§3 Der Algorithmus von Frank-Wolfe für die (konvexe) Optimierung mit linearen Nebenbedingungen (1956)

3.1 Der allgemeine Fall:

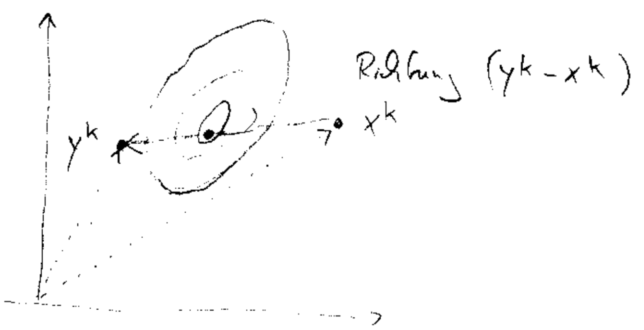
$$\min f(x) \quad \text{unter} \quad Ax = b \quad x \geq 0$$

Algorithmus ist iterativ und erzeugt eine Folge von Punkten $x^0, x^1, \dots$ wobei x^{k+1} wie folgt aus x^k berechnet wird

(1) Bestimme eine Optimallösung y^k des LP

$$\begin{aligned} \min \quad & Df^T(x^k) \cdot x \\ \text{unter} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

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



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 $x^0, x^1, \dots$ eine konvergente Teilfolge und jede solche Teilfolge konvergiert gegen ein globales Minimum von f auf X

Beweis der Konvergenz folgt einem sehr allgemeinen Prinzip, das gleich für Varianten oder Familien von Algorithmen gültig ist

Sind $A_1, \ldots, A_p$ Iterationsalgorithmen, so erzeugen sie, angewendet auf x^k die Menge $A(x^k) = \{A_1(x^k), A_2(x^k), \ldots, A_p(x^k)\}$

Abstrakt: Eine Klasse A von Iterationsalgorithmen ist eine mengenwertige Fkt $A : X \to 2^Y$

Schreibweise $A : X \xrightarrow{m} Y$ "point-to-set map"

Ein Algorithmus, der durch eine mengenwertige Funktion A gegeben ist ist <u>global konvergent</u>, wenn für jeden Anfangspunkt $x^0 \in X$ jede Folge (x^k) mit $x^{k+1} \in A(x^k)$ eine Teilfolge enthält, die gegen einen Punkt konvergiert, der notwendige Bedingungen für Optimalität erfüllt (z.B. Kuhn-Tucker)

Stetigkeit von "normalen" Fkt wird durch <u>Abgeschlossenheit</u> von mengenwertigen Funktionen verallgemeinert:

$A : X \xrightarrow{m} Y$ ist <u>abgeschlossen in $x \in X$</u> $:\Longleftrightarrow$

$$\left.\begin{array}{r} x^k \to x \text{ in } X \\ y^k \to y \text{ in } Y \\ \text{mit } y^k \in A(x^k) \end{array}\right\} \Rightarrow y \in A(x)$$

A heißt abgeschlossen auf $S \subseteq X$ wenn A in jedem $x \in S$ abgeschlossen ist

$f : X \to Y$ stetig $\Rightarrow$ $F : X \xrightarrow{m} Y$ mit $F(x) = \{f(x)\}$ abgeschlossen

Verkettung mengenwertiger Funktionen

$A : X \xrightarrow{m} Y \quad , \quad B : Y \xrightarrow{m} Z$

Die <u>Verkettung</u> C von A und B (bezeichnet als $B \circ A$) ist definiert als

$$C : A \xrightarrow{m} Z \quad \text{mit} \quad C(x) := \bigcup_{y \in A(x)} B(y)$$

Viele Algorithmen (Klassen) ergeben sich als solche Verkettung.

Auch der Frank-Wolfe Algorithmus:

$D \mathrel{\hat{=}}$ Ermittlung von y (LP lösen)

$U \mathrel{\hat{=}}$ Line Search

genauer: $D : X \xrightarrow{m} X \times \mathbb{R}^n$ mit $D(x^0) = \{ (x^0, d) \mid d = y - x^0$ und y ist Optimallösung von $\min \nabla f(x^0)^T x$ unter $Ax = b, x \geq 0$

Richtungen

$U : X \times \mathbb{R}^n \xrightarrow{m} \mathbb{R}^n \quad U(x^0, d) = \{ z \in [x^0, x^0 + d] \mid$ z minimiert/approximiert f auf $[x^0, x^0+d] = [x^0, y]$

3.2 PROPOSITION: Sei $A : X \xrightarrow{m} Y$, $B : Y \xrightarrow{m} Z$. Sei

(i) A abgeschlossen in $x \in A$ und B abgeschlossen auf $A(x)$

(ii) Y kompakt

Dann ist $C = B \circ A$ abgeschlossen in X

Beweis: Sei $x^k \to x$ in X, $z^k \to z$ in Z mit $z^k \in C(x^k)$

zeige: $z \in C(x)$

$z^k \in C(x^k) \Rightarrow \exists\, y^k \in \underbrace{A(x^k)}_{\subseteq Y}$ mit $z^k \in B(y^k)$

Y kompakt $\Rightarrow \exists$ konvergente Teilfolge $\{y^l\}$ von $\{y^k\}$
mit $y^l \to y$ in Y

A abgeschlossen in x $\Rightarrow$ $y \in A(x)$

$z^k \to z$ in Z $\Rightarrow$ $z^l \to z$ in Z

B abgeschlossen auf $A(x) \ni y$ $\Rightarrow$ $z \in B(y) \subseteq C(x)$ $\square$

<u>Zur Konvergenz:</u>

Betrachte Optimierungsproblem in X.
Sei $\Omega := \{x \in X \mid$ notwendige Bed. für Optimalität erfüllt $\}$
Sei A ein Algorithmus für dieses Problem repräsentiert durch
eine Mengenfunktion $A: X \xrightarrow{m} X$.

Globale Konvergenz von A (d.h. unabhängig vom Start x^0)
gegen ein $x \in \Omega$ hängt von 2 Sachen ab:

(1) Abgeschlossenheit von A auf $X \setminus \Omega$

(2) der Existenz einer Abstiegsfunktion $z: X \to \mathbb{R}$

$z: X \to \mathbb{R}^1$ heißt <u>Abstiegsfunktion (relativ zu A)</u> $:\Leftrightarrow$

(1) z ist stetig

(i) $x \notin \Omega \Rightarrow z(y) < z(x) \quad \forall y \in A(x)$

Algorithmus liefert pro Iteration bessere Punkte bzgl. z

(ii) $x \in \Omega \Rightarrow z(y) \le z(x) \quad \forall y \in A(x)$

und auf Ω keine schlechteren als x bzgl z

3.3 SATZ (Zangwill 69)

X, Ω wie oben. $A : X \xrightarrow{m} X$ ein Algorithmus dafür.

Sei $\{x^k\}$ eine von A erzeugte Folge, d.h. $x^{k+1} \in A(x^k)$

Es gelte:

(C1) Alle x^k liegen in einer kompakten Menge $K \subseteq X$

(C2) Es gibt relativ zu A eine Abstiegsfunktion z

(C3) A ist abgeschlossen auf $X - \Omega$

$A(x) \neq \emptyset$ für alle $x \in X - \Omega$

Dann enthält $\{x^k\}$ eine konvergente Teilfolge, und der Limes jeder konvergenten Teilfolge liegt in Ω

Beweis: $\{x^k\} \subseteq K$, K kompakt $\Rightarrow \exists$ konvergente Teilfolge $\{x^\ell\}$

Sei $x = \lim\limits_{\ell} x^\ell$

z stetig $\Rightarrow z(x^\ell) \to z(x)$

(a) $\{z(x^k)\}$ ist konvergent und $\lim z(x^k) = z(x)$

$x^{k+1} \in A(x^k) \overset{\text{Def } z}{\Rightarrow} z(x^{k+1}) \le z(x^k)$

$\Rightarrow \{z(x^k)\}$ und $\{z(x^\ell)\}$ sind monoton fallende Folgen

$z(x^\ell) \to z(x) \Rightarrow$ zu $\varepsilon > 0 \ \exists \ \ell_\varepsilon$ mit

$$z(x^\ell) - z(x) < \varepsilon \quad \forall \ \ell \geq \ell_\varepsilon$$

$$\Rightarrow z(x^k) - z(x) = \underbrace{z(x^k) - z(x^{\ell_\varepsilon})}_{\leq 0 \text{ für } k \geq \ell_\varepsilon} + z(x^{\ell_\varepsilon}) - z(x)$$

$$\leq z(x^{\ell_\varepsilon}) - z(x) < \varepsilon \qquad \text{für } k \geq \ell_\varepsilon$$

$\Rightarrow$ (a) und Monotonie

(b) $\underline{x \in \Omega}$

Betrachte Folge $\{x^{\ell+1}\} \subseteq K$

K kompakt $\Rightarrow \exists$ konvergente Teilfolge $(x^{\ell+1})_{\ell \in L'}$

mit $\lim\limits_{\ell \in L'} x^{\ell+1} =: x'$

$$\Rightarrow \left.\begin{array}{l} x^\ell \to x \\ x^{\ell+1} \to x' \\ x^{\ell+1} \in \underbrace{A(x^\ell)}_{\subseteq Y} \end{array}\right\} \text{ für } \ell \in L', \ \ell \to \infty \qquad (*)$$

Falls $x \in \Omega \Rightarrow$ fertig

Falls $x \notin \Omega \overset{(C3)}{\Rightarrow} A$ abgeschlossen in $x \overset{(*)}{\Rightarrow} x' \in A(x)$

$z(x^k) \to z(x) \Rightarrow z(x^{\ell+1}) \to z(x') = z(x)$

Teilfolgen monoton fallender konvergenter Folgen haben denselben Limes

$$\text{ABER} \left.\begin{array}{l} z \text{ Abstiegsfkt} \\ x \notin \Omega \\ x' \in A(x) \end{array}\right\} \Rightarrow z(x') < z(x) \text{ Widerspruch } \square$$

Bemerkungen:

(1) Die Abstiegsfunktion φ ist i.A. gleich der zu minimierenden Funktion f $\Rightarrow$ f muss als stetig vorausgesetzt werden

(2) Um (C1) zu erfüllen wird oft vorausgesetzt, dass

$$X_\alpha := \{ x \in X \mid f(x) \leq f(x^0) =: \alpha \}$$

beschränkt ist (*)

f stetig, (*) $\Rightarrow$ X_α kompakt und ist Menge K, die alle $\{x^k\}$ der Folge enthält

(3) (C1) und (C2) gelten meistens bei Abstiegsverfahren, aber (C3) ist kritisch und der Hauptgrund für fehlende Konvergenz

---

Beweis von Satz 3.1

Sei $\Omega := \{ x \in X \mid x \text{ ist Minimalpunkt von } f \}$

X beschränkt, abgeschlossen $\Rightarrow$ X kompakt

$\{x^k\} \subseteq X$ $\Rightarrow$ alle Punkte der Folge in kompakter Menge enthalten

$\Rightarrow$ (C1) von Satz 3.3.

$x^k \notin \Omega$ $\Rightarrow$ kein Minimalpunkt

$\Rightarrow$ in jeder ε-Umgebung von x^k gibt es besseren Punkt x

Taylor Entwicklung in x^k $\Rightarrow$

$$f(x) = f(x^k) + \nabla f(x^k)^T (x - x^k) + \underbrace{R_2(x)}_{\to 0}$$

$f(x) < f(x^k)$ $\Rightarrow$ $\nabla f(x^k)^T (x - x^k) < 0$ $\Rightarrow$ $\nabla f(x^k)^T x < \nabla f(x^k)^T x^k$

$\Rightarrow$ für Minimierer y von mit $\nabla f(x^k)\, x$ gilt

$$\nabla f(x^k)^T y < \nabla f(x^k)^T x^k$$

$\Rightarrow$ $y - x^k$ ist echt verbessernde Richtung

$\Rightarrow$ $f(x^{k+1}) < f(x^k)$ $\Rightarrow$ erste Bedingung aus (C2)

allgemein gilt $\nabla f(x^k)^T (y - x^k) \le \nabla f(x^k)(x^k - x^k) = 0$

$\uparrow$

minimiert $\nabla f(x^k) x$

$\Rightarrow$ Richtung $y - x^k$ ist nicht verschlechternd

$\Rightarrow$ $f(x^{k+1}) \le f(x^k)$ $\Rightarrow$ zweite Bedingung aus (C2)

<u>zu (C3):</u>

Frank-Wolfe ist Komposition der mengenwertigen Funktionen

D und U

D liefert zu x^k die möglichen Richtungen d, also

Paare (x^k, d) mit $d = (y - x^k)$

y Minimierer von $\nabla f(x^k)^T x$ unter $Ax = b,\ x \ge 0$

U liefert zu (x^k, d) die möglichen Minimierer / Approximierer

von f auf der Strecke $x^k,\ x^k + d = y$

zeige: a) D abgeschlossen in jedem $x \in X \setminus \Omega$

b) Bildbereich von U ist kompakt

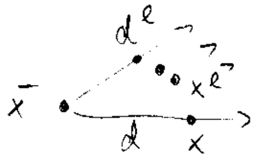
c) U ist abgeschlossen in jedem (x, d)

zu b): Bildbereich von U ist $X \times \{$mögliche Richtungen $y - x\}$

$y \in X,\ x \in X$, X kompakt

$\Rightarrow$ Bildbereich $\subseteq X + X$ kompakt zu c) $\left.\begin{array}{l}(\bar{x}, d^\ell) \to (\bar{x}, d) \\ x^\ell \in U(\bar{x}, d^\ell) \\ x^\ell \to x\end{array}\right\}$ $\overset{\text{zu zeigen}}{\Longrightarrow}$ $x \in U(\bar{x}, d)$

folgt aus Stetigkeit von f



x^ℓ minimiert bzgl. $(\bar{x}, d^\ell)$, $x^\ell \to x$, $d^\ell \to d$

$\Rightarrow$ für ℓ groß genug ist $f(x^\ell) \approx f(x)$
$d^\ell \approx d$

$\Rightarrow$ x minimiert bzgl. $(\bar{x}, d)$

[genauer mit Epsilontik]

zu a) D ist abgeschlossen in $\bar{x} \in X - \Omega$

Sei $\left.\begin{array}{l} x^k \to \bar{x} \\ y^k \to \bar{y} \quad \text{mit } (x^k, y^k - x^k) \in D(x^k)\end{array}\right\}$ $\overset{\text{zu zeigen}}{\Longrightarrow}$ $(\bar{x}, \bar{y}) \in D(\bar{x})$

$(x^k, y^k - x^k) \in D(x^k) \Rightarrow$ y^k minimiert $\nabla f(x^k)^T x$ unter $Ax = b$, $x \geq 0$

$\Rightarrow \nabla f(x^k)^T y^k \leq \nabla f(x^k)^T x \quad \forall x \in X$

halte x fest und lasse $k \to \infty$

$\Rightarrow$ (∇f stetig) $\nabla f(\bar{x})^T \cdot \bar{y} \leq \nabla f(\bar{x})^T x$ für jedes feste $x \in X$

$\Rightarrow$ $\bar{y}$ ist Minimierer von $\nabla f(\bar{x})^T x$ unter $x \in X$

$\Rightarrow (\bar{x}, \bar{y} - \bar{x}) \in D(\bar{x})$

(C1)-(C3) $\Rightarrow$ Behauptung mit Satz (3.3). $\square$

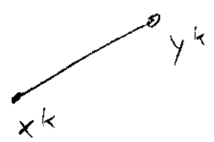
Bleiben 2 Fragen:

(A) Wie macht man die Line Search?

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

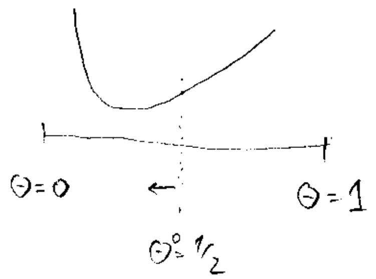
zu (A) Line Search

eine Möglichkeit: binäre Suche auf



≙ Minimierung einer 1-dimensionalen konvexen stetigen Fkt.

$$g(\theta) = f(x^k + \theta(y^k - x^k))$$



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

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

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

↑

unter Umständen teuer (nicht bei SO)

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

zu (B) Abbruchkriterium

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

x^* optimal $\underset{\text{mit } z = x^*}{\overset{\uparrow}{\Rightarrow}}$ $f(x^*) \geq f(x^k) + \nabla f(x^k)^T (x^* - x^k) \quad \forall\, k$

$\quad\quad$ $\llcorner$ $\{x^k\}$ Folge des Frank-Wolfe-Algo

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

$\quad\quad$ $\llcorner$ y^k minimiert $\nabla f(x^k)^T x$

$$\Rightarrow \underbrace{f(x^k) - f(x^*)}_{\substack{\text{verbleibende} \\ \text{"Optimalitätslücke"}}} \leq \underbrace{|\nabla f(x^k)^T (y^k - x^k)|}_{\text{im Algorithmus bekannte Größen}}$$

$\Rightarrow$ Optimalitätslücke kann im Verfahren kontrolliert werden

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

Im (SO) haben wir das Minimierungsproblem

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

$$\text{mit} \quad x_a(f) = \sum_{P\ni a} f_P \qquad f \text{ Pfadfluss}$$

$$\sum_{P\in \mathcal{P}_k} f_P = d_k \quad \forall\, k\in C$$

$$f_P \geq 0$$

$$\frac{\partial C}{\partial f_P}(x(f)) = \frac{\partial}{\partial f_P} \sum_{a\in A} \underbrace{x_a(f)\cdot \tau_a(x_a(f))}_{=:\, c_a(x_a(f))}$$

vgl. § 2.6

$$= \sum_{a\in A} \frac{\partial c_a}{\partial x_a}(x_a)\cdot \underbrace{\frac{\partial x_a}{\partial f_P}(f)}_{=1 \text{ wenn } a\in P \text{ und } 0 \text{ sonst}} \qquad \text{Kettenregel}$$

$$= \sum_{a\in P} \frac{\partial c_a}{\partial x_a}(x_a)$$

$$= \sum_{a\in P} \underbrace{\left(x_a\cdot \tau_a'(x_a) + \tau_a(x_a)\right)}_{c_a'(x_a) \text{ aus § 1}}$$

$$= \text{Länge des Weges } P \text{ bzgl. } c_a'(x_a)$$

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

$$= \sum_{P} \sum_{a\in P} \frac{dc_a}{dx_a}(\bar{x}_a)\cdot f_P = \sum_{P} \frac{\partial C}{\partial f_P}(\bar{f})\cdot f_P$$

$$= \nabla C_1(x(\bar{f}))^T x(f)$$

Also $\nabla C_1(\bar{f})^T f = \nabla C(\bar{x})^T \cdot x$

In Pfad-formulierung — In Kanten-formulierung

$\Rightarrow$ das zu lösende LP im Frank-Wolfe Algorithmus kann in Kantenform gelöst werden (brauchen nicht exponentiell viele Wege zu Formulierung)
und die Bewertung der Kante ist $\frac{dc_a}{dx_a}(x_a^l) = \underbrace{x_a^l \cdot \tau'(x_a^l) + \tau(x_a^l)}_{c_a'(x_a^k)}$
mit x^l = Kantenfluss in momentaner Iteration l

$\Rightarrow$ $\min \nabla C_1(x^l)^T x$

unter $\sum_{P \in \mathcal{P}_k} f_P = d_k \quad k \in C$

$f_P \geq 0$

$\hat{=}$ Min Cost MultiCommodity Fluss Problem mit Kantenkosten $c_a'(x_a^l)$

da keine Kapazitäten
$\Rightarrow$ zerfällt in kürzeste-Weg-Problem für jede Commodity k bzgl. Kantenkosten $c_a'(x_a^l)$

3.4 SATZ: Der erste Teil des Frank-Wolfe Algorithmus reduziert sich in jeder Iteration auf die Berechnung eines kürzesten Weges für jedes Quelle-Senke Paar bzgl. der Kantenkosten $c_a'(x_a^l) = x_a^l \tau_a'(x_a^l) + \tau_a(x_a^l)$

Line-Search:

Sei y^ℓ die optimale Lösung dieses ersten Teils

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

vorige Lösung f^ℓ besteht bereits aus gewissen Wegen $P \in \mathcal{P}_k$

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

Neue Lösung $f^{\ell+1}$ ergibt sich dann als

$$f^\ell + \Theta(y^\ell - f^\ell)$$

Fließt bereits Fluss auf Q_k, d.h. $f_{Q_k}^\ell > 0$

so ergibt sich für die entsprechende Komponente von $f^{\ell+1}$:

$$f_{Q_k}^{\ell+1} = f_{Q_k}^\ell + \Theta(d_k - f_{Q_k}^\ell) = \underbrace{(1-\Theta) f_{Q_k}^\ell + \Theta \cdot d_k}_{\text{Konvexkombination alte und neue Lösung}}$$

Fließt noch kein Fluss auf Q_k,

so ergibt sich die entsprechende Komponente von $f^{\ell+1}$ als

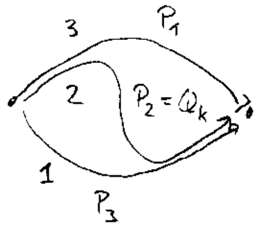
$$f_{Q_k}^{\ell+1} = f_{Q_k}^\ell + \Theta(d_k - f_{Q_k}^\ell) = \Theta d_k$$

und für alle Komponenten $P \in \mathcal{P}_k$, $P \neq Q_k$ wird Fluss weggenommen

$$f_P^{\ell+1} = f_P^\ell + \Theta(0 - f_P^\ell) = (1-\Theta) f_P^\ell$$

Beispiel:

$d_k = 6$



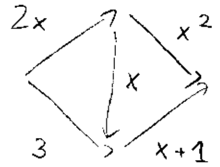
neue Fluss für Commodity k ergibt sich zu

$$\begin{pmatrix}3\\2\\1\end{pmatrix}(1-\Theta) + \Theta\begin{pmatrix}0\\6\\0\end{pmatrix} = \begin{pmatrix}3-3\Theta\\2+4\Theta\\1-\Theta\end{pmatrix} \geq 0$$

Wert von Θ ergibt sich durch die Line Search

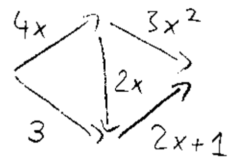
bzgl. $\nabla C(x^k + \Theta(y^k - x^k))(y^k - x^k)$ (*)

z.B.



$\tau_a(x)$

$\rightarrow$



$c'_a(x)$

$$\nabla C(x) = \begin{pmatrix} 4x \\ 3 \\ 2x \\ 3x^2 \\ 2x+1 \end{pmatrix}$$

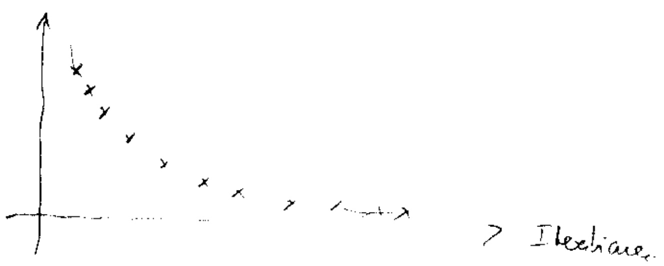
Wert von (*) für Θ:
$$\begin{pmatrix} 4\cdot(5+\Theta) \\ 3\cdot(1-\Theta) \\ 2\cdot(2+4\Theta) \\ 3\cdot(3-3\Theta)^2 \\ 2(3+3\Theta)+1 \end{pmatrix}^T \cdot \begin{pmatrix} 6-5 \\ 0-1 \\ 6-2 \\ 0-3 \\ 6-3 \end{pmatrix}$$

$\Downarrow$

3.5 PROPOSITION Line Search benötigt nur die Wege des Flusses führen + die neu berechneten Wege in der laufenden Iteration (also nicht alle potentiell exponentiell viele). Die Bestimmung von Θ kann im Kantenraum erfolgen.

Abbruch entweder über die Optimalitätslücke (§ 3.1) oder über Beobachtungen der Weglängen bzgl. $c'_a(x_a)$ (Abbruch wenn sie ungefähr gleich sind, d.h. neu berechnete Wege unterscheiden sich kaum von den vorherigen, dies nutzt die UE Interpretation von SO)

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



Bemerkungen:

(1) Das Verfahren von Frank-Wolfe wird in diesem Zusammenhang auch als convex combinations Algorithmus bezeichnet

(2) Es kann auch für die Berechnung des UE verwendet werden

$$C(x(f)) = \sum_{a \in A} \underbrace{\int_0^{x_a(f)} \tau_a(t)\,dt}_{C_a(x_a)}$$, Nebenbedingungen bleiben gleich

$$\Rightarrow C'_a(x_a) = \tau_a(x_a)$$

Ansonsten bleibt alles gleich!

3.6 SATZ Bei der Berechnung des UE mit Frank-Wolfe reduziert sich im ersten Teil die Berechnung auf die Berechnung eines kürzesten Weges für jedes Quelle-Senke Paar, bzgl. der Kantenkosten $\tau_a(x_a^\ell)$

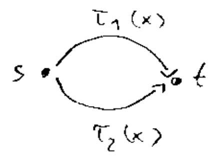
(3) Ein naheliegender "Fixpunkt-Ansatz" funktioniert i.A. nicht

$$f^{\ell+1} = \text{optimaler Fluss bzgl der Fahrzeiten } \tau_a(x_a^\ell(f^\ell))$$

aus der vorigen Iteration

da Oszillation auftreten kann

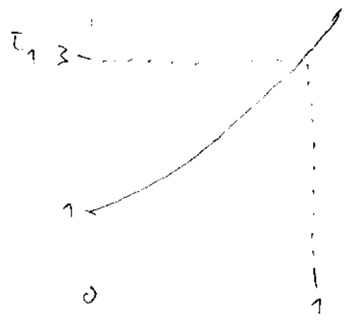
Beispiel:

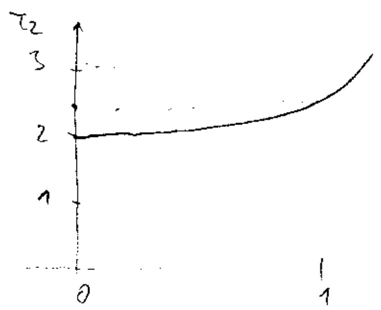


$d = 1$

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

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

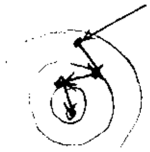




$\boxed{\text{Aufgabe 5}}$

Startfluss $x_1 = 1$, $x_2 = 0$. Berechne Folge der Flüsse mit "Fixpunkt-Ansatz" und mit dem Frank-Wolfe Algo

(4) Bei Frank-Wolfe kann Zickzack-Verhalten auftreten



Es gibt verschiedene Verbesserungen um dies zu vermeiden, z.B.
die Partan Methode (LeBlanc, Helgason, Boyce 85)

intelligentere Line Search durch Verwendung der
zwei letzten Iterationen, d.h. x^l, y^l und x^{l-1}, y^{l-1}
$\Rightarrow$ 30% Verbesserung der Rechenzeit in unseren Rechnungen

$\boxed{\text{Aufgabe 6}}$: Was muss geändert werden, wenn zusätzlich der Fluss pro Kante durch eine Kapazität beschränkt ist, d.h. $x_a \le u_a$?

3.3 Das eingeschränkte Systemoptimum

Motivation:

Vermeide Preis der Anarchie $\frac{UE}{SO}$

Vermeide Unfairness des SO (einzelne bekommen "lange" Routen)

Unfairness einer Routenzuweisung f

$$= \max_k \frac{\max \{ \tau_P(f) \mid P \in \mathcal{P}_k \}}{\text{Fahrzeit } L_k(UE) \text{ im } UE}$$

Bemerkung: unfairness (UF) = 1

unfairness (SO) -> ∞ (§ 1)

Daher Idee (Jahn, M., Schulz, Stier 05)

Lasse nur Wege zu, deren Fahrzeit bzgl. UE - Fahrzeiten

nur wenig über $L_k(f)$ liegen.

d.h. $\mathcal{P}_k^\varepsilon = \{ P \in \mathcal{P}_k \mid \sum_{a \in P} \tau_a(UF_a) \le (1+\varepsilon) L_k(UE) \}$

$\Downarrow$

Herleitung des Frank-Wolfe Algorithmus wie bisher, nur mit

Menge $\mathcal{P}_k^\varepsilon$ statt $\mathcal{P}_k$

hängt nicht vom aktuellen Fluss f ab

d.h. Nebenbedingungen sind immer $\begin{cases} \sum_{P \in \mathcal{P}_k^\varepsilon} f_P = d_k & k \in C \\ f_P \ge 0 \end{cases}$

=> Berechnung der Abstiegsrichtung führt auf die Aufgabe

Berechne pro Commodity $k \in C$ einen kürzesten Weg bzgl. $c_a'(x_a^\ell)$ aus $\mathcal{P}_k^\varepsilon$

$\downarrow$

<u>Constrained shortest path Problem</u> (CSP)

Gegeben: Digraph $G = (V, A)$, $s, t \in V$

2 Kantenbewertungen τ_a "Zeit"

ℓ_a "Länge", Schranke L

Aufgabe: Berechne kürzesten s,t Weg P bzgl. τ_a

mit $\ell(P) = \sum_{a \in P} \ell_a \leq L$

Leider Komplexitätssprung: CSP ist (schwach) NP-schwer

genaueres Studium in § 4