§3 Der Algorithmus von Frank-Wolfe für die (konvexe) Optimierung mit linearen Nebenbedingungen

3.1 Der allgemeine Fall

$$\min f(x) \quad \text{unter} \quad Ax = b, \; x \geq 0$$
$$f: \mathbb{R}^n \to \mathbb{R}^1$$

Der Algo ist iterativ und erzeugt eine Folge von Punkten

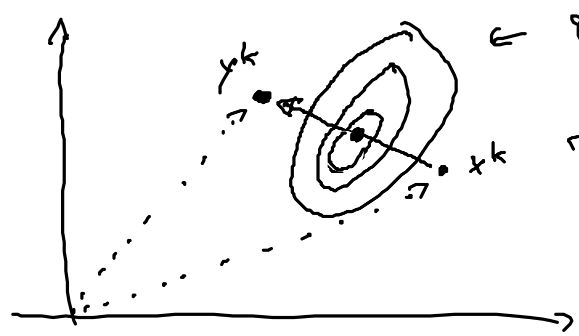
$$x^0, x^1, x^2 \ldots \quad x^k, x^{k+1} \ldots$$

x^{k+1} wird wie folgt aus x^k berechnet:

(1) Bestimme eine Optimallösung y^k des LP (lineares Problem!)

$$\min \underbrace{\nabla f^T(x^k)}_{ZF} \cdot x \qquad \text{"steilster Abstieg"}$$
$$\text{unter} \quad Ax = b$$
$$x \geq 0$$

(2) Wähle x^{k+1} als Minimierer von $f(x)$ auf dem Intervall $[x^k, y^k]$ "line search"



← Pkt mit gleichem ZF Wert $f(x)$

Richtung $y^k - x^k$

3.1 SATZ: Sei f stetig diffbar konvex und sei

$$X = \{ x \in \mathbb{R}^n \mid Ax = b \, , x \geq 0 \} \text{ beschränkt}$$

Dann enthält die Folge $x^0, x^1, x^2, \ldots$ eine konvergente Teilfolge und jede solche Teilfolge konvergiert gegen ein globales Minimum von f auf X

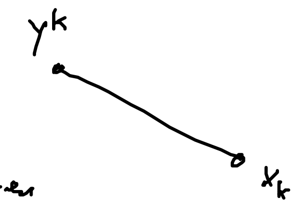
Bleiben 2 Fragen:

(1) Wie macht man die Line Search?

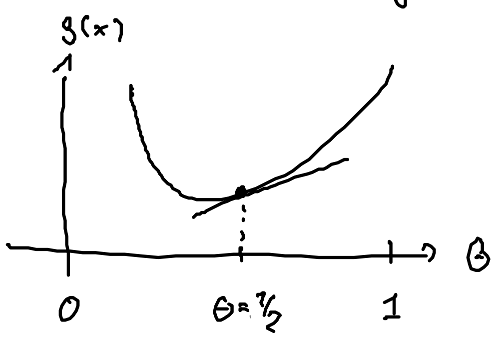
(2) Wann bricht man die Folge der x^k ab?

zu (1) Line Search:

eine Möglichkeit: binäre Suche auf



$\hat{=}$ Minimierung einer 1-dim. konvexen stetigen Fkt $g(\theta) = f(x^k + \theta(y^k - x_k))$ $\quad 0 \leq \theta \leq 1$



dann muss iterativ die Ableitung

$$\frac{dg}{d\theta}(\theta) = \nabla f(x^k + \theta(y^k - x^k))^T (y^k - x^k)$$

ausgewertet werden

Ist sie > 0 so $\theta := \theta - \theta/2$

Ist sie < 0 so $\theta := \theta + \theta/2$

Ist sie 0 oder das verbleibende Intervall

$(1/2)^i \, \| y^k - x^k \|$ klein genug, so STOP

Abbruch nach i Iterationen $(1/2)^i \, \| y^k - x^k \| < \varepsilon$

$\Rightarrow \quad i \approx \log\left(\frac{1}{\varepsilon} \cdot \| y^k - x^k \|\right)$ viele Auswertungen von ∇f

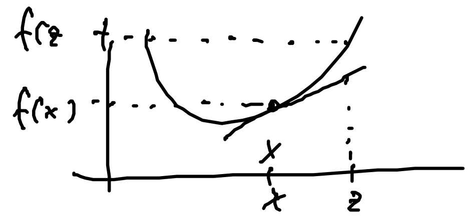
$\uparrow$

unter Umständen teuer (nicht bei SO)

daher auch andere Methoden betrachtet, z.B. Methode von Armijo
(num. Mathematik)

zu (2) Abbruchkriterium

f konvex $\underset{\text{diffbar}}{\Longleftrightarrow}$ $\quad f(z) \geq f(x) + \nabla f(x)^T \cdot (z - x) \quad \forall x, z$



x^* optimal $\underset{\substack{z = x^* \\ x = x^k}}{\Rightarrow}$ $\quad f(x^*) \geq f(x^k) + \nabla f(x^k)^T (x^* - x^k) \quad \forall k$

$\geq f(x^k) + \nabla f(x^k)^T (y^k - x^k) \quad \forall k$

$\uparrow$ y^k minimiert $\nabla f(x^k)^T \cdot x$

$\Rightarrow \quad \underbrace{f(x^k) - f(x^*)}_{\text{verbleibende Optimalitätslücke}} \leq \left| \nabla f(x^k)^T (y^k - x^k) \right|$

$\uparrow \quad \uparrow$ im Algorithmus bekannt

$\Rightarrow$ Optimalitätslücke kann im Verfahren kontrolliert werden

3.2 Die Spezialisierung auf das Systemoptimum (SO) und das User Equilibrium (UE)

(SO): $\quad \min \; C(x(f)) = \sum_{a \in A} x_a(f) \cdot \tau_a(x_a(f))$

mit $x_a(f) = \sum_{P \ni a} f_P$

$$\sum_{P \in \mathcal{P}_k} f_P = d_k$$

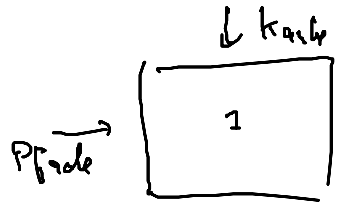
$$f_P \geq 0$$

$$\frac{\partial C}{\partial f_P}(x(f)) = \sum_{a \in P} \frac{d c_a}{d x_a}(x_a) \qquad c_a(x_a) = x_a \cdot \tau_a(x_a)$$

vgl. § 2

Also ist $\nabla C(\bar{x})^T x = \sum_{a \in A} \frac{d c_a}{d x_a}(\bar{x}_a) \cdot x_a = \sum_{a \in A} \frac{d c_a}{d x_a}(\bar{x}_a) \cdot \sum_{P \ni a} f_P$

Kantenweise



$$= \sum_P \sum_{a \in P} \frac{d c_a}{d x_a}(\bar{x}_a) \cdot f_P$$

$$= \sum_P \frac{\partial C}{\partial f_P}(\bar{f})\, f_P$$

Also: $\nabla C(\bar{x})^T x = \nabla C(\bar{f})^T f$

in Kantenvariable "nur" m — in Pfadvariable (exponentiell viele in m)

⇒ Das zu lösende LP im Frank-Wolfe also kann in Kantenform gelöst werden (brauchen nicht exponentiell vielen Wege in der Weg-Formulierung).

Die Bewertung der Kante ist $\frac{d}{d x_a} c_a(x^k) = x_a^k \cdot \tau_a'(x_a^k) + \tau_a(x_a^k)$

x^k = Kantenfluss in momentaner Iteration

≈ 0

$$\boxed{\begin{aligned} &\min \quad \nabla C(x^{\ell})^T x \\ &\text{unter} \quad \sum_{P \in \mathcal{P}_k} f_P = d_k \quad \forall\, k \in C \\ &\qquad\qquad f_P \geq 0 \end{aligned}}$$

x^{ℓ} momentane Iteration

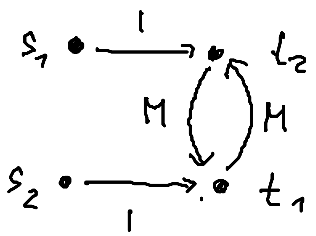
$\forall k \hat{=}$ Multi-Commodity-Fluss Problem

$\uparrow$

verschiedene s_k, t_k

MCF-Problem für ein festes k mit Kantenkosten $c_a'(x_a^{\ell})$

daher nicht als MCF-Problem auffassbar



$d_1 = d_2 = 1$

ABER: das Multi-Comm. Fluss Problem zerfällt hier (da keine Kantenkapazitäten) in $|C|$ viele kürzeste-Wege-Probleme

1-kürzester Weg für jede commodity k

$\Downarrow$

3.2 SATZ: Der erste Teil des Frank-Wolfe Algo. reduziert sich in jeder Iteration auf die Berechnung eines kürzesten Weges für jedes Quelle-Senke Paar (s_k, t_k) bzgl der Kantenkosten

$$c_a'(x_a^{\ell}) = x_a^{\ell} \cdot \tau_a'(x_a^{\ell}) + \tau_a(x_a^{\ell})$$

<u>Line Search:</u>

Sei y^{ℓ} die optimale Lösung des ersten Teils

$\Rightarrow$ y^ℓ besteht aus $|C|$ vielen Wegen Q_k, $k \in C$, mit $y_{Q_k} = d_k$

Vorige Lösung/Iteration f^ℓ besteht aus gewissen Wegen $P \in \mathcal{P}_k$

mit $\sum_{P \in \mathcal{P}_k} f_P^\ell = d_k$

Neue Lösung $f^{\ell+1} = f^\ell + \Theta(y^\ell - f^\ell)$

<u>3 Fälle:</u> a) $f_{Q_k}^\ell > 0$

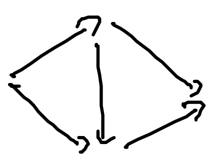
$\Rightarrow f_{Q_k}^{\ell+1} = f_{Q_k}^\ell + \Theta(d_k - f_{Q_k}^\ell) = (1-\Theta) f_{Q_k}^\ell + \Theta d_k$

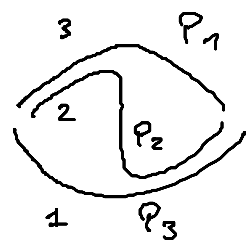
b) $f_{Q_k}^\ell = 0$ $\Rightarrow$ gleiche Formel $\Rightarrow f_{Q_k}^{\ell+1} = \Theta d_k$

c) $P \in \mathcal{P}_k$, $P \neq Q_k$ $(\Rightarrow y_P^\ell = 0)$

$\Rightarrow f_P^{\ell+1} = f_P^\ell - \Theta f_P^\ell = (1-\Theta) f_P^\ell$

<u>Bsp:</u>





$d_k = 6$ Iteration ℓ

$P_2 = Q_k$

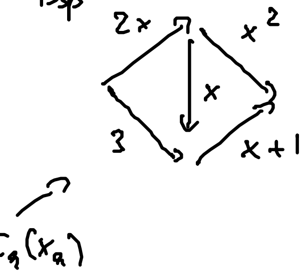
neuer Fluss in Iteration $\ell+1$

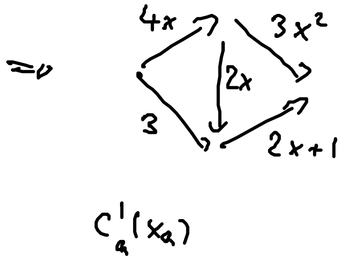
ergibt sich zu $\begin{pmatrix}3\\2\\1\end{pmatrix}(1-\Theta) + \Theta\begin{pmatrix}0\\6\\0\end{pmatrix} = \underbrace{\begin{pmatrix}3-3\Theta\\2+4\Theta\\1-\Theta\end{pmatrix}}_{\text{"}}$

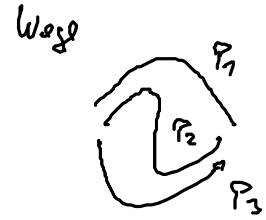
<u>Wert von Θ</u>

ergibt sich durch Line Search bzgl $\nabla C(\overbrace{x^\ell + \Theta(y^\ell - x^\ell)}^{\text{"}})^T(y^\ell - x^\ell)$ $(*)$

im Bsp







Wert von (*) für Θ:

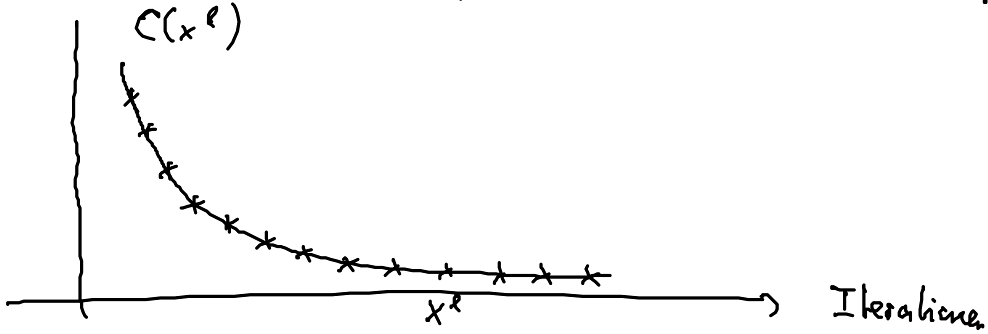
$$\begin{pmatrix} 4(5+\Theta) \\ 3(1-\Theta) \\ 2(2+4\Theta) \\ 3(3-3\Theta)^2 \\ 2(3+3\Theta)+1 \end{pmatrix}^T \cdot \begin{pmatrix} 6-5 \\ 0-1 \\ 6-2 \\ 0-3 \\ 6-3 \end{pmatrix} = g(\Theta)$$

Min bestimmen

3.3 Proposition: Line Search benötigt nur die Wege, die Fluss führen + die neu berechneten kürzesten Wege der momentanen Iteration (also nicht alle exponentiell vielen). Die Bestimmung von Θ kann im Kantenraum erfolgen

Abbruch entweder über die Optimalitätslücke (siehe § 3.1) oder über Beobachtung der Weglängen bzgl. $c_a'(x_a^\ell)$ [Abbruch wenn sie ungefähr gleich sind, d.h. die neu berechneten Wege Q_k unterscheiden sich kaum von den vorherigen, dies nutzt die Interpretation von SO als ein UE aus]

Typischerweise ist der Gewinn groß in den ersten Iterationen, später kleiner



Bemerkungen:

(1) Der Algo von Frank-Wolfe wird in diesem Zusammenhang auch als "convex combinations" Algo bezeichnet

$$x^{l+1} = (1-\Theta)x^l + \Theta y^l)$$

(2) Er kann auch für die Berechnung des UE benutzt werden

$$C_1(x(f)) = \sum_{a \in A} \int_0^{x_a(f)} \tau_a(t)\,dt$$

gleiche Nebenbed wie SO

$$\Rightarrow \quad c_a'(x_a) = \tau_a(x_a)$$

Ansonsten bleibt alles gleich

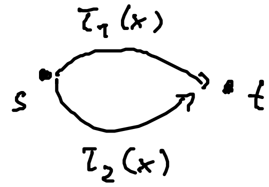
> **3.4 SATZ:** Bei der Berechnung des UE mit Frank-Wolfe reduziert sich der erste Teil einer Iteration auf die Berechnung eines kürzesten Weges für Quelle-Senke Paar k bzgl der Kantenkosten $\tau_a(x_a^l)$

(3) Naheliegender Fixpunkt-Ansatz

f^{l+1} = opt. Fluss bzgl. Fahrzeiten $\tau_a(x_a^l)$ aus der vorigen Iteration

Fkt. i.A. nicht, Oszillation kann eintreten

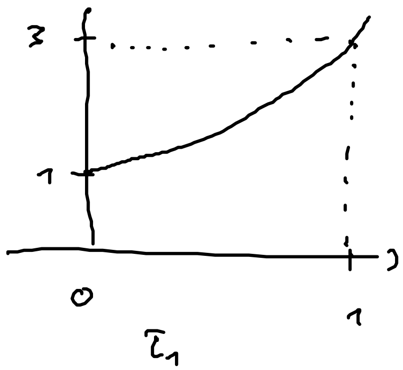
Bsp:

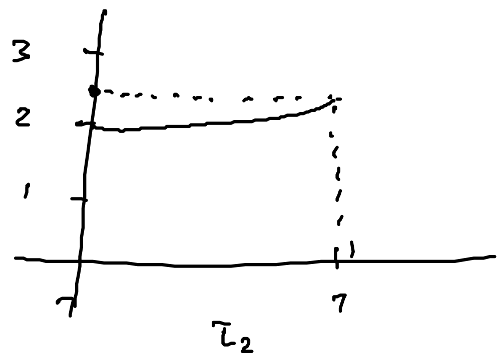


$d = 1$

$$\tau_1(x) = 1 + \frac{2x}{2-x}$$

$$\tau_2(x) = 2 + \frac{x/2}{2-x}$$

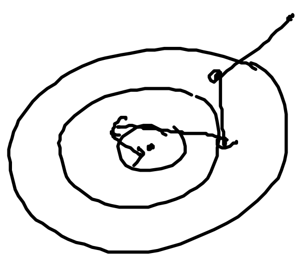




Startfluss $x_1 = 1 \quad x_2 = 0$

$\boxed{\text{Aufg. 4:}}$ Berechne die Folge der Flüsse mit "Fixpkt"-Ansatz und mit Frank Wolfe

(4) Bei Frank-Wolfe kann Zickzack Verhalten auftreten



$\Rightarrow$ $\exists$ verschiedene Verbesserungen

z.B. die Partan Methode

$\uparrow$

intelligentere Line Search durch Verwendung von $x^l, y^l, x^{l-1}, y^{l-1}$

$\Rightarrow$ 30% Verbesserung der Rechenzeit am Bsp Berlin

$\boxed{\text{Aufg. 5:}}$ Was muss geändert werden, wenn zusätzlich der Fluss pro Kante durch $x_a \leq u_a$ (Kapazität) beschränkt ist?