

## Quantitative Methoden 2. Gurobi/Python-Aufgabe

Abgabe: 07.07.2017

### Aufgabe 3 (Transportproblem)

(20 Punkte)

Es sei eine Menge von Gütern gegeben. Beim Transport einer Charge von Gut  $k$  fallen  $\text{kosten}[k]$  Euro pro Stunde an.

Des weiteren gibt es eine Menge von Städten. In jeder Stadt  $i$  liegen  $\text{verfuegbarkeiten}[i, k]$  Chargen des Gutes  $k$  vor, und in jeder Stadt  $i$  gibt es einen Bedarf von  $\text{bedarfe}[i, k]$  Chargen des Gutes  $k$ .

Die Fahrzeit zwischen Stadt  $i$  und Stadt  $j$  beträgt  $\text{fahrzeiten}[i, j]$  Stunden. Außerdem können insgesamt höchstens  $\text{kapazitaeten}[i, j]$  Chargen von Stadt  $i$  nach Stadt  $j$  transportiert werden.

Es ist nun zu entscheiden, wieviele Chargen von Gut  $k$  jeweils von Stadt  $i$  zu Stadt  $j$  transportiert werden sollen. Gehen Sie davon aus, dass die Güter nur direkt von einer Stadt zur anderen Stadt (d.h. ohne Umwege über andere Städte) transportiert werden. Dabei sollen die insgesamt anfallenden Transportkosten minimiert, aber alle Bedarfe gedeckt und die vorhandenen Kapazitäten nicht überschritten werden.

Modellieren Sie dieses Problem als LP und ergänzen Sie die Datei `transportmodel.py` an den in der Datei vorgegebenen Stellen, d.h. ergänzen Sie gegebenenfalls Attribute für die vorgegebenen Entscheidungsvariablen (z.B. `lb`, `ub`, `obj`) und ergänzen Sie die fehlenden linearen Nebenbedingungen (Constraints). Die Variable mit dem Namen  $x_{k,i,j}$  soll als Wert die Anzahl an Chargen von Gut  $k$  erhalten, die von Stadt  $i$  zu Stadt  $j$  transportiert werden. Sie können Ihr Modell testen, indem Sie den Code in `transportdata1.py` bzw. `transportdata2.py` ausführen (dort wird die von Ihnen ergänzte `solve()`-Methode aus `transportmodel.py` aufgerufen).

Fügen Sie an den Stellen, an denen Sie `transportmodel.py` ergänzen, Kommentare hinzu, was die entsprechenden Zeilen bedeuten. Das erhöht im Fall von falschen Bewertungen das Verständnis Ihres Modells und erleichtert Ihnen bei komplizierteren Modellen, den Überblick zu behalten.

Bitte laden Sie die von Ihnen ergänzte Datei `transportmodel.py` über das Abgabesystem

[https://orb.or.rwth-aachen.de/ss17\\_qm/](https://orb.or.rwth-aachen.de/ss17_qm/)

bis spätestens 07.07.2017 um 23:59 hoch. Sollten Sie noch keine Zugangsdaten per E-Mail erhalten haben, wenden Sie sich bitte an [walter@or.rwth-aachen.de](mailto:walter@or.rwth-aachen.de). Die Abgabe erfolgt einzeln, d.h. es gibt keine Gruppen. Nachdem Sie sich erstmals mit Ihrer E-Mail-Adresse und Ihrem Passwort (das Sie per E-Mail erhalten haben) im Abgabesystem einloggen, müssen Sie eine neue Gruppe, die nur aus Ihnen besteht, erstellen.

### Hinweise:

- Die Dateien `transportdata1.py` und `transportdata2.py` sollen nicht von ihnen verändert werden. Sie nehmen nur Ergänzungen an der Datei `transportmodel.py` vor und laden auch nur diese hoch.
- Die Dateien `transportdata1.py` und `transportdata2.py` müssen sich im gleichen Ordner wie die Datei `transportmodel.py` befinden.
- Schreiben Sie das Lineare Programm zunächst auf Papier. Wenn Sie von dessen Korrektheit überzeugt sind, fügen Sie nacheinander die Constraints hinzu, ergänzen gegebenenfalls die Attribute der Variablen (z.B. `lb`, `ub`, `obj`) und passen die Zielfunktion an.
- Orientieren Sie sich beim Hinzufügen der Constraints an den Beispielen aus den Veranstaltungen und Code von Aufgabe 0.
- Achten Sie darauf, dass Ihre Nebenbedingungen linear sind, d.h. insbesondere, dass keine Entscheidungsvariablen miteinander multipliziert werden. Modelle mit nichtlinearen Nebenbedingungen führen zu 0 Punkten.
- Achten Sie darauf, dass Sie keine Syntaxfehler einbauen und Ihr Code mit Python 2.7 läuft, da andernfalls mit 0 Punkten bewertet wird.
- Die optimalen Zielfunktionswerte für `transportdata1.py` bzw. `transportdata2.py` sind 3514 bzw. 3041.5.
- Sie dürfen auch zusätzlich eigene Variablen hinzufügen. Die vorgegebenen Variablen müssen aber mitverwendet werden – Änderungen am Namen der Variablen sind nicht erlaubt und führen zu einer Bewertung mit 0 Punkten.
- Beachten Sie, dass es nicht nur darum geht, bei den Instanzen den richtigen Zielfunktionswert zu berechnen, sondern das Problem korrekt zu modellieren.
- Korrekte Zielfunktionswerte auf den Testinstanzen führen nicht automatisch zur vollen Punktzahl.
- Zögern Sie nicht, Ihre Fragen im Forum oder an eine Suchmaschine im Netz zu stellen.