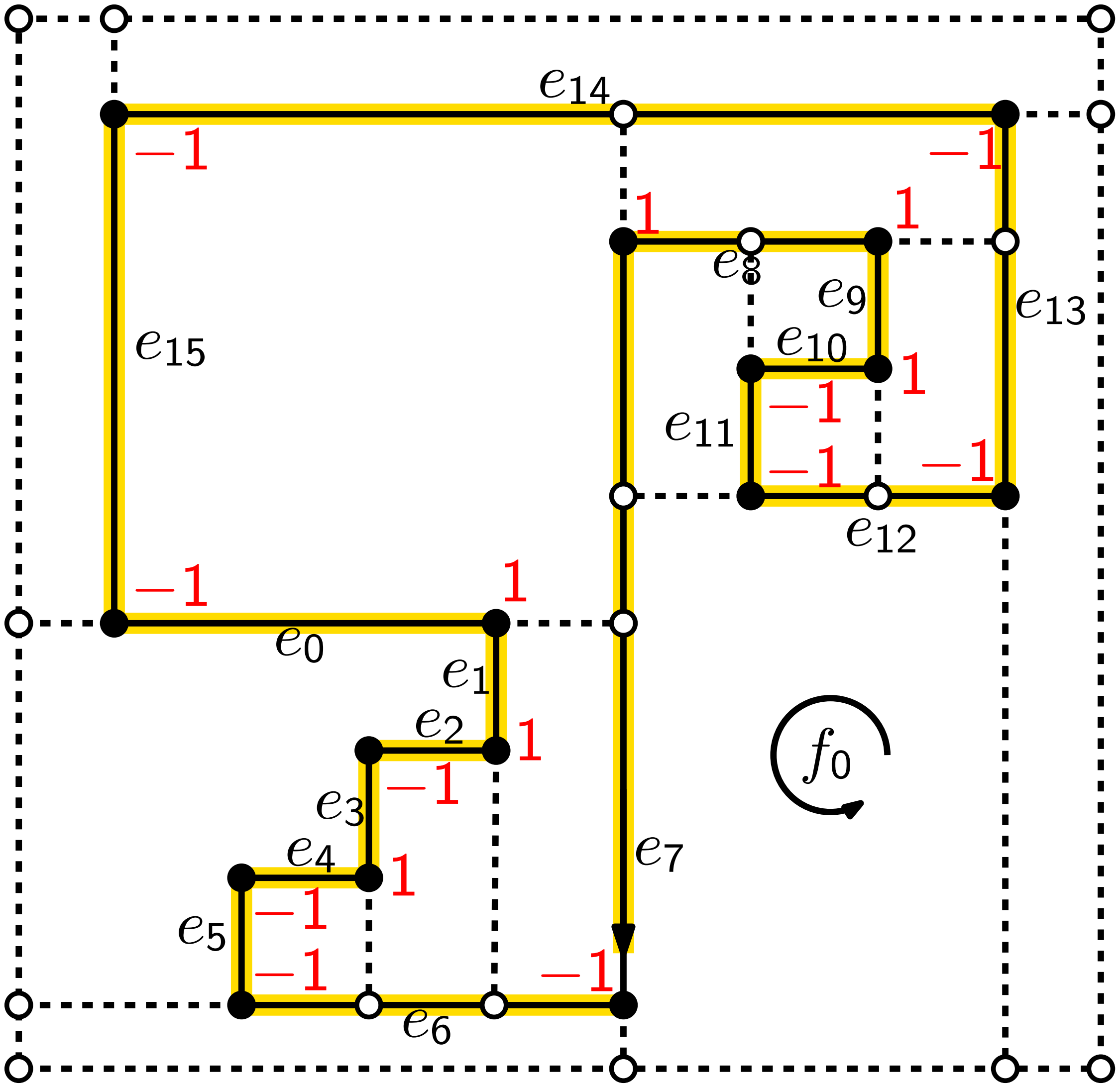
Institute for Theoretical Informatics

Algorithmics Group

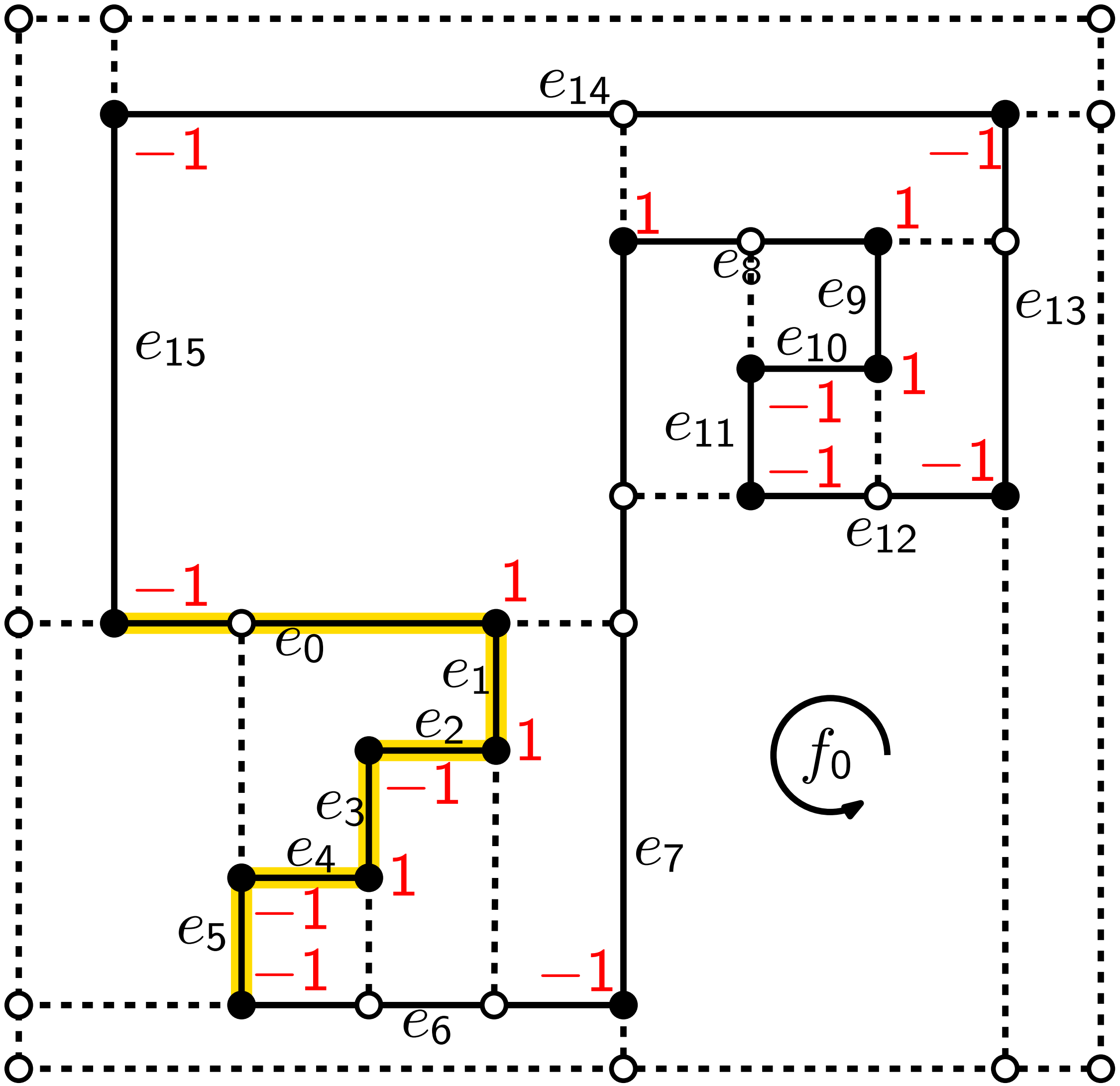
Verfeinerung von (G, H) – äußere Facette



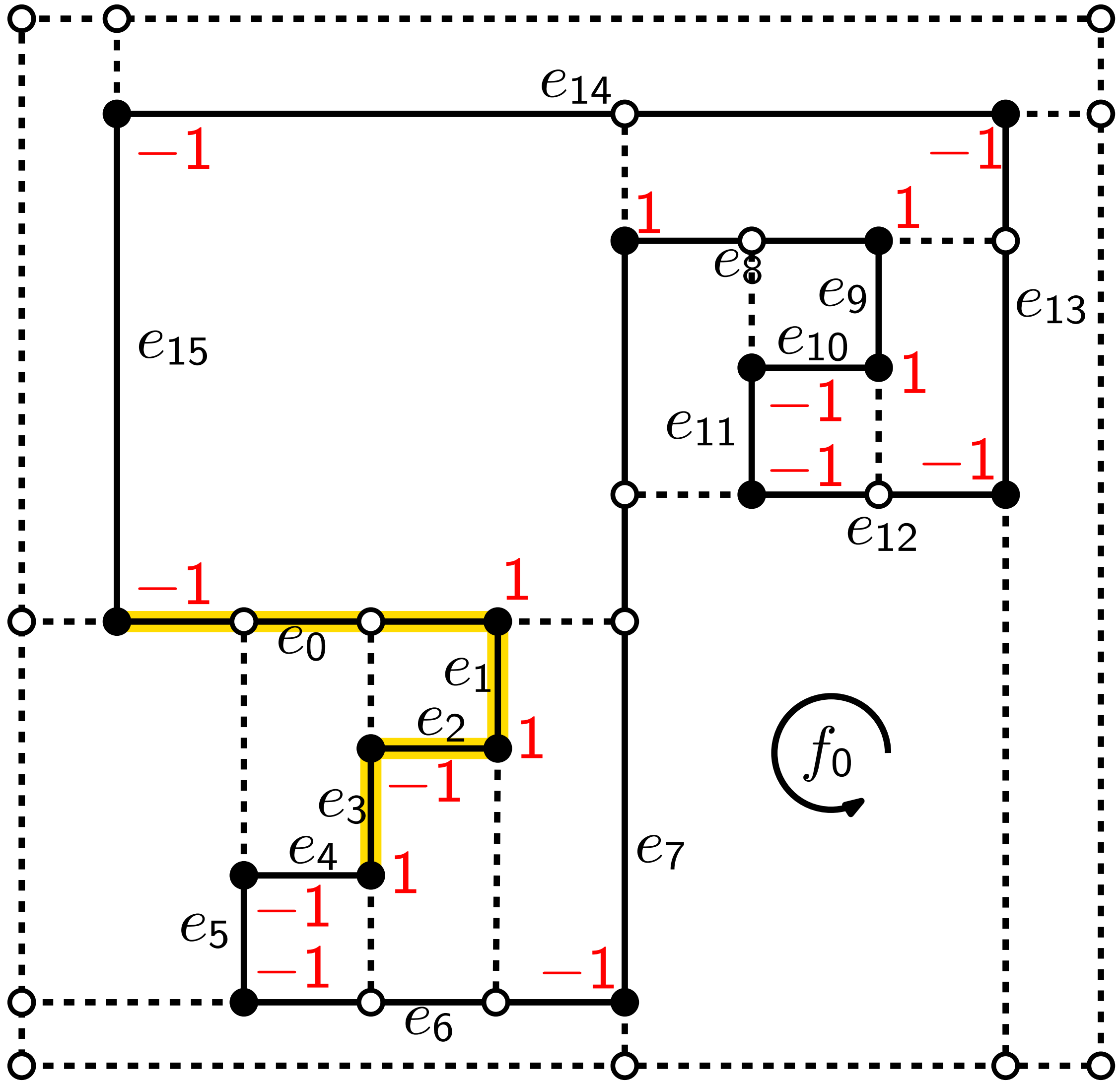
Verfeinerung von (G, H) – äußere Facette



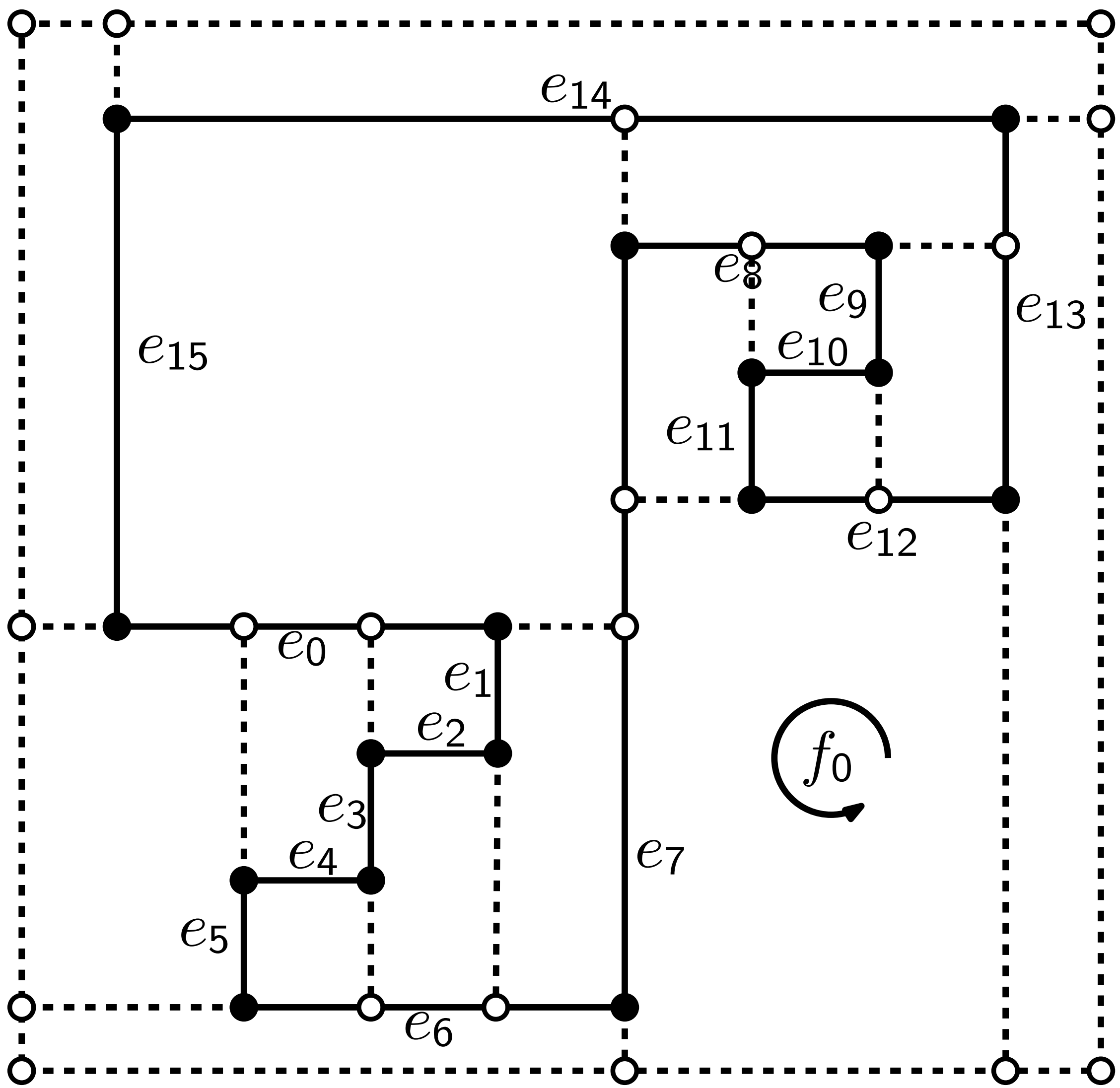
Verfeinerung von (G, H) – äußere Facette



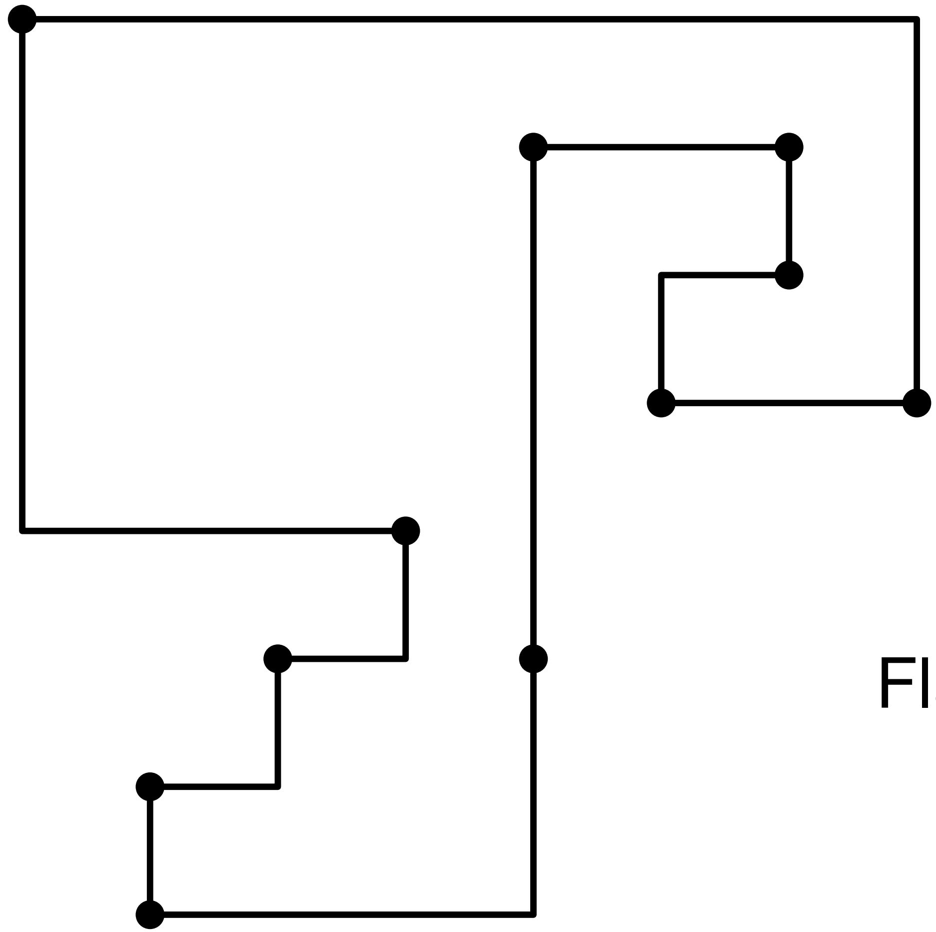
Verfeinerung von (G, H) – äußere Facette



Verfeinerung von (G, H) – äußere Facette



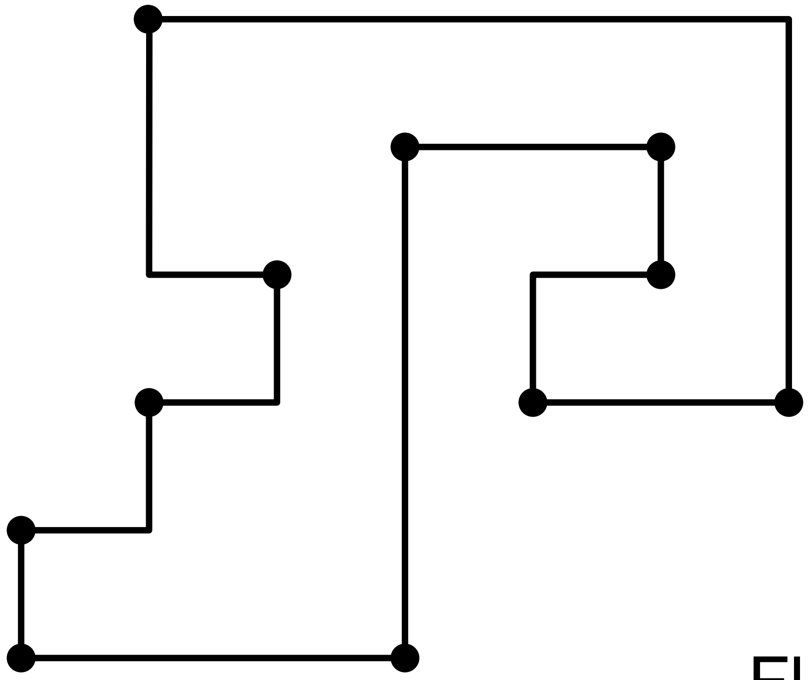
Verfeinerung von (G, H) – äußere Facette



Flächenminimal?

Institute for Theoretical Informatics
Algorithmics Group

Verfeinerung von (G, H) – äußere Facette



Flächenminimal?

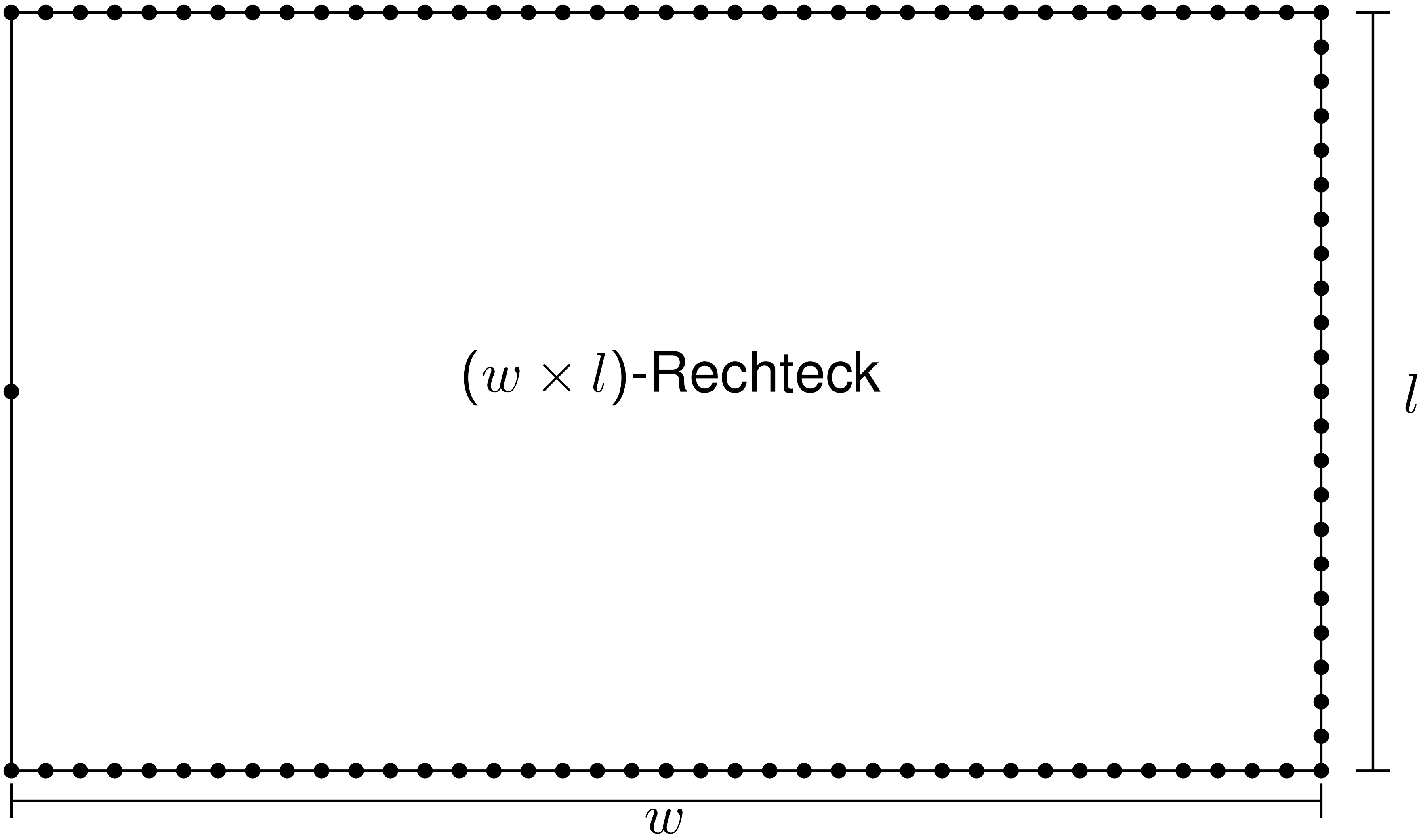
Nein!

Institute for Theoretical Informatics
Algorithmics Group

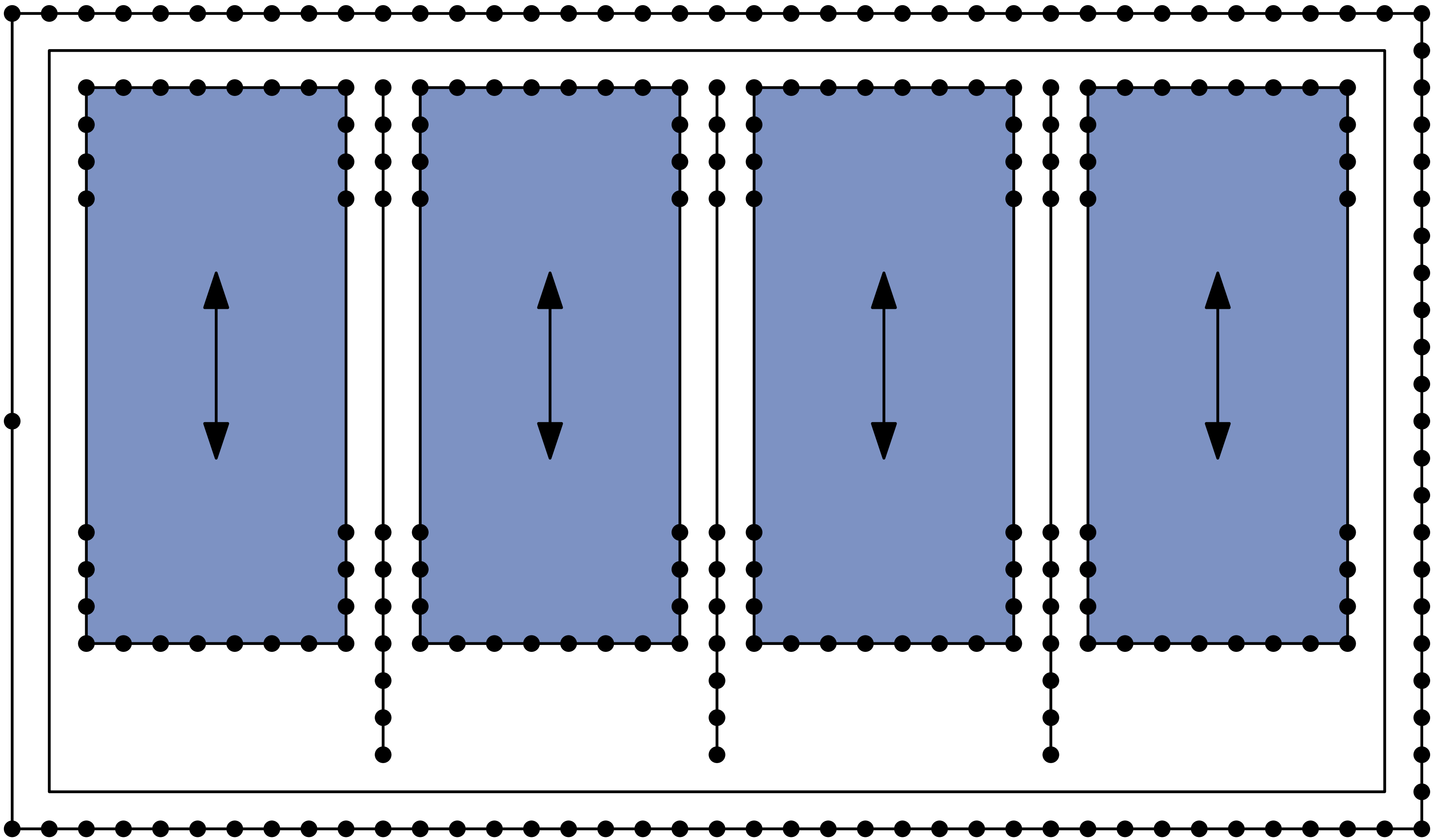
Kompaktierung ist NP-schwer [Patrignani '01]

- Grobstruktur von (G, H)
  - Begrenzung
  - Gürtel
  - Klauselgadgets
  - Variablengadgets
- Bestimme geeigneten Wert K
- (G, H) lässt sich in Fläche K zeichnen gdw. Φ erfüllbar

Begrenzung, Gürtel, Kolbengadget

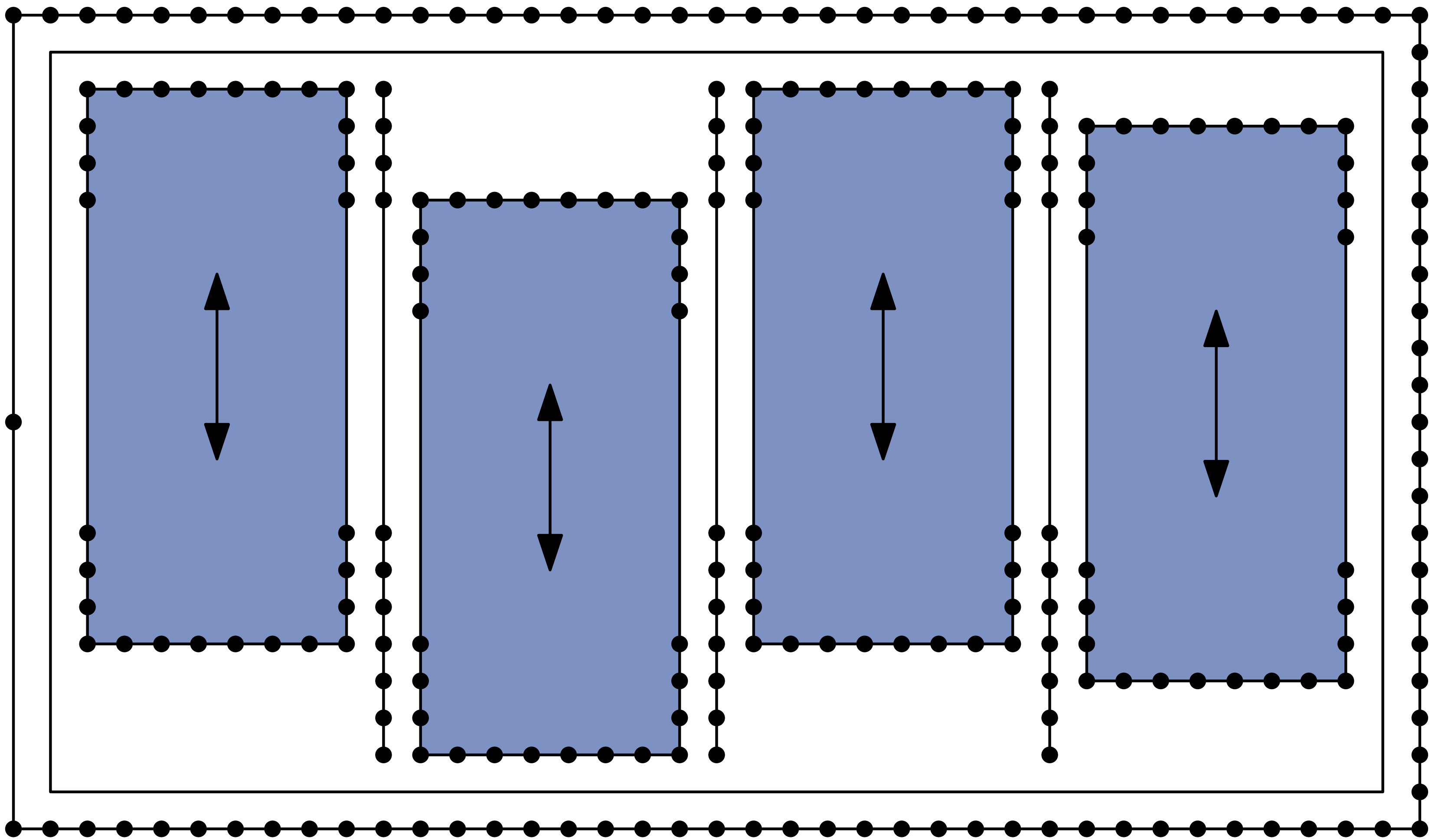


Begrenzung, Gürtel, Kolbengadget



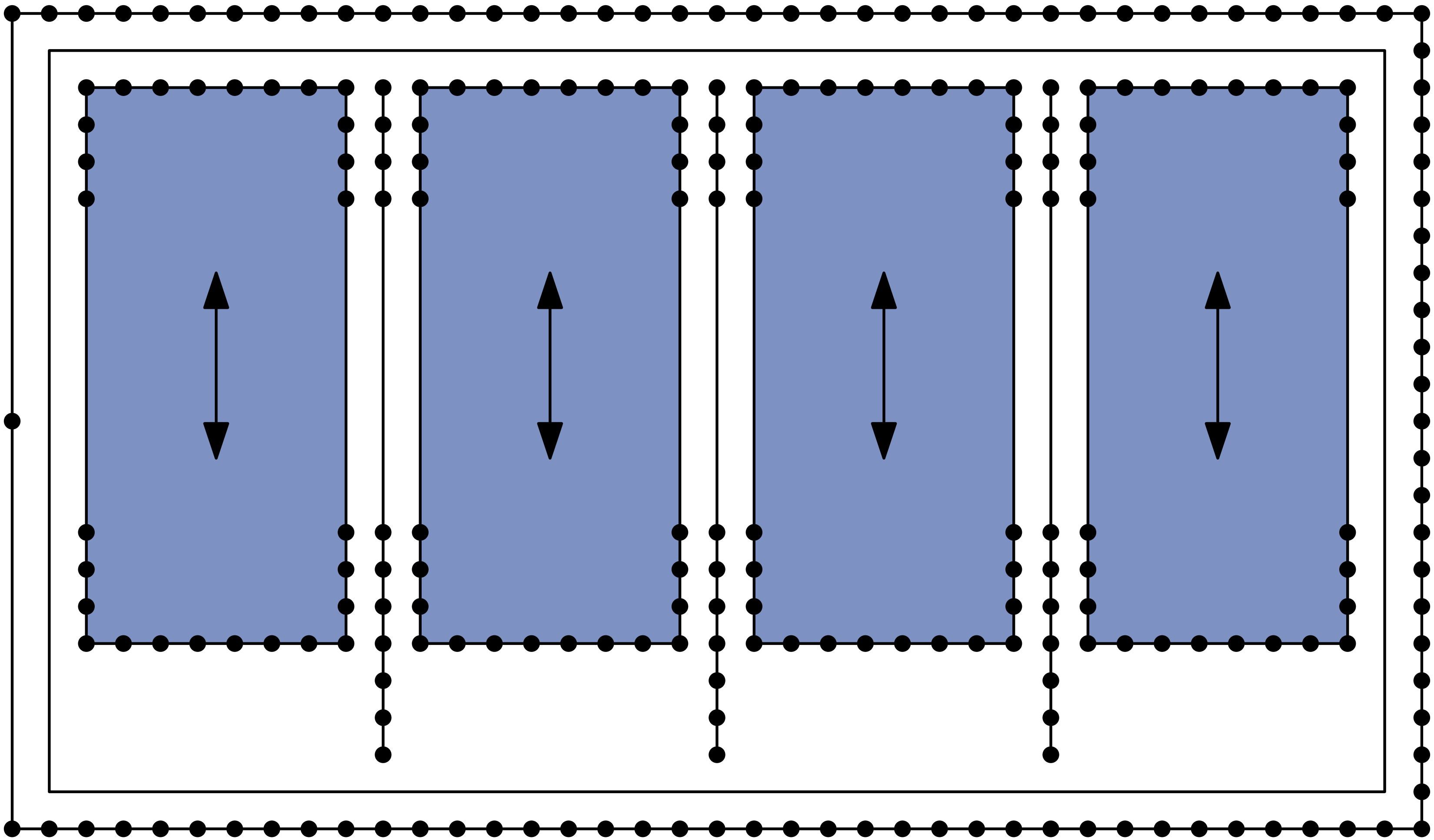
Institute for Theoretical Informatics
Algorithmics Group

Begrenzung, Gürtel, Kolbengadget



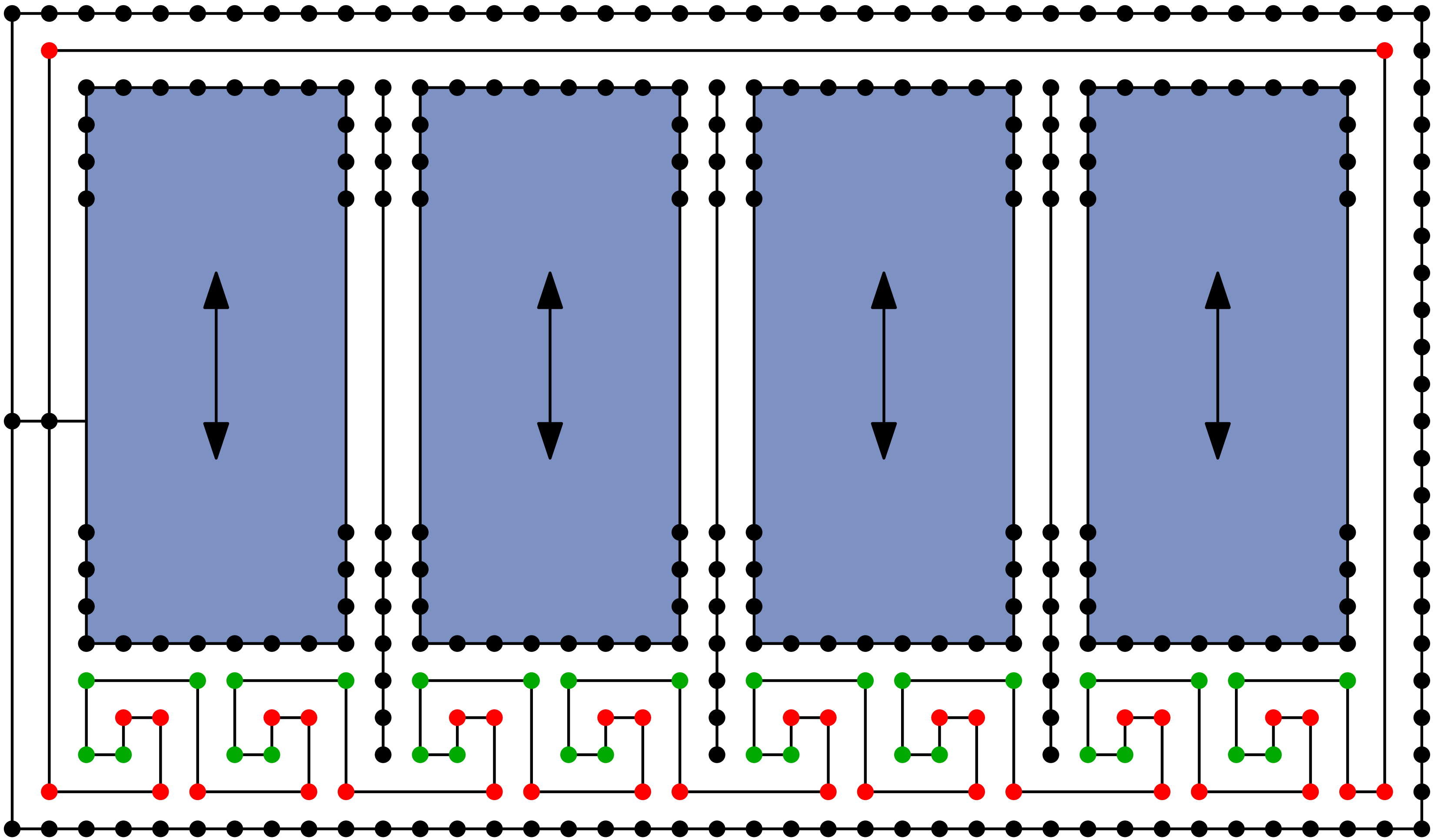
Institute for Theoretical Informatics
Algorithmics Group

Begrenzung, Gürtel, Kolbengadget



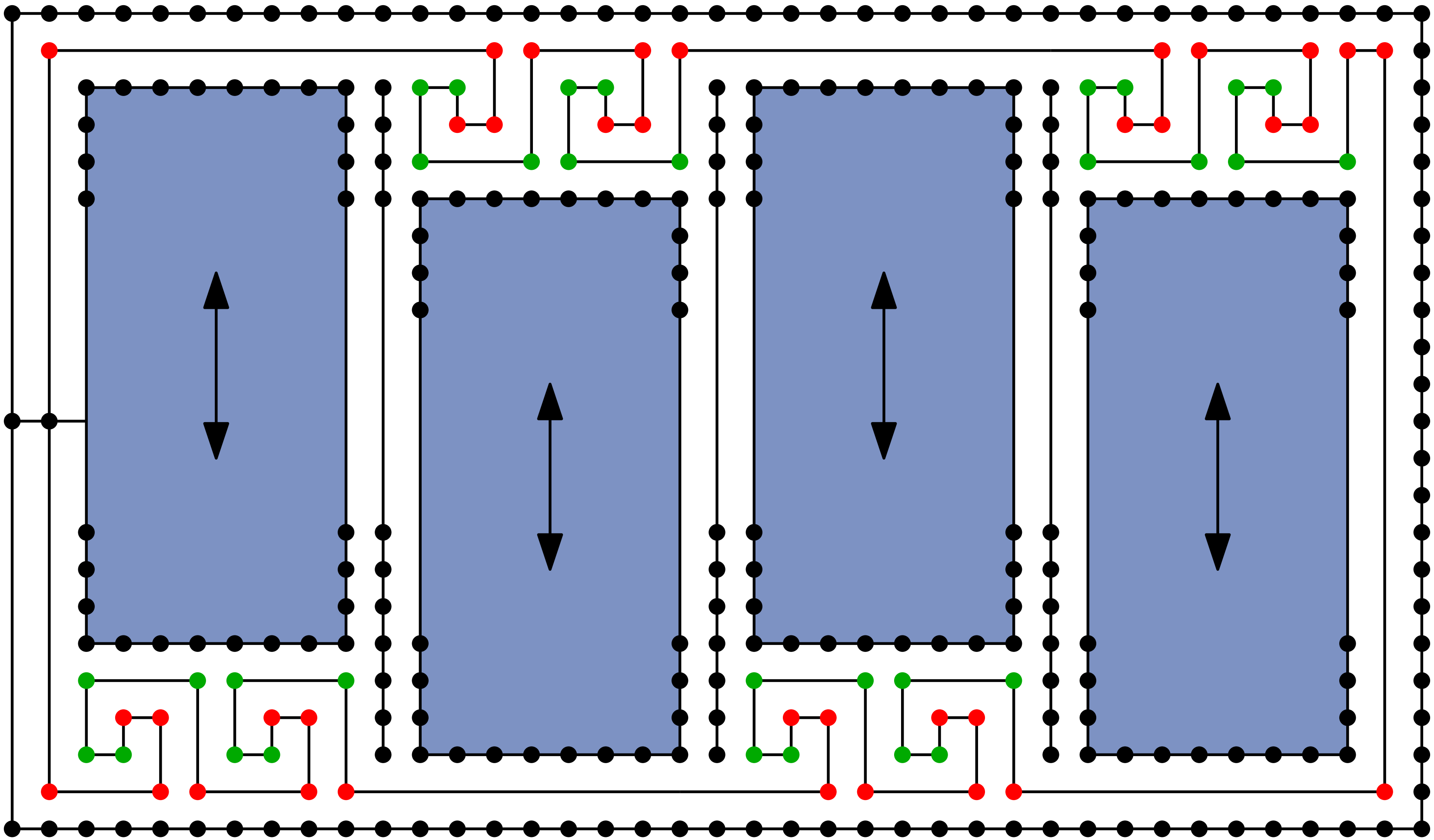
Institute for Theoretical Informatics
Algorithmics Group

Begrenzung, Gürtel, Kolbengadget



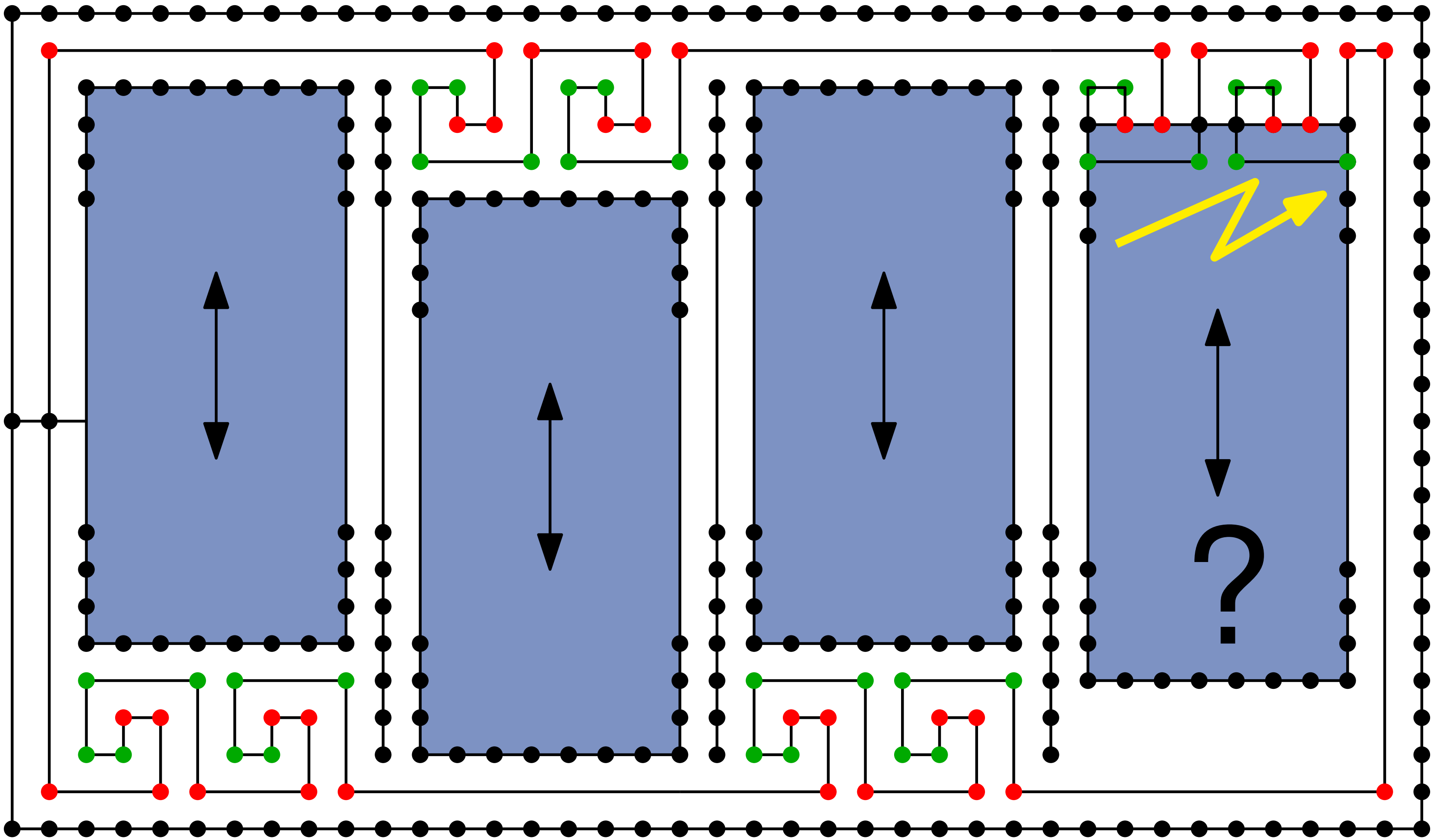
Institute for Theoretical Informatics
Algorithmics Group

Begrenzung, Gürtel, Kolbengadget



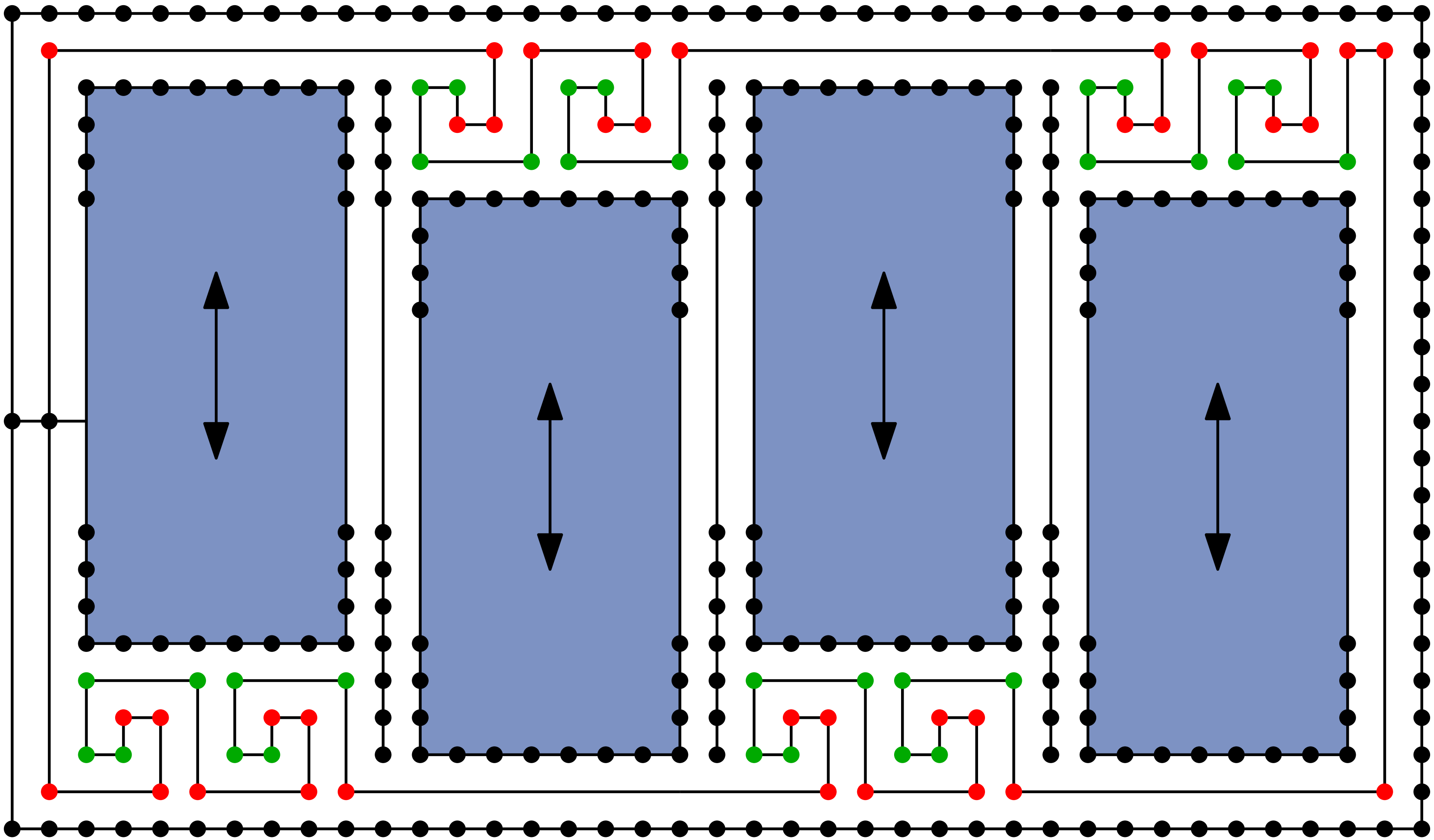
Institute for Theoretical Informatics
Algorithmics Group

Begrenzung, Gürtel, Kolbengadget



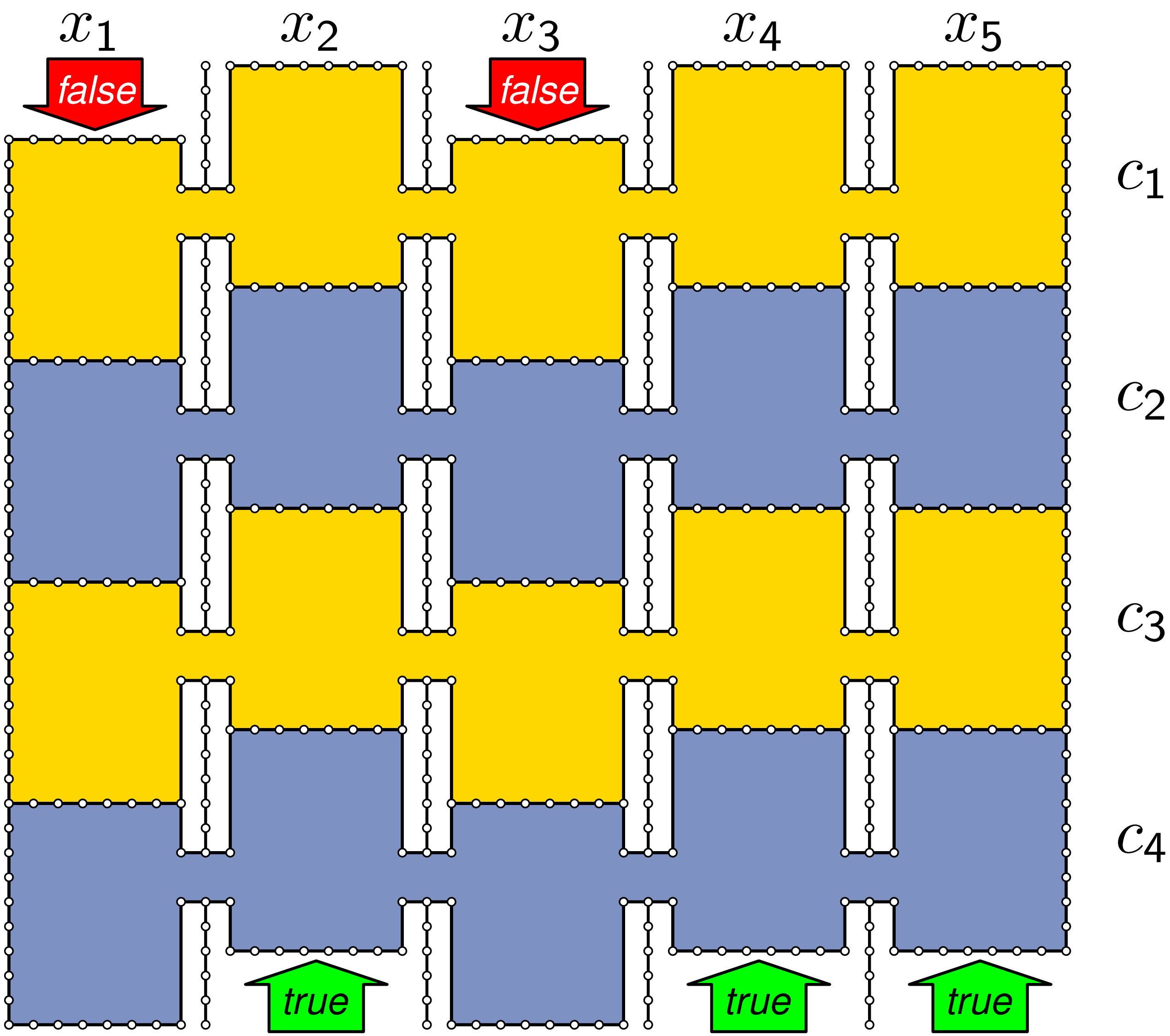
Institute for Theoretical Informatics
Algorithmics Group

Begrenzung, Gürtel, Kolbengadget



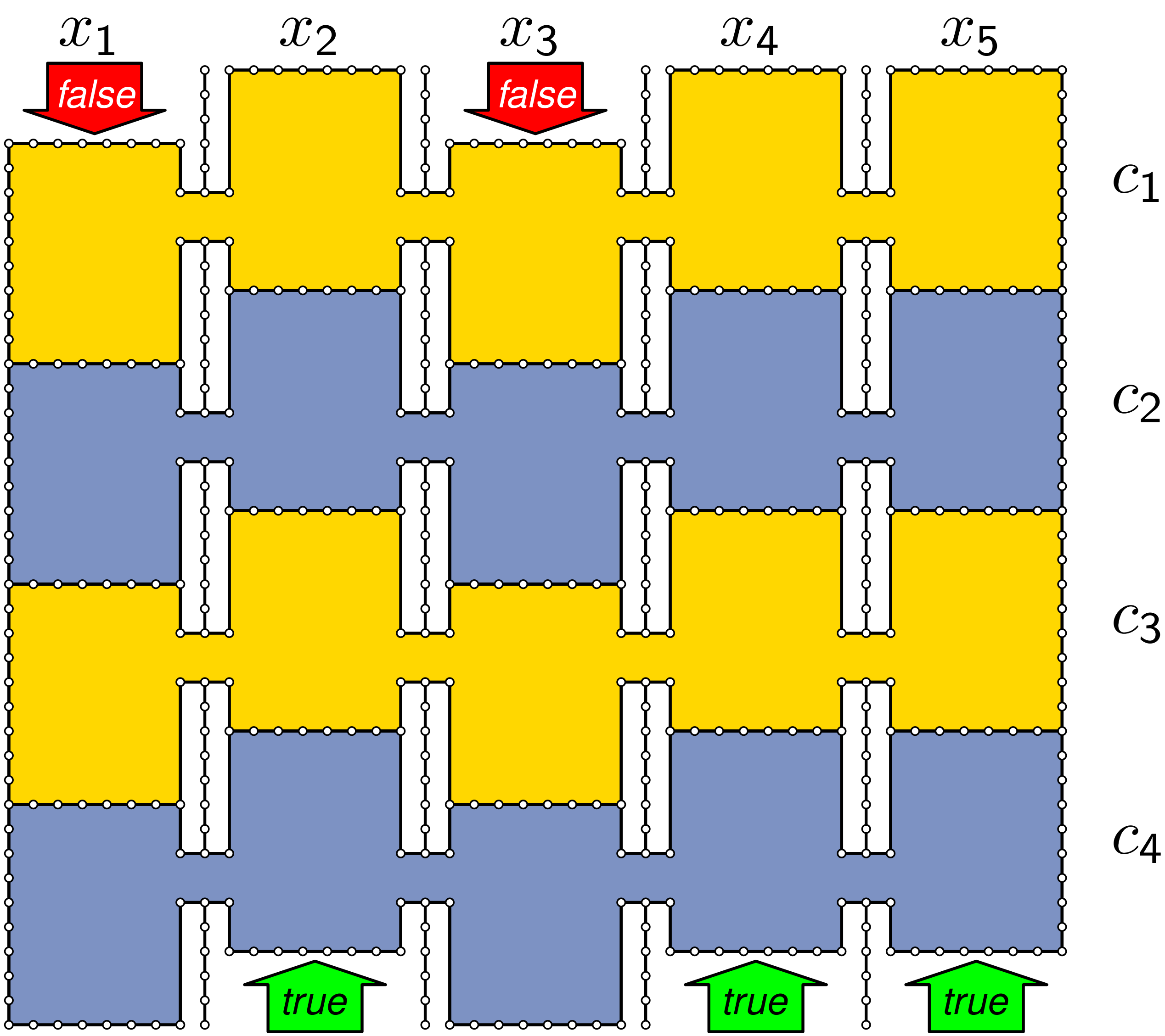
Institute for Theoretical Informatics
Algorithmics Group

Klauselgadgets



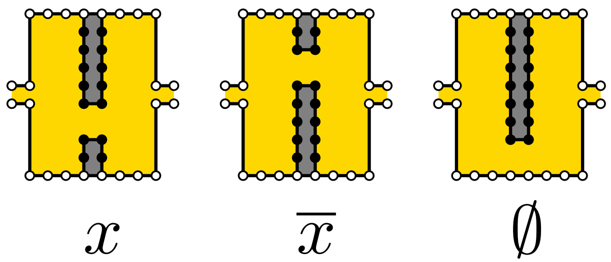
Institute for Theoretical Informatics
Algorithmics Group

Klauselgadgets

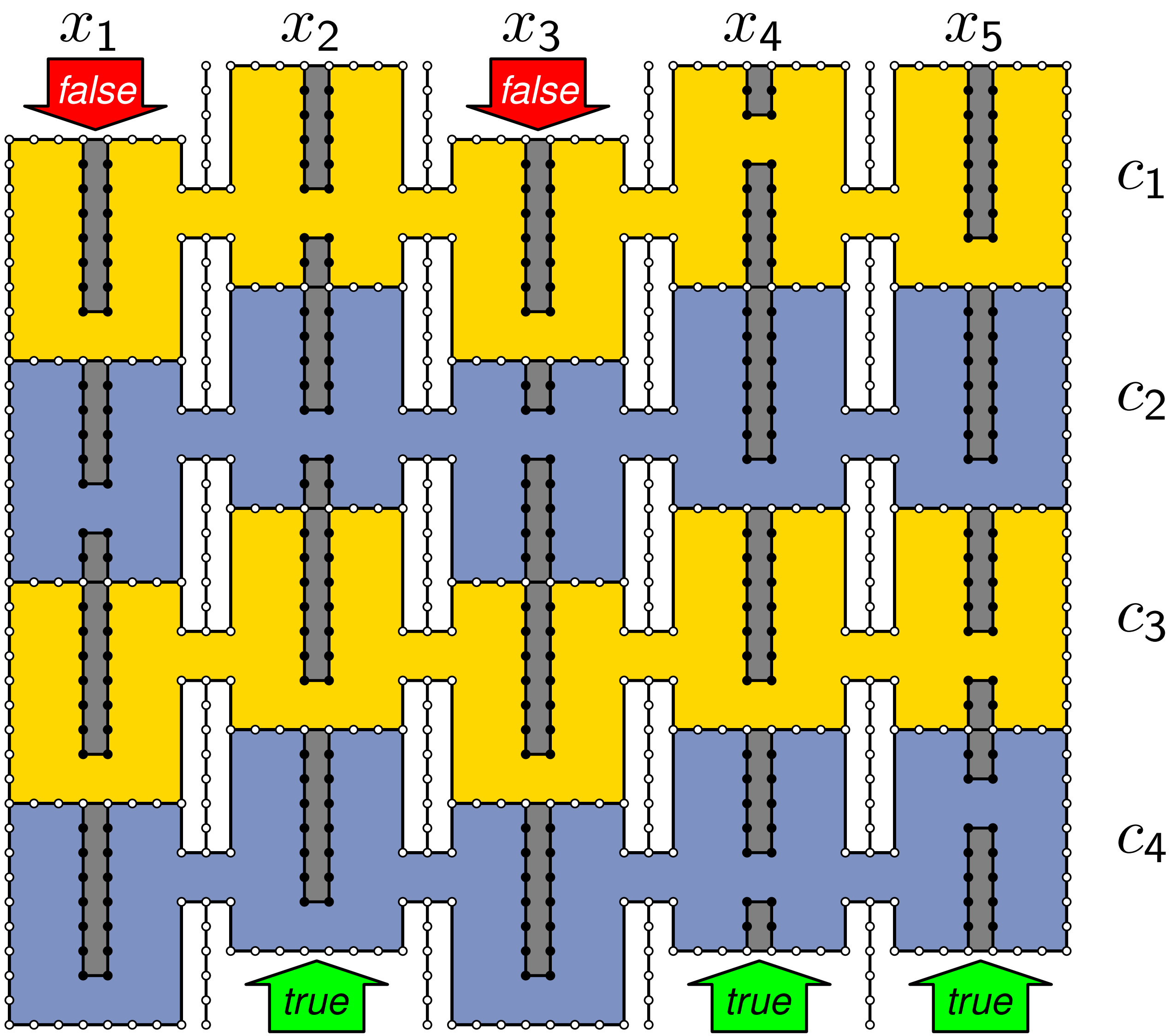


Beispiel:

$$c_1 = x_2 \vee \overline{x_4}$$
$$c_2 = x_1 \vee x_2 \vee \overline{x_3}$$
$$c_3 = x_5$$
$$c_4 = x_4 \vee \overline{x_5}$$



Klauselgadgets



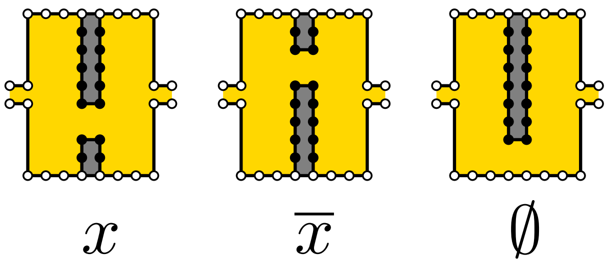
Beispiel:

$c_1 = x_2 \vee \overline{x_4}$

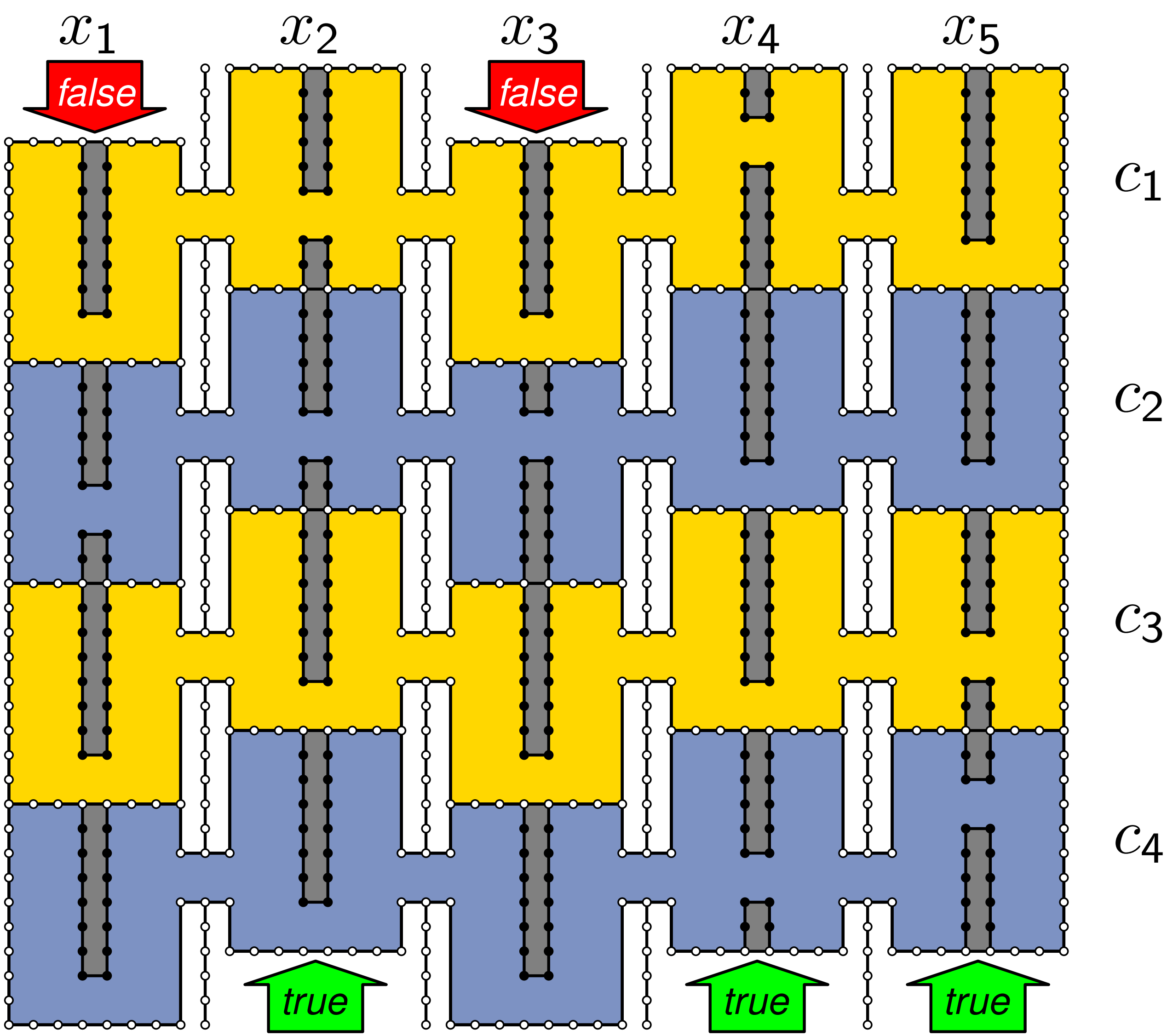
$c_2 = x_1 \vee x_2 \vee \overline{x_3}$

$c_3 = x_5$

$c_4 = x_4 \vee \overline{x_5}$

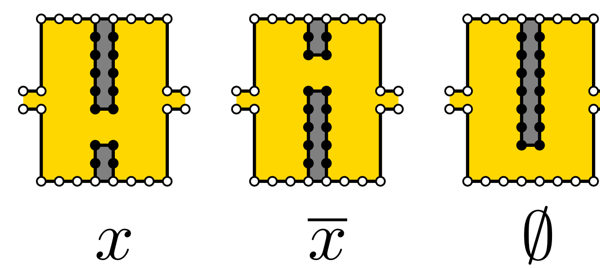


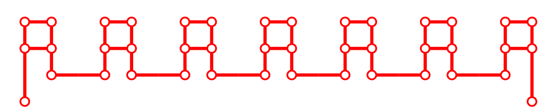
Klauselgadgets



Beispiel:

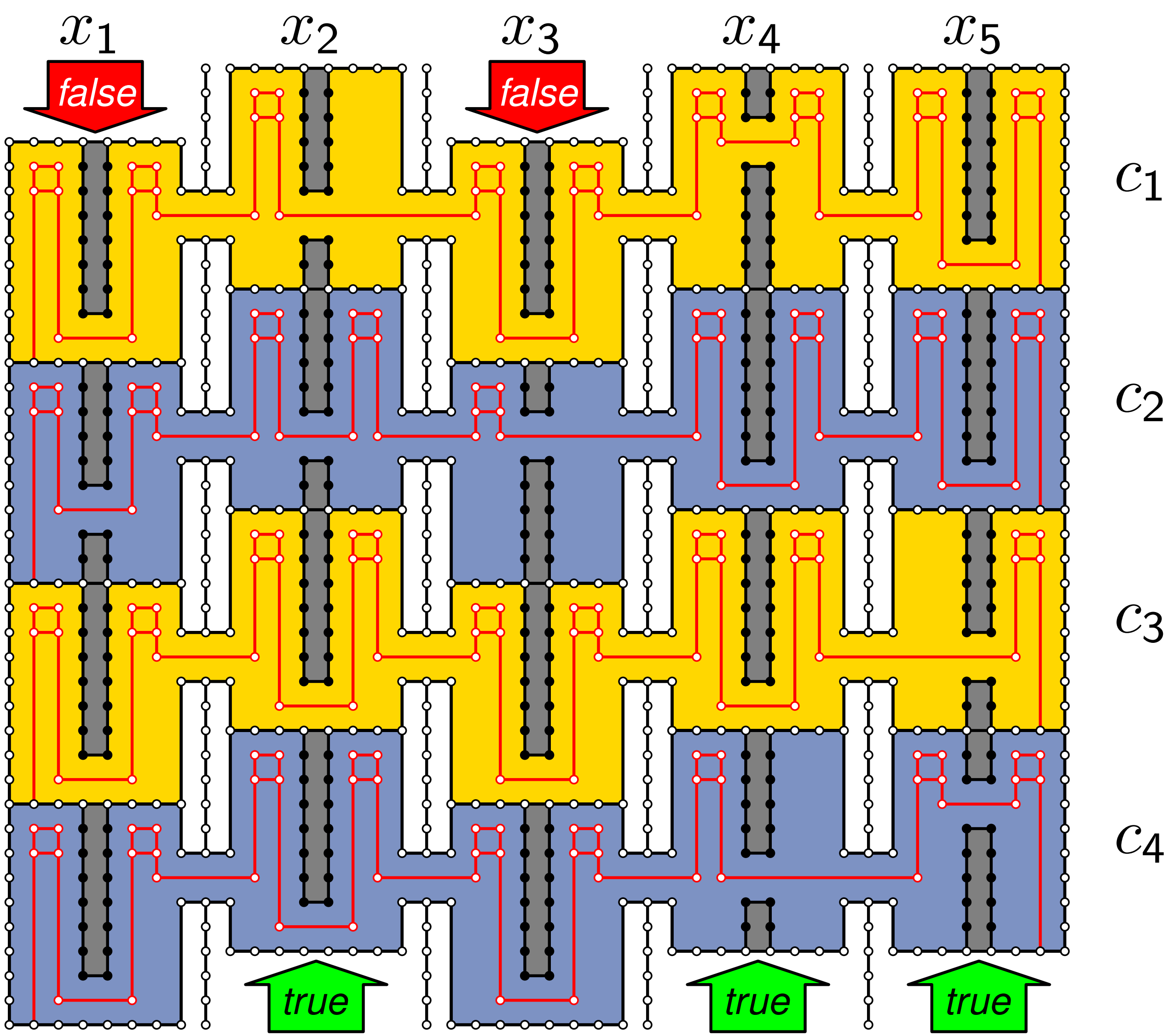
$$c_1 = x_2 \vee \overline{x_4}$$
$$c_2 = x_1 \vee x_2 \vee \overline{x_3}$$
$$c_3 = x_5$$
$$c_4 = x_4 \vee \overline{x_5}$$





lege $(2n - 1)$-A-Kette durch jede Klausel

Klauselgadgets



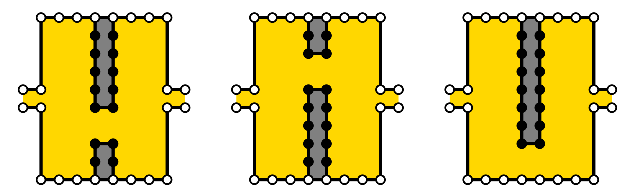
Beispiel:

$c_1 = x_2 \vee \overline{x_4}$

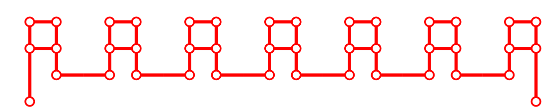
$c_2 = x_1 \vee x_2 \vee \overline{x_3}$

$c_3 = x_5$

$c_4 = x_4 \vee \overline{x_5}$



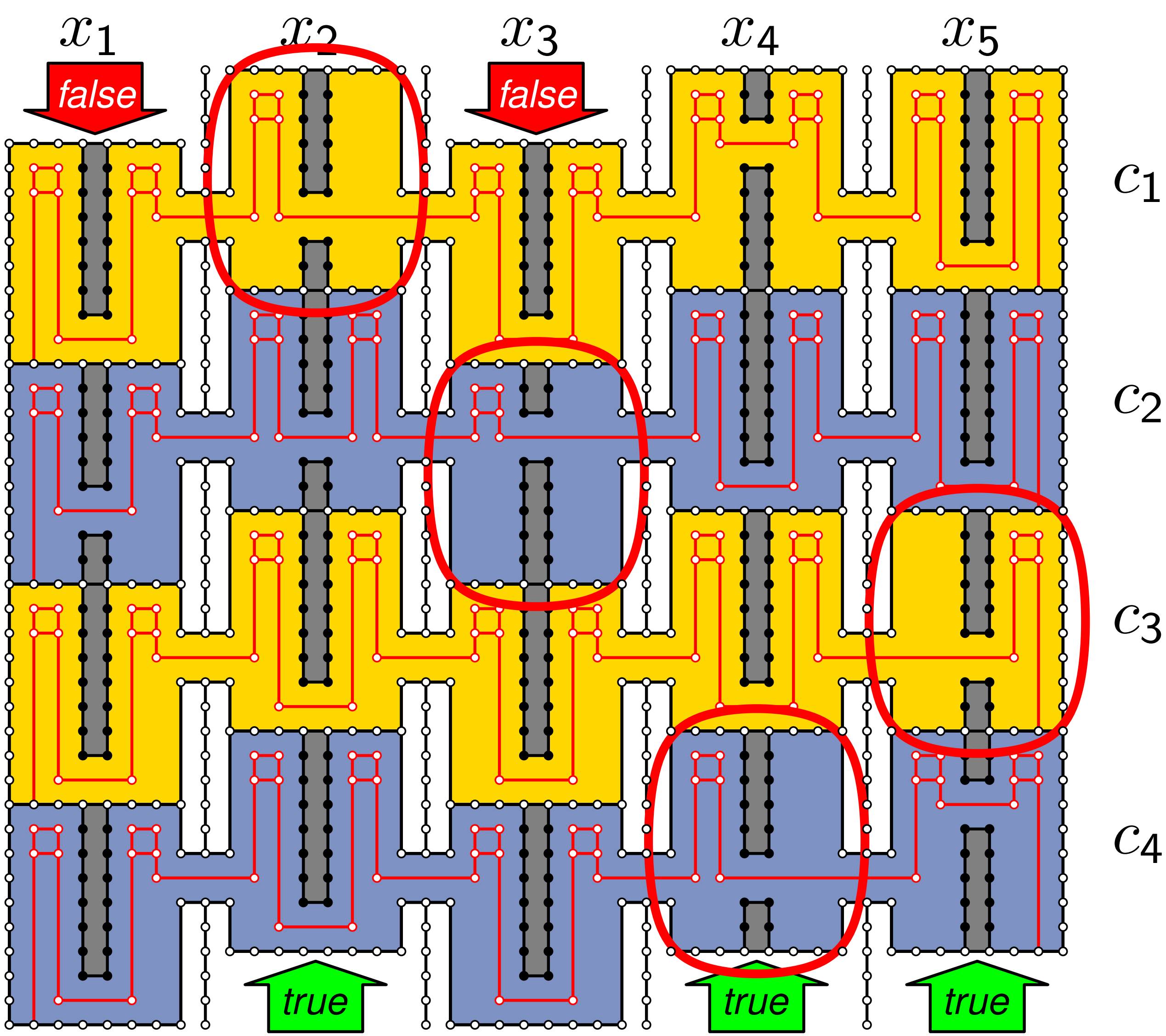
x $\quad$ $\overline{x}$ $\quad$ $\emptyset$



lege $(2n - 1)$-A-Kette durch jede Klausel

Institute for Theoretical Informatics
Algorithmics Group

Klauselgadgets



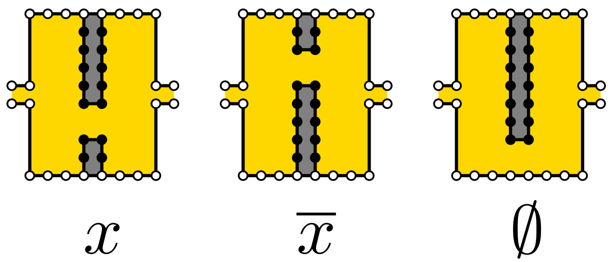
Beispiel:

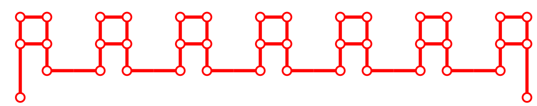
$c_1 = x_2 \vee \overline{x_4}$

$c_2 = x_1 \vee x_2 \vee \overline{x_3}$

$c_3 = x_5$

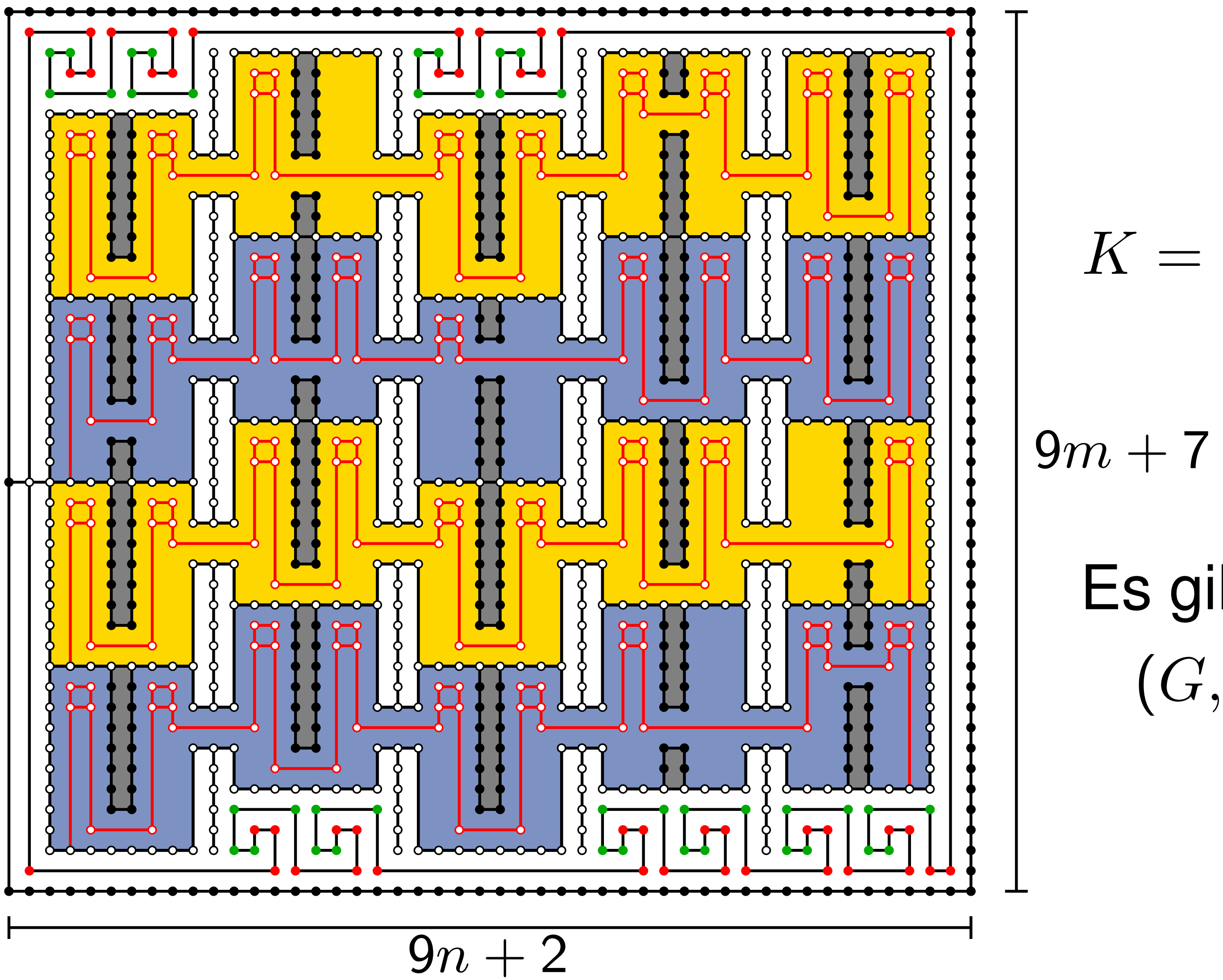
$c_4 = x_4 \vee \overline{x_5}$





lege $(2n - 1)$-A-Kette durch jede Klausel

Komplette Reduktion



Setze

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

Es gilt:

(G, H) auf Fläche K zeichenbar

$\Leftrightarrow$

Φ erfüllbar