Otto-von-Guericke-Universität Magdeburg
Institut für Mathematische Optimierung
Prof. Dr. Friedrich Juhnke
Dr. Matthias Köppe, Gebäude 18, Raum 311

Wintersemester 2003/2004

2. Übungsblatt

Lineare Optimierung

`http://www.math.uni-magdeburg.de/~mkoeppe/lehre/opt1-2003`

Abgabe der Übungsaufgaben: bis Donnerstag, 30. Oktober, zu Beginn der Übung

5. Aufgabe **(10 Punkte)**

Eine Raffinerie mischt aus vier Rohtreibstoffen drei Sorten Benzin. Zu den Treibstoffen gibt es die folgenden Angaben.

Typ	Oktanzahl	Verfügbare Menge in Barrels/Tag	Preis in $/Barrel
1	68	4000	31.02
2	86	5050	33.15
3	91	7100	36.35
4	99	4300	38.75

Die drei Benzinsorten sollen mindestens 95, 90, bzw. 85 Oktan haben und können zu einem Preis von $45.15, $42.95, bzw. $40.99 pro Barrel verkauft werden. Von Sorte 1 können höchstens 10000 Barrels/Tag abgesetzt werden; von Sorte 3 müssen mindestens 15000 Barrels/Tag hergestellt werden. Rohtreibstoffe, die nicht zur Benzinherstellung verwendet werden, werden zu einem Preis von $38.95/Barrel ($36.85/Barrel) verkauft, wenn ihre Oktanzahl mindestens 90 (weniger als 90) beträgt.

(a) Formulieren Sie das Problem, den täglichen Gewinn der Raffinerie zu maximieren, als LP.

(b) Geben Sie das Problem im MPS-Format in den Rechner ein (`aufgabe5.mps`).

(c) Verwenden Sie den Befehl

```
glpsol --max --output aufgabe5.sol aufgabe5.mps
```

zum Lösen des Problems. In der Datei `aufgabe5.sol` ist dann die Lösung zu finden. Überprüfen Sie, ob die gefundene Lösung wenigstens zulässig für Ihr Modell ist.

6. Aufgabe **(10 Punkte)**

Sei $\emptyset \neq A \subseteq \mathbf{R}^n$. Der *Abschluß* $\operatorname{cl} A$ sei definiert als Vereinigung von A mit allen Häufungspunkten von A. Zu zeigen:

$$\operatorname{cl} A = \bigcap_{\epsilon > 0} A + \epsilon B,$$

wobei $B = \{\, \mathbf{x} \in \mathbf{R}^n : \|\mathbf{x}\| \leq 1 \,\}$ die abgeschlossene Einheitskugel ist.

7. Aufgabe **(10 Punkte)**

Seien $A, B, C \subseteq \mathbf{R}^n$. Beweisen Sie:

(a) $(A \cap B) + C \subseteq (A + C) \cap (B + C)$.

(b) Sind A, B abgeschlossen und $A \cup B$, C konvex, so gilt in (a) Gleichheit.

(c) $(A \cup B) + C = (A + C) \cup (B + C)$.

8. Aufgabe **(10 Punkte)**

Sei $C \subseteq \mathbf{R}^n$, $C \neq \emptyset$, abgeschlossen und konvex. Die Projektion $P_C(\mathbf{x}) \in C$ des Punktes $\mathbf{x} \in \mathbf{R}^n$ auf C ist (laut Vorlesung) definiert durch

$$\big\| P_C(\mathbf{x}) - \mathbf{x} \big\| \leq \big\| \boldsymbol{\xi} - \mathbf{x} \big\| \quad \forall \boldsymbol{\xi} \in C.$$

Dadurch ist eine Projektionsabbildung $P_C : \mathbf{R}^n \to C$ definiert.

(a) In der Vorlesung wurde gezeigt:

$$\big(\mathbf{x} - P_C(\mathbf{x})\big)^\top \big(\boldsymbol{\xi} - P_C(\mathbf{x})\big) \leq 0 \quad \forall \boldsymbol{\xi} \in C.$$

Zeigen Sie: Es gilt sogar

$$\big[\, \mathbf{p} = P_C(\mathbf{x}) \,\big] \iff \big[\, \mathbf{p} \in C \text{ und } (\mathbf{x} - \mathbf{p})^\top (\boldsymbol{\xi} - \mathbf{p}) \leq 0 \quad \forall \boldsymbol{\xi} \in C \,\big].$$

(b) Zeigen Sie: $P_C(\mathbf{x}) = \mathbf{x} \iff \mathbf{x} \in C$.

(c) Zeigen Sie:

$$\big\| P_C(\mathbf{x}) - P_C(\mathbf{y}) \big\|^2 \leq \big(P_C(\mathbf{x} - P_C(\mathbf{y})\big)^\top (\mathbf{x} - \mathbf{y}) \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{R}^n.$$

(d) Zeigen Sie:

$$\big\| P_C(\mathbf{x}) - P_C(\mathbf{y}) \big\| \leq \big\| \mathbf{x} - \mathbf{y} \big\| \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{R}^n,$$

d. h., P_C ist lipschitzstetig. Ist P_C gleichmäßig stetig auf $\mathbf{R}^n$?

Programmieraufgabe, Teil I **(20 Punkte)**

Ziel der Programmieraufgabe ist es, den im Verlauf der Vorlesung eingeführten Simplex-Algorithmus zur Lösung linearer Optimierungsprobleme zu programmieren. Ihr Programm soll in C geschrieben sein und muß auf unseren Rechnern laufen.

Der erste Teil der Programmieraufgabe ist bis zum **Freitag, 7. November**, zu bearbeiten.

Aufgabe: Schreiben Sie ein Programm, das in der Lage ist, eine MPS-Datei einzulesen. Zunächst müssen Sie nicht die eingelesenen Daten speichern (das wird in Teil II gemacht). Geben Sie nur die Anzahl der (Un-)gleichungen, die Anzahl der Variablen und die Anzahl der Nichtnullkoeffizienten aus.

Hinweise:

1. Zum Einlesen der Datei sind die folgenden Standardfunktionen nützlich: `fopen()`, `fclose()`, `fgets()`, `sscanf()`.

2. Um die Anzahl der Variablen und (Un-)gleichungen zu zählen, sollten Sie speichern, welche Variablen- und (Un-)gleichungsnamen Sie schon gesehen haben. Dazu empfiehlt es sich (im einfachsten Fall), je ein Array anzulegen, welches die Namen speichert.

   Sie dürfen ausnutzen, daß aufgrund der Definition des MPS-Formats ein Name höchstens 8 Zeichen lang ist. Damit Sie sich zunächst die dynamische Speicherverwaltung sparen können, dürfen Sie weiterhin annehmen, daß höchstens 1000 Variablen und höchstens 1000 Ungleichungen eingelesen werden; das sollte für einfache Testbeispiele ausreichen.

   Um zwei Namen zu vergleichen, können Sie die Standardfunktion `strcmp()` verwenden.