Technische
Universität
Braunschweig

Institute of Operating Systems
and Computer Networks

Mathematische Methoden der Algorithmik
1. Große Übung

Dominik Krupke

November 13, 2017

Organisation - Semesterplan

Mathematische Method

2. Simplex algorithm
3. Duality
4. Integer optimization

Schedule :

[Subscribe Calendar | Download Calendar]

Date	Description
24.10.2017, 15:00 Uhr	1. Vorlesung (SN19.3)
31.10.2017, 13:15 Uhr	Ausgabe 1. Hausaufgabe
07.11.2017, 15:00 Uhr	2. Vorlesung (SN19.3)
13.11.2017, 15:00 Uhr	1. Große Übung (SN19.3)
14.11.2017, 13:15 Uhr	Abgabe 1. Hausaufgabe, Ausgabe 2. Hausaufgabe (IZ)
14.11.2017, 15:00 Uhr	3. Vorlesung (SN19.3)
21.11.2017, 09:45 Uhr	1. Kleine Übung (IZ161)
21.11.2017, 15:00 Uhr	4. Vorlesung (SN19.3)
27.11.2017, 15:00 Uhr	2. Große Übung (SN19.3)
28.11.2017, 13:15 Uhr	Abgabe 2. Hausaufgabe, Ausgabe 3. Hausaufgabe (IZ)
28.11.2017, 15:00 Uhr	5. Vorlesung (SN19.3)
05.12.2017, 09:45 Uhr	2. Kleine Übung (IZ161)
05.12.2017, 15:00 Uhr	6. Vorlesung (SN19.3)
11.12.2017, 15:00 Uhr	3. Große Übung (SN19.3)
12.12.2017, 13:15 Uhr	Abgabe 3. Hausaufgabe, Ausgabe 4. Hausaufgabe (IZ)
12.12.2017, 15:00 Uhr	7. Vorlesung (SN19.3)
19.12.2017, 09:45 Uhr	3. Kleine Übung (IZ161)
19.12.2017, 15:00 Uhr	8. Vorlesung (SN19.3)
08.01.2018, 15:00 Uhr	4. Große Übung (SN19.3)
09.01.2018, 13:15 Uhr	Abgabe 4. Hausaufgabe, Ausgabe 5. Hausaufgabe (IZ)
09.01.2018, 15:00 Uhr	9. Vorlesung (SN19.3)
19.12.2018, 09:45 Uhr	4. Kleine Übung (IZ161)
16.01.2018, 15:00 Uhr	10. Vorlesung (SN19.3)
23.01.2018, 13:15 Uhr	Abgabe 5. Hausaufgabe (IZ)
23.01.2018, 09:45 Uhr	5. Große Übung (IZ161)
23.01.2018, 15:00 Uhr	11. Vorlesung (SN19.3)
29.01.2018, 15:00 Uhr	5. Kleine Übung (SN19.3)
30.01.2018, 15:00 Uhr	12. Vorlesung (SN19.3)

References :

1. B. Korte and J. Vygen: *Combinatorial Optimization: Theory and Algorithms*, Algorithms and Combinatorics, Springer, 2005 (kv-cota-05, BibTeX)

https://www.ibr.cs.tu-bs.de/courses/ws1718/mma/index.html <[1/1] 56%

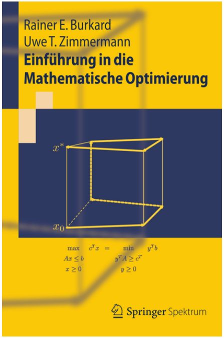
Organisation - Hausaufgaben

- 5 Blätter mit je 60 Punkten
- Studienleistung: 50% der HA-Punkte
- Die Blätter werden gegen Ende gewöhnlich schwerer und aufwendiger.
- Abgabe ist immer eine Woche vor der kleinen Übung.
- In der kleinen Übung werden die korrigierten Hausaufgaben zurückgegeben und besprochen.
- Euer Tutor ist Jannik Heroldt, j.heroldt@tu-bs.de
- Die kleinen Übungen können und sollen auch für grundlegende Verständisfragen genutzt werden.

Organisation - Skript und Literatur

- Es wird *eventuell* ein neues Skript geben. Schreibt trotzdem mit.
- Die Folien der Übungen werden online gestellt.

- Es gibt Unmengen an Literatur:
  - Die Zielgruppe variiert stark!
  - Die Notationen ebenfalls.
  - Ich finde das Buch von Burkard und Zimmermann für Einsteiger geeignet.
  - Kostenlos im Uni-netz erhältlich: `https://link.springer.com/book/10.1007/978-3-642-28673-5`
  - Für Fortgeschrittene gibt es bessere englische Literatur.
- Ihr könnt auch passende MOOCs finden.



Lineare Optimierung

- Variablen:
  - $x_1, x_2, x_3 \in \mathbb{R}$
  - In Linear Integer Programming erlauben wir auch $x_4 \in \mathbb{Z}$.
- Zielfunktion: Linear Kombination aus Variablen
  - $\min / \max 30 * x_1 + 40 * x_2$ also Konstante*Variable+Konstante*Variable*...
  - Nicht $\min x_1 * x_2$ oder $\max \sqrt{x_1}$
- Restriktionen: Lineare Gleichungen/Ungleichungen
  - $x_1 \geq 0$
  - $10 * x_1 + 20 * x_2 \leq 50$
  - $x_1 + 2 * x_3 = 5$

Schreibweise: Alles das Gleiche

$$\begin{aligned} \max \quad & c^T \mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} \leq b \\ & \mathbf{x} \geq 0 \end{aligned}$$

$$\begin{aligned} \max \quad & \sum_i c_i \mathbf{x_i} \\ & \textstyle\sum_i a_{ji} \mathbf{x_i} \leq b_j \quad \forall j \\ & \mathbf{x_i} \geq 0 \quad \forall i \end{aligned}$$

$$\max \quad \begin{bmatrix} c_1 & c_2 & c_3 \end{bmatrix} * \begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \mathbf{x_3} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} * \begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \mathbf{x_3} \end{bmatrix} \leq \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \mathbf{x_3} \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{aligned} \max \quad & c_1\mathbf{x_1} + c_2\mathbf{x_2} + c_3\mathbf{x_3} \\ & a_{11}\mathbf{x_1} + a_{12}\mathbf{x_2} + a_{13}\mathbf{x_3} \leq b_1 \\ & a_{21}\mathbf{x_1} + a_{22}\mathbf{x_2} + a_{23}\mathbf{x_3} \leq b_2 \\ & \mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3} \geq 0 \end{aligned}$$

Erstes Problem: Maschinenbelegung

Maschinenbelegungsproblem

- Was wollen wir?
  - Zwei Arten von Werkstücken herstellen welche auf drei unterschiedlichen Maschinen in beliebiger Reihenfolge bearbeitet werden.
- Was ist gegeben?
  - Maximale Laufzeit der Maschinen, Bearbeitungszeit der Werkstücke, Wert von Werkstücken.
- Was ist die Frage?
  - Wie viel Wert an Werkstücke kann maximal hergestellt werden?

Maschinenbelegungsproblem - Instanz

- Zwei Werkstücke W_1, W_2, drei Maschinen M_1, M_2, M_3.
- Ein Werkstück W_1 hat einen Wert von 40\$ muss 10 Stunden auf M_1, 10 Stunden auf M_2 und 20 Stunden auf M_3 bearbeitet werden.
- Ein Werkstück W_2 hat einen Wert von 50\$ muss 10 Stunden auf M_1, 30 Stunden auf M_2 und 10 Stunden auf M_3 bearbeitet werden.
- Auf Maschine M_1 stehen uns 8000 Arbeitsstunden zur Verfügung.
- Auf Maschine M_2 stehen uns 18000 Arbeitsstunden zur Verfügung.
- Auf Maschine M_3 stehen uns 14000 Arbeitsstunden zur Verfügung.

Maschinenbelegungsproblem - Instanz

	W_1	W_2	Laufzeit
M_1	10h	10h	8000h
M_2	10h	30h	18000h
M_3	20h	10h	14000h
Wert	40$	50$	

Idee: Wir nehmen einfach das 'billigste' Werkstück und produzieren so viel wie möglich davon!

$$W_1: \quad 700 * \begin{bmatrix} 10 \\ 10 \\ 20 \end{bmatrix} = \begin{bmatrix} 7000 \\ 7000 \\ 14000 \end{bmatrix} \leq \begin{bmatrix} 8000 \\ 18000 \\ 14000 \end{bmatrix} \Rightarrow 700 * 40\$ = 28000\$$$

$$W_2: \quad 600 * \begin{bmatrix} 10 \\ 30 \\ 10 \end{bmatrix} = \begin{bmatrix} 6000 \\ 18000 \\ 6000 \end{bmatrix} \leq \begin{bmatrix} 8000 \\ 18000 \\ 14000 \end{bmatrix} \Rightarrow 600 * 50\$ = 30000\$$$

Maschinenbelegungsproblem - Linear Program

Variablen: x_1 Anzahl von W_1, x_2 Anzahl von W_2.

$$\max 40 * x_1 + 50 * x_2$$

$$x_1 * \begin{bmatrix} 10 \\ 10 \\ 20 \end{bmatrix} + x_2 * \begin{bmatrix} 10 \\ 30 \\ 10 \end{bmatrix} \leq \begin{bmatrix} 8000 \\ 18000 \\ 14000 \end{bmatrix}$$

$$x_1, x_2 \geq 0$$

oder anders geschrieben:

$$\max 40 * x_1 + 50 * x_2$$

$$10 * x_1 + 10 * x_2 \leq 8000$$

$$10 * x_1 + 30 * x_2 \leq 18000$$

$$20 * x_1 + 10 * x_2 \leq 14000$$

$$x_1, x_2 \geq 0$$

Lösungen

Zulässige Lösungen:

$$x^{(1)} = \begin{bmatrix} 700 \\ 0 \end{bmatrix}, x^{(2)} = \begin{bmatrix} 0 \\ 600 \end{bmatrix}, x^{(3)} = \begin{bmatrix} 300 \\ 500 \end{bmatrix}, x^{(2)} = \begin{bmatrix} 100 \\ 100 \end{bmatrix}$$

Unzulässige Lösungen:

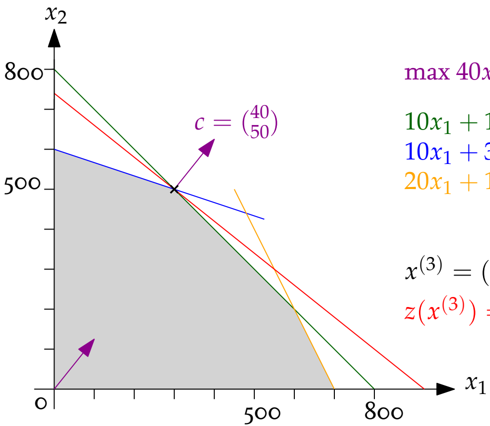
$$x^{(5)} = \begin{bmatrix} 200 \\ 600 \end{bmatrix}$$

$$M_1 : 200 * 10 + 600 * 10 = 8000 \leq 8000$$

$$M_2 : 200 * 10 + 600 * 30 = \mathbf{20000} > \mathbf{18000}$$

$$M_3 : 200 * 20 + 600 * 10 = 10000 \leq 14000$$

Graphisch



$\max 40x_1 + 50x_2$

$10x_1 + 10x_2 \leq 8000$
$10x_1 + 30x_2 \leq 18000$
$20x_1 + 10x_2 \leq 14000$

$x^{(3)} = \binom{300}{500}$
$z(x^{(3)}) = 37000$

Solver

- Kommerziell: IBM CPLEX, Gurobi, etc.
- Kostenlos für Studenten
- OSS: GLPK, LP_Solve.
- GLPK im Browser: `http://hgourvest.github.io/glpk.js/`

```
Maximize
40 x1 + 50 x2

Subject To
10 x1 + 10 x2 <= 8000
10 x1 + 30 x2 <= 18000
20 x1 + 10 x2 <= 14000

Bounds
0 <= x1
0 <= x2

End
```

Zweites Problem: Whisky Panschen

Whisky Panschen mit Linear Programming

Eine Whisky-Importgesellschaft hat zwar einen unbeschränkten Absatzmarkt für ihre Produkte, jedoch limitieren Importbeschränkungen den monatlichen Einkauf auf maximal 2000 L *Sir Roses* zu 35 Euro/L, 2500 L *Highland Wind* zu 25 Euro/L und 1200 L *Old Frenzy* zu 20 Euro/L. Daraus werden drei Mischungen (Blends) *A*, *B* und *C* zu den Preisen 34.00 Euro/L, 28.50 Euro/L bzw. 22.50 Euro/L hergestellt, die folgenden Anforderungen genügen müssen:

- A wenigstens 60% Sir Roses, höchstens 20% Old Frenzy
- B wenigstens 15% Sir Roses, höchstens 60% Old Frenzy
- C höchstens 50% Old Frenzy

Welche Mischungen ergeben den höchsten Profit?

Drittes Problem: Semesterplanung

Semesterplanung mit Integer Programming

	CP	Aufwand	Freude	Zeit
Astronomie	5	2	0	Do
Besenflugstunden	3	5	10	Mi
Geschichte der Zauberei	5	4	-5	Do
Kräuterkunde	4	4	5	Mi
Vert. gd dunklen Künste	10	10	20	Di
Verwandelung (Nötig: Zauberkunst)	6	8	10	Di
Zauberkunst	8	6	5	Mo
Zaubertränke	10	12	-10	Fr
Quidditch (Nötig: Besenflugstunden)	0	8	30	Do

- Mindestens 25 Credit Points
- Maximal 40h/Woche Aufwand
- Keine Wochentagskollisionen.
- Maximiere die Freude.