Laborationsinformation

Laboration 4: Lösning av heltalsproblem i Julia/JuMP

Syfte: Laborationen syftar till att ge övning i att lösa strukturerade heltalsoptimeringsproblem med hjälp av ett modelleringsspråk (i detta fall Julia/JuMP) samt ge viss insikt i relationerna mellan heltalsproblem och deras LP-relaxation. Dessutom skall laborationen påvisa möjligheterna att utnyttja ett problems specifika struktur, dels genom att modifiera problemformuleringen så att en bättre LP-relaxation erhålls.

Förberedelseuppgifter:

1. Läs lathunden `quickguide_Julia.pdf` om hur Julia/JuMP kan användas för att lösa optimeringsproblem.

Praktiskt:

- Notera att modellen skrivs på en fil och problemdata skrivs på en annan fil.
- För att öppna och editera textfiler i datorsalarna rekommenderas `vscode`.
- Skapa en mapp där du vill ha dina filer.
- Ladda ner filerna i mappen "Lab 4" på Lisam och lägg dem i den nyligen skapade mappen.
- Öppna ett terminalfönster och förflytta dig till mappen.
- För att få tillgång till Julia och VSCode, skriv:

```
module load courses/TAOP33
```

- Starta Julia genom att skriva i terminalen:

```
julia
```

- För att köra en Julia-fil `"filnamn.jl"`, skriv i Julia:

```
include("filnamn.jl")
```

Problemställning

Det optimeringsproblem som ska behandlas i denna laboration är det kapaciterade lokaliseringsproblemet. Låt m vara antalet möjliga platser för lokalisering av anläggningar (t.ex. fabriker), n antalet kunder, d_j efterfrågan hos kund j, s_i kapaciteten hos en anläggning på plats i, f_i den fasta kostnaden för att öppna en anläggning på plats i, c_{ij} transportkostnaden per enhet till kund j från anläggningsplats i, och e en diskonteringsfaktor. Samtliga koefficienter kan antas vara positiva.

Variabeldefinition: $y_i = 1$ om en anläggning placeras på plats i, och 0 om inte.
x_{ij} = transporterad mängd från anläggning på plats i till kund j.

Problemet att finna lösningen med lägst totalkostnad kan modelleras på följande sätt.

$$
\begin{array}{llllll}
v^* = & \min & \displaystyle\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} ef_i y_i & & & \\
& \text{s.t.} & \displaystyle\sum_{j=1}^{n} x_{ij} \leq s_i y_i & i = 1,\ldots,m & (1) & \\
& & \displaystyle\sum_{i=1}^{m} x_{ij} = d_j & j = 1,\ldots,n & (2) & \textbf{(P1)} \\
& & x_{ij} \geq 0 & i = 1,\ldots,m, j = 1,\ldots,n & (3) & \\
& & y_i \in \{0,1\} & i = 1,\ldots,m & (4) &
\end{array}
$$

Ofta avser egentligen kostnaderna olika tidsperioder. De fasta anläggningskostnaderna kan avse nybyggnation idag, medan transportkostnaderna kan avse samlade transportkostnader över en längre period (kanske flera år). Man måste då göra dessa kostnader jämförbara, och diskonteringsfaktorn e kan användas för detta. Man kan även prova olika scenarior (för t.ex. olika nivå på räntan). De första körningarna ska göras med $e = 1$.

Ibland finns en konflikt mellan fasta och rörliga kostnader. En plats med låga transportkostnader ligger troligen centralt, vilket leder till höga priser för mark och byggnation, dvs. höga fasta kostnader. Låga fasta kostnader uppträder troligen långt från kunderna, vilket medför höga trasnportkostnader.

En annan svårighet är att en anläggning antingen öppnas i sin helhet eller inte alls, även om bara en liten del av dess kapacitet behöver utnyttjas. LP-relaxationen av problemet är lättare att lösa, men ger ofta orealistiska lösningar där delar av anläggningar öppnas.

Ni ska lösa följande instanser av lokaliseringsproblemet.

Name	m	n
floc1	3	5
floc2	12	18
floc3	10	25
floc6	20	100
floc7	30	150
floc8	30	200

Dessa är givna i julia-datafiler med namnen floc1.jl, floc2.il, etc.

Laborationsuppgifter

Resultat och svar på frågor skrivs ner på separat resultatblad.

1. Betrakta den matematiska formuleringen P1. Hur många variabler och bivillkor finns i denna formulering, om vi har m platser för fabriker och n kunder? Specificera för floc6 och floc8. (Man räknar inte med $x \geq 0$ eller $y \in \{0, 1\}$.)

2. Fundera över uppgift 6. Hur många bivillkor får modellen P2? Jämför med svaret i uppgift 1 ovan. Specificera för floc6 och floc8.

3. Öppna och kika på innehållet i modellfilen "lokal.jl" samt datafilerna. Utgå ifrån formuleringen ovan och komplettera modellfilen med övriga definitioner, dvs. variabeldefinition, målfunktion och bivillkor.

4. Lös problem P1 genom att köra modellfilen "lokal.jl" med de olika instanserna. För varje instans, ange det optimala målfunktionsvärdet (dvs. minimala totalkostnader). Hur många trädsökningsnoder ("noder" i tabellen, se till höger i utskrifterna i terminalen) har genererats för varje instans? Hur många simplexiterationer ("iter" i tabellen, se till vänster i utskrifterna i terminalen) har gjorts? Hur lång tid tog lösandet (se "Time" i utskrifterna i terminalen)?

   På bilden nedan (som är utskriften för floc1), inringad med grönt, är antalet iterationer, slutligen 13 för detta exempel. Inringade i rött är antalet noder/LP-problem, den första är de som skapats men inte lösts och den andra är de lösta, så här tog det 3 noder.

```
>>> SOLVING LOKAL PROBLEM <<<

      0: obj =   0.000000000e+00 inf =   4.000e+02 (5)
      9: obj =   5.142115556e+03 inf =   0.000e+00 (0)
*    12: obj =   3.662115556e+03 inf =   0.000e+00 (0)
+    12: mip =     not found yet >=              -inf        (1; 0)
+    13: >>>>>   3.880000000e+03 >=   3.880000000e+03   0.0% (2; 0)
+    13: mip =   3.880000000e+03 >=     tree is empty   0.0% (0; 3)

 Time used: 8.702278137207031e-5
```

5. Lös *LP-relaxationen*, LP1, till dessa instanser genom att ta bort heltalskravet i modell P1. (Behåll övre gränsen 1 på y-variablerna.) LP-relaxationen ger en undre gräns till v^*. Hur stor är skillnaden i totalkostnad mellan heltalsoptimum och dess LP-relaxation? Den viktigaste delen av en lösning är vilka anläggningar som skall öppnas. Vilken information om detta ger LP-relaxationen? Studera och skriv upp y-lösningarna.

6. För ett heltalsproblem är det önskvärt att LP-relaxationen ger en uppskattning som ligger så nära heltalsoptimum som möjligt. Utgå från modell P1 och lägg till följande grupp av bivillkor. Kalla den nya modellen P2.

$$x_{ij} \leq d_j y_i \quad i = 1, \ldots, m, \; j = 1, \ldots, n$$

   Dessa bivillkor kan ge en starkare LP-relaxation (dvs. värden på y som ligger närmare 1 när de inte kan vara 0).

7. Lös LP-relaxationerna, LP2, med modell P2. Ger denna LP-relaxationen bättre uppskattningar och information? Jämför y-lösningarna med de i uppgift 5.

8. Lös heltalsproblemet med modell P2. Notera antalet trädsökningsnoder och antalet simplexiterationer, och jämför med resultatet i uppgift 4.

9. Lös floc3 med P1 för $e = 0.01$, 0.1, 1 (redan gjort), 10 och 100. Notera målfunktionsvärden, samt hur många anläggningar som öppnas i optimallösningen.