Netzwerkalgorithmen

Sommersemester 2022

Stefan Näher
Universität Trier
naeher@uni-trier.de

Vorlesung 11

Das Min-Cost-Flow Problem (MFC) (Eingabe)

1. Ein **gerichteter Graph** $G = (V, E)$
2. **Kapazitätsschranken** $u : E \longrightarrow \mathbb{R}_0^+$
   u_{ij} ist obere Schranke für den maximalen Flusswert x_{ij} über die Kante (i, j).
3. Eine **Kosten**-Funktion $c : E \longrightarrow \mathbb{R}$
   c_{ij} sind die Kosten für den Transport **einer Einheit** Fluss über die Kante (i, j).
4. Eine **Supply**-Funktion $b : V \longrightarrow \mathbb{R}$
   $b(i)$ ist der Supply (oder die Einspeisung) am Knoten i

   Einspeise-Knoten, wenn $b(i) > 0$

   Entnahme-Knoten, wenn $b(i) < 0$

   Transport-Knoten, wenn $b(i) = 0$

MCF-Problem (ein Transportproblem)

Berechne einen Fluss x in G, d.h. $x : E \longrightarrow \mathbb{R}_0^+$ mit

1. $0 \leq x_{ij} \leq u_{ij}$ für alle $(i, j) \in E$ (Kapazitätsbedinung)
2. $b(i) = \sum_{(i,j)\in E} x_{ij} - \sum_{(k,i)\in E} x_{ki}$ für alle $i \in V$ (Massenbalance)
3. $\sum_{(i,j)\in E} c_{ij} \cdot x_{ij}$ ist minimal.

Ein Fluss x der nur die Bedingungen 1 und 2 erfüllt, heißt **feasible** (siehe Übung).

Bemerkungen

1. Notwendige Bedingung (für einen feasible Flow)

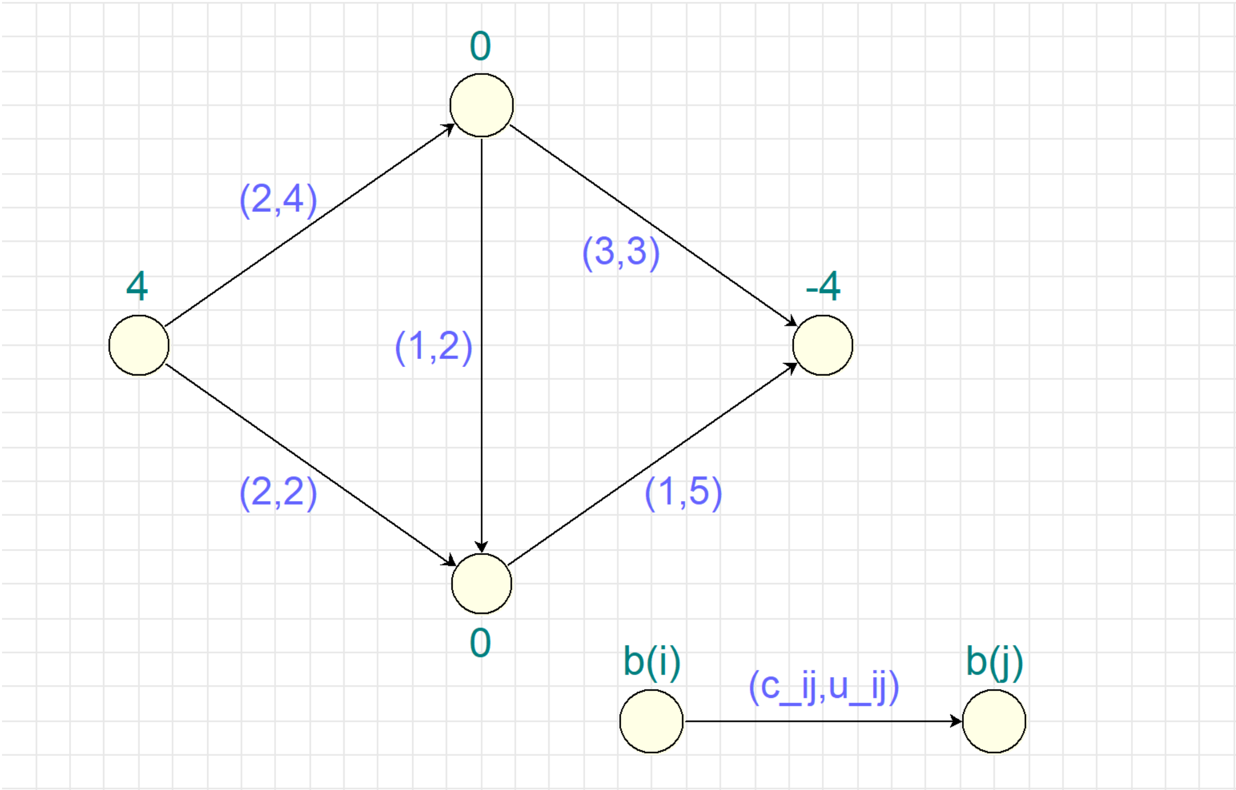
$$\sum_{i \in V} b(i) = 0$$

2. Ein Min-Cost-Flow x ist ein feasible Flow mit minimalen Gesamtkosten.

3. Ein MCF-Problem mit $b(i) = 0$ für alle $i \in V$ heißt **Zirkulationsproblem**.

4. Der Nullfluss $x = 0$ ist stets eine feasible Circulation und optimal, wenn $c_{ij} \geq 0$ für alle Kanten (i, j).

MCF-Beispiel



Pseudofluss (Wiederholung)

Eine Funktion $x : E \longrightarrow \mathbb{R}_0^+$ mit $0 \leq x_{ij} \leq u_{ij}$ für alle Kanten $(i, j) \in E$ heißt **Pseudofluss**.

Ein Pseudofluss verletzt i.A. die Massenbalance-Bedingung.

Der **Überschuss** oder **Excess** eines Knotens i

$$e(i) = b(i) - \sum_{(i,j)\in E} x_{ij} + \sum_{(k,i)\in E} x_{ki}$$

Ein Psudofluss x mit $e(i) = 0$ für alle $i \in V$ ist ein Fluss.

Restnetzwerk G(x) (mit Kosten)

Sei x ein (Pseudo-)Fluss in G

1. Für jede Kante (i, j) mit $x_{ij} < u_{ij}$ enthält $G(x)$ die Kante (i, j) mit Restkapazität $r_{ij} = u_{ij} - x_{ij}$ und Kosten c_{ij}

2. Für jede Kante (i, j) mit $x_{ij} > 0$, enthält $G(x)$ die Gegenkante (j, i) mit Restkapazität $r_{ji} = x_{ij}$ und Kosten $-c_{ij}$ (also negative Kosten).

Reduzierte Kosten

Sei $\pi : V \longrightarrow \mathbb{R}$ eine Funktion (Potentialfunktion)

$\pi(i)$ heißt **Potential** des Knotens i

Dann sind die reduzierten Kosten c_ij^π einer Kante (i, j) bzgl. des Potentials π wie folgt definiert

$$c_{ij}^{\pi} = c_{ij} + \pi(j) - \pi(i)$$

Beobachtungen

1. Reduzierte Kosten eines Pfads P von v nach w

$$\sum_{(i,j)\in P} \left(c_{ij}^{\pi}\right) = \sum_{(i,j)\in P} \left(c_{ij} - \pi(i) + \pi(j)\right) = \sum_{(i,j)\in P} c_{ij} - \pi(v) + \pi(w)$$

2. Reduzierte Kosten eines Kreises W

$$\sum_{(i,j)\in W} \left(c_{ij}^{\pi}\right) = \sum_{(i,j)\in W} \left(c_{ij} - \pi(i) + \pi(j)\right) = \sum_{(i,j)\in P} c_{ij}$$

Optimalitätsbedingungen

1. Negative Cycle Optimality

Satz 1:
Ein feasible Flow x ist Lösung des MCF-Problems (d.h. ist optimal), **genau dann wenn** das Restnetzwerk $G(x)$ keinen Zyklus mit negativen Kosten enthält.

Beweis : Übung.

Optimalitätsbedingungen

2. Reduced Cost Optimality

Satz 2:
Ein feasible Flow x ist **genau dann** optimal, **wenn** es ein Knotenpotential $\pi : V \longrightarrow \mathbb{R}$ gibt, so dass

$$c_{ij}^{\pi} \geq 0 \text{ für alle } (i, j) \text{ in } G(x)$$

Beweis (1)

$\Longleftarrow$

Es gilt $c_{ij}^{\pi} \geq 0$ für alle (i, j) in $G(x)$.

Dann hat ein beliebiger Zyklus W in $G(x)$ die Kosten

$$\sum_{(i,j)\in W} c_{ij} = \sum_{(i,j)\in W} c_{ij}^{\pi} \geq 0$$

d.h. $G(x)$ enthält **keinen negativen Zyklus** und die Optimalität folgt aus **Satz 1**.

Beweis (2)

$\Longrightarrow$

Sei x ein **optimaler Fluss**.

Dann folgt aus Satz 1: $G(x)$ hat keinen negativen Zyklus.

Sei $d : V \longrightarrow \mathbb{R}$ ein **Kürzestes-Wege-Labeling** der Knoten - kann z.B. mit dem Bellman-Ford Algorithmus mit beliebigem Startknoten s berechnet werden.

Aus der **Δ-Ungleichung** für d folgt für alle Kanten (i, j) in $G(x)$
$d(j) \leq d(i) + c_{ij}$ oder $c_{ij} + d(i) - d(j) \geq 0$

Dann gilt $c^{\pi}_{ij} \geq 0$ für das Potential $\pi(i) = -d(i)$.

Eine weitere Optimalitätsbedingung

3. Complementary Slackness Optimality

Ein feasible Flow x ist **genau dann** optimal, **wenn** es ein Knotenpotential π gibt, so dass für alle Kanten $(i, j) \in E$ gilt:

a) $c_{ij}^{\pi} > 0 \Rightarrow x_{ij} = 0$

b) $0 < x_{ij} < u_{ij} \Rightarrow c_{ij}^{\pi} = 0$

c) $c_{ij}^{\pi} < 0 \Rightarrow x_{ij} = u_{ij}$

Beweis : Übung.

Ein erster Algorithmus

Successive Shorteset Paths

Berechne Folge von optimalen Pseudoflows und dazugehörigen Potentialen

$$(x_0, \pi_0),\ (x_1, \pi_1),\ \ldots,\ (x_\ell, \pi_\ell) \text{ mit}$$

1. $x_0 = 0$ und $\pi_0 = 0$

2. $c_{ij}^{\pi_k} \geq 0$ für alle (i, j) in $G(x_k)$ für $0 \leq k \leq \ell$.

3. x_ℓ ist ein Fluss.

Successive Shortes Paths

1. $x \leftarrow 0$
2. $\pi \leftarrow 0$
3. Wähle in jedem Schritt einen Excess-Knoten s und einen Defizit-Knoten t d.h. $e(s) > 0$ und $e(t) < 0$
4. Berechne Kürzeste-Wege Distanzen $d(i)$ von s in $G(x)$ bzgl. der Kosten c_{ij}^{π}
5. $\pi \leftarrow \pi - d$
6. Erhöhe x entlang eines kürzesten Pfades P von s nach t um

$$\delta = \min\left(e(s), -e(t), \min_{(i,j)\in P} r_{ij}\right)$$

Laufzeitanalyse

1. Da stets $c_{ij}^{\pi} \geq 0$ gilt, können die Distanzen d in Zeile 4 des Algorithmus mit dem Algorithmus von Dijsktra berechnet werden.

2. Jede Iteration des Algorithmus vermindert den Gesamtüberschuss $\sum_{e(i)>0} e(i)$ um mindestens 1 Einheit.

3. Also ist die maximale Anzahl von Iterationen (d.h. Ausführungen von Dijkstra)

$$\leq \sum_{b(i)>0} b(i) \leq n \cdot U$$

4. Damit ist die Gesamtlaufzeit $\mathcal{O}\left(n \cdot U \cdot (n \log n + m)\right)$