DATV: Introduktion til optimering og operationsanalyse

Multicommodity Flow Problem

David Pisinger, *Projektopgave 2*, blok 2, DIKU, 2005-06

Vejledende svar (på skitseform)

Svar 1 Optimale løsning er $z = 12$ med
$x^1_{23} = 1, x^1_{31} = 1$

$x^2_{32} = 1$

$x^3_{12} = \frac{1}{2}, x^3_{23} = \frac{1}{2}, x^3_{14} = \frac{1}{2}, x^3_{43} = \frac{1}{2}$

■

Svar 2 Modellen er:

```
minimize
 1x121+1x122+2x123+2x141+2x142+4x143+3x151+3x152+6x153+4x211
  +4x212+8x213+1x231+1x232+2x233+1x311+1x312+2x313+4x321+4x322
  +8x323+2x431+2x432+4x433+3x521+3x522+6x523
subject to
 1x231-1x311-1x321+1x431 = 0
 1x141-1x431 = 0
 1x151-1x521 = 0
 -1x122-1x142-1x152+1x212+1x312 = 0
 1x142-1x432 = 0
 1x152-1x522 = 0
 2x123-2x213-2x233+2x323+2x523 = 0
 2x143-2x433 = 0
 2x153-2x523 = 0
 1x211+1x231 = 1
 1x211+1x311 = 1
 1x312+1x322 = 1
 1x122+1x322+1x522 = 1
 2x123+2x143+2x153 = 2
 2x233+2x433 = 2
 1x121+1x122+2x123 <= 1
 1x141+1x142+2x143 <= 2
 1x151+1x152+2x153 <= 3
 1x211+1x212+2x213 <= 2
 1x231+1x232+2x233 <= 2
 1x311+1x312+2x313 <= 1
 1x321+1x322+2x323 <= 2
 1x431+1x432+2x433 <= 2
 1x521+1x522+2x523 <= 3
bounds
 0<=x121<=1
 0<=x141<=1
 0<=x151<=1
 0<=x211<=1
 0<=x231<=1
 0<=x311<=1
 0<=x321<=1
 0<=x431<=1
 0<=x521<=1
 0<=x122<=1
 0<=x142<=1
 0<=x152<=1
 0<=x212<=1
 0<=x232<=1
 0<=x312<=1
 0<=x322<=1
 0<=x432<=1
 0<=x522<=1
 0<=x123<=1
 0<=x143<=1
 0<=x153<=1
 0<=x213<=1
```

```
 0<=x233<=1
 0<=x313<=1
 0<=x323<=1
 0<=x433<=1
 0<=x523<=1
end
```

Den tilhørende løsning er $z = 12$ med

```
CPLEX> dis sol var -
Variable Name           Solution Value
x123                          0.500000
x143                          0.500000
x231                          1.000000
x233                          0.500000
x311                          1.000000
x322                          1.000000
x433                          0.500000
```

■

Svar 3 $x^1_{21} = 1, x^2_{12} = 1, x^2_{31} = 1, x^3_{14} = 1, x^3_{43} = 1$
Optimale løsning $z = 14$ ■

Svar 4 Tabellen er

vare k	rute r	d_k	pris pr enhed	samlet pris
1	2→1	1	4	4
1	2→3→1	1	2	2
2	3→1→2	1	2	2
2	3→1→5→2	1	7	7
2	3→2	1	4	4
3	1→4→3	2	4	8
3	1→5→2→3	2	7	14

■

Svar 5 Samlet model:

```
minimize
+4x1+2x2+2x3+7x4+4x5+8x6+14x7
subject to
        +1x3                 <= 1      \\ (1,2)
                    +2x6     <= 2      \\ (1,4)
            +1x4        +2x7 <= 3      \\ (1,5)
+1x1                         <= 2      \\ (2,1)
    +1x2                +2x7 <= 2      \\ (2,3)
    +1x2+1x3+1x4             <= 1      \\ (3,1)
                +1x5         <= 2      \\ (3,2)
                    +2x6     <= 2      \\ (4,3)
            +1x4        +2x7 <= 3      \\ (5,2)

+x1+x2 >= 1
+x3+x4+x5 >= 1
+x6+x7 >= 1
binary
  x1 x2 x3 x4 x5 x6 x7
end
```

■

Svar 6
```
Dual simplex - Optimal:  Objective =    1.4000000000e+01
Solution time =   -0.00 sec.  Iterations = 0 (0)

CPLEX> dis sol var -
Variable Name           Solution Value
x1                            1.000000
```

```
x3                              1.000000
x6                              1.000000
All other variables in the range 1-7 are zero.
CPLEX> dis sol dua -
Constraint Name             Dual Price
c6                           -2.000000
c10                           4.000000
c11                           4.000000
c12                           8.000000
All other dual prices in the range 1-19 are zero.
```

■

Svar 7 Samme løsning findes. ■

Svar 8 Eksempel 1 er en instans hvor path formuleringen giver en strammere nedre grænseværdi $z_{LP} = 14$ end den simple formulering $z_{LP} = 12$.

Enhver løsning til path problemet er også en løsning til den simple formulering, idet vi blot lader x_{ij}^k svare til path variablene. Flow bevarelse, strømning fra s og til t er oplagt overholdt. ■

Svar 9 y_{ij} skal tolkes således at man betaler prisen $-y_{ij}$ for at benytte kant (i, j).

$\bar{y}_k$ tolkes som en besparelse på $\bar{y}_k$ hvis man finder en alternativ rute fra s_k til t_k. ■

Svar 10 Reducerede omkostninger:

vare k	rute r	samlet pris	reduceret pris
1	2→1	4	0
1	2→3→1	2	-2
2	3→1→2	2	-2
2	3→1→5→2	7	3
2	3→2	4	0
3	1→4→3	8	0
3	1→5→2→3	14	6

Så vi kan vælge ruten 2→3→1 som har den mest negative reducerede omkostning. ■

Svar 11 Hvis ruten 2→3→1 tilføjes til modellen får vi de duale priser $y_{31} = -2$, mens øvrige $y_{ij} = 0$ og $\bar{y}_1 = 4, \bar{y}_2 = 4, \bar{y}_3 = 8$.

■

Svar 12 Processen standser når alle reducerede omkostninger er ikke-negative. ■

Svar 13 De reducerede omkostninger bliver:

vare k	rute r	samlet pris	reduceret pris
1	2→1	4	0
1	2→3→1	2	0
2	3→1→2	2	0
2	3→1→5→2	7	5
2	3→2	4	0
3	1→4→3	8	0
3	1→5→2→3	14	6

Da ingen ruter har negativ reduceret omkostning slutter vi. Vi har i alt benyttet 4 kolonner. ■

Hvis kolonnegenerering driller

CPLEX kan drille en del når man kører kolonnegenerering. Hvis man bruger formuleringen

```
bounds
0 <= x1 <= 1
```

vil CPLEX have en dual variabel svarende til begrænsningen, men den vil aldrig fortælle hvad den er. Derfor bliver de reducerede omkostninger forkert udregnet, og stop-kriteriet for kolonnegenerering nås aldrig.

Det samme problem kan opstå hvis man tilføjer begrænsningerne

```
0 <= x1 <= 1
```

som normale begrænsninger. CPLEX vil nu røbe den duale variabel, og hvis man er meget omhyggelig med de reducerede omkostninger, vil man få det rigtige svar, men der er mange faldgruber.

Det nemmeste er dog helt at udelade begrænsninger på x-variablene. Det er ikke nødvendigt at sikre $x \leq 1$ da x aldrig vil antage en større værdi, og $x \geq 0$ er underforstået for alle LP-problemer.

Kolonnegenerering uden det store skrivearbejde

For små-problemer som det betragtede kan man med fordel opskrive alle kolonner som i nedenstående fil. For de kolonner som p.t. ikke skal medtages i modellen, sættes den tilhørende x-variabel til 0. Dette sparer ikke alene en del skrivearbejde, men den duale variabel svarende til begrænsningen $x = 0$ vil netop angive den reducerede omkostning for den nye rute.

```
minimize
+4x1+2x2+2x3+7x4+4x5+8x6+14x7
subject to
        +1x3                <= 1      \\ (1,2)
                    +2x6    <= 2      \\ (1,4)
            +1x4        +2x7 <= 3     \\ (1,5)
+1x1                        <= 2      \\ (2,1)
    +1x2                +2x7 <= 2     \\ (2,3)
    +1x2+1x3+1x4            <= 1      \\ (3,1)
                +1x5        <= 2      \\ (3,2)
                    +2x6    <= 2      \\ (4,3)
            +1x4        +2x7 <= 3     \\ (5,2)

+x1+x2 >= 1
+x3+x4+x5 >= 1
+x6+x7 >= 1

x2 = 0
x3 = 0
x4 = 0
x7 = 0
end
----------------------------------------------------------------------
x2 = 0 x3 = 0 x4 = 0 x7 = 0

Dual simplex - Optimal:  Objective =    1.6000000000e+01
Solution time =    0.00 sec.  Iterations = 0 (0)
CPLEX> dis sol var -
Variable Name           Solution Value
x1                            1.000000
x5                            1.000000
x6                            1.000000
CPLEX> dis sol dua -
Constraint Name             Dual Price
c10                           4.000000
c11                           4.000000
c12                           8.000000
c13                          -2.000000
c14                          -2.000000
----------------------------------------------------------------------
x3 = 0 x4 = 0 x7 = 0

Dual simplex - Optimal:  Objective =    1.4000000000e+01
```

```
Solution time =    0.00 sec.  Iterations = 0 (0)
CPLEX> dis sol var -
Variable Name           Solution Value
x2                            1.000000
x5                            1.000000
x6                            1.000000
CPLEX> dis sol dua -
Constraint Name             Dual Price
c6                           -2.000000
c10                           4.000000
c11                           4.000000
c12                           8.000000
----------------------------------------------------------------------
```