Tentissä on viisi tehtävää, jotka arvosteellaan asteikolla 0-6. Tehtävien alakohdat ovat keskenään samanarvoisia ellei toisin mainita.

Tehtävä 1

Mitä seuraavat käsitteet tarkoittavat?

(a) Monitahokas (*polyhedron*). (1 p)
(b) Täydentyvyysehdot (*complementary slackness conditions*). (1 p)
(c) Simplex-algoritmin eksponentiaaliaikaisuus. (1 p)

Ovatko seuraavat väittämät oikein vai väärin? Perustele vastauksesi.

(d) Varjohinta (*shadow price*) kertoo kohdefunktion muutoksen kun sitä vastaavan muuttujan kerrointa kohdefunktiossa muutetaan yksi yksikkö. (1.5 p)
(e) Verkkotehtävän käypä puuratkaisu (*feasible tree solution*) voi olla degeneroitunut. (1.5 p)

Vastaukset

(a) Monitahokas on joukko, joka voidaan esittää äärellisellä määrällä lineaarisia epäyhtälörajoituksia.
(b) Täydentyvyysehdot ovat joukko optimaalisuusehtoja primaali- ja duaaliavaruuden käyville pisteille.
(c) Simplex-algoritmin eksponentiaaliaikaisuudella tarkoitetaan sitä, että kun algoritmi käy läpi käyvän alueen kulmapisteitä, voi näiden kulmapisteiden määrä kasvaa eksponentiaalisesti tehtävän dimension tai rajoitteiden määrän kasvaessa, jolloin pahin mahdollinen ratkaisuaikakin kasvaa eksponentiaalisesti.
(d) Väärin. Varjohintoja on yhtä monta kuin rajoituksia, jolloin muuttujiin ei mielekkäästi voi liittää varjohintaa.
(e) Oikein. Jos puussa on yksikin kaari jossa kulkee nollavirtaus, voi sen korvata millä hyvänsä muulla kannan ulkopuolisella kaarella jonka lisääminen ei muodosta puuhun kiertokulkua.

Tehtävä 2

Yritys voi investoida 100 tuhatta euroa kahdessa vaiheessa projektiin, jonka onnistumistodennäköisyys on p, tai obligaatioihin varmalla 7 % korolla. Jos yritys investoi projektiin ensimmäisessä vaiheessa saa se sijoitukselleen 500 % tuoton projektin onnistuessa mutta häviää kaiken projektin epäonnistuessa. Toisessa vaiheessa projektin onnistuminen on selvinnyt ja sijoitukselle saa 10 % tuoton jos projekti on onnistunut, muulloin projektiin ei voi investoida ja pääomalle tulee nollatuotto. Yrityksen riskirajoitukset määräävät että ensimmäisessä vaiheessa projektiin ei saa investoida yli 3 kertaa enempää kuin toisessa vaiheessa. Yritys pyrkii maksimoimaan odotusarvoisen tuottonsa ja on päätynyt seuraavaan optimointimalliin: Olkoon x_1 ja x_2 investoinnit vaiheessa 1 ja 2 ja x_3 investointi obligaatioihin. Tällöin maksimaalisen odotusarvon tuottaa strategia joka on ratkaisu ongelmalle

$$\begin{aligned} \max \quad & p \cdot (6x_1 + 1.1x_2) + (1-p) \cdot x_2 + 1.07x_3 \\ \text{s.e.} \quad & x_1 + x_2 + x_3 = 100 \quad \text{Budjetti} \\ & x_1 \leq 3x_2 \quad \text{Riskirajoitus} \\ & x_1, x_2, x_3 \geq 0\ . \end{aligned}$$

Yritys haluaa sinun toteuttavan herkkyyanalyysin parametrisella ohjelmoinnilla todennäköisydelle p:

(a) Muodosta tehtävälle Simplex-taulukko haluamallasi kannalla. (2 p)
(b) Sovella parametrista ohjelmointia todennäköisyyden $0 \leq p \leq 1$ suhteen. (3 p)
(c) Määritä optimaalinen kohdefunktion arvo p:n funktiona. (1 p)

Vastaukset

(a) Tehtävä on standardimuodossa

$$\begin{aligned} \min \quad & -6px_1 + (-1+0.1p)x_2 - 1.07x_3 \\ \text{s.e.} \quad & x_1 + x_2 + x_3 = 100 \\ & x_1 - 3x_2 + x_4 = 0 \\ & x_1, x_2, x_3, x_4 \geq 0 \ . \end{aligned}$$

Valitaan aloituskantaan x_3 ja x_4, jolloin saadan taulukoksi

107	$(1.07-6p)$	$(0.07+0.1p)$	0	0
$x_3 = 100$	1	1	1	0
$x_4 = 0$	1	-3	0	1

(b) Taulukko on optimaalinen jos $1.07 - 6p \geq 0 \Rightarrow p \leq 1.07/6 \approx 0.1783$ ja $0.07 + 0.1p \geq 0 \Rightarrow p \geq -0.07/0.1 = -0.7$, eli kun $0 \leq p \leq 1.07/6$ (todennäköisyydet positiivisia). Muulloin muuttujan x_1 redusoitu kustannus on negatiivinen ja se tulee kantaan ja muuttuja x_4 poistuu kannasta. Optimaalinen askelpituus on nolla.

107	0	$-17.9p + 3.28$	0	$6p - 1.07$
$x_3 = 100$	0	4	1	-1
$x_1 = 0$	1	-3	0	1

Taulukko on optimaalinen jos $p \geq 1.07/6 \approx 0.1783$ ja $-17.9 + 3.28p \geq 0 \Rightarrow p \leq 3.28/17.9 \approx 0.1832$, eli kun $1.07/6 \leq p \leq 3.28/17.9$. Muulloin muuttujan x_2 redusoitu kustannus on negatiivinen ja se tulee kantaan ja muuttuja x_3 poistuu kannasta. Optimaalinen askelpituus on 25.

$25 + 447.5p$	0	0	$4.475p - 0.82$	$1.525p - 0.25$
$x_2 = 25$	0	1	$\frac{1}{4}$	$-\frac{1}{4}$
$x_1 = 75$	1	0	$\frac{3}{4}$	$\frac{1}{4}$

Tämä taulukko on optimaalinen kun $p \geq 0.82/4.475 \approx 0.18324$ ja $p \geq 0.25/1.525 \approx 0.1639$ eli kun $p \geq 0.82/4.475$. Nyt kaikki p:n arvot on katettu.

(c) Kohdefunktion f arvo määräytyy kaavalla

$$f(p) = \begin{cases} 107, & 0 \leq p \leq 164/895 \approx 0.18324 \\ 25 + 447.5p, & 164/895 < p \leq 1 \end{cases}$$

Tehtävä 3

Kuntoilija suunnittelee ruoka-aineita aterioihinsa siten, että hän saa tietyn määrän kutakin ravinnetta mahdollisimman halvalla. Oheisessa taulukossa on ruoka-aineiden ravinnepitoisuudet (g/kg) ja hinnat (EUR/kg) sekä ravinteiden tavoitemäärät (g).

	Vihannekset	Liha	Vilja	Tavoitemäärä
Hinta	3	6	2	
Proteiini	100	400	50	100
Hiilihydraatti	300	200	600	200

(a) Muodosta ongelmasta lineaarisen ohjelmoinnin tehtävä ja sen duaali.

(b) Selitä yksityiskohtaisesti miten saamasi duaalitehtävä voidaan tulkita ravinnemyyjän hinnoitteluongelmana? (luennolla nk. Dieetti-ongelma)

(c) Ratkaise duaalitehtävä graafisesti ja laske duaalin ratkaisun perusteella primaalin ratkaisu.

Vastaukset

(a) Olkoon x_1, x_2, x_3 ostettujen vihannesten, lihan ja viljan määrät (kg). Tällöin ongelma on

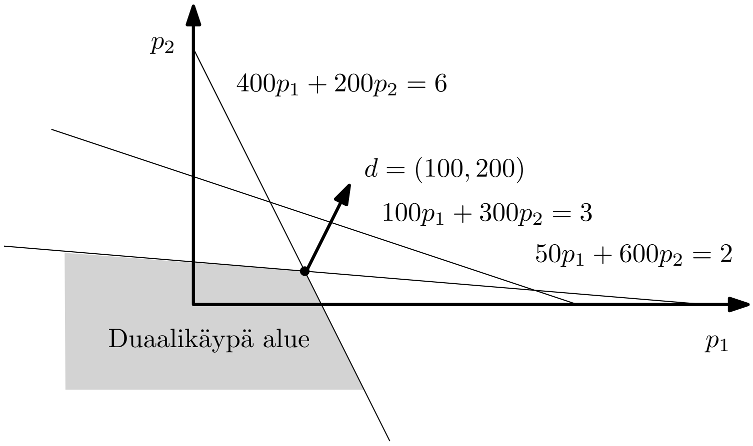
$$\begin{array}{rr} \min & 3x_1 + 6x_2 + 2x_3 \\ \text{s.e.} & 100x_1 + 400x_2 + 50x_3 = 100 \\ & 300x_1 + 200x_2 + 600x_3 = 200 \\ & x_1, x_2, x_3 \geq 0 \ , \end{array}$$

ja sen duaali on

$$\begin{array}{rr} \max & 100p_1 + 200p_2 \\ \text{s.e.} & 100p_1 + 300p_2 \leq 3 \\ & 400p_1 + 200p_2 \leq 6 \\ & 50p_1 + 600p_2 \leq 2 \ . \end{array}$$

(b) Ravinnemyyjä määrittää proteiinin ja hiilihydraatin optimaaliset hinnat p_1 ja p_2 (EUR/g) tietäen että asiakas tarvitsee 100 g proteiinia ja 200 g hiilihydraattia. Hän maksimoi tuottonsa $100p_1 + 200p_2$, mutta jotta asiakas ei ostaisi ravinteita vihannesten, lihan ja viljan muodossa pitää hänen varmistua siitä että ruoka-aineiden sisältämät ravinteet eivät ole edullisempia kuin ravinteina ostettuna. Esimerkiksi vihanneskilon sisältämä proteiinin ja hiilidydraattien kilohinta ravinnemyyjältä ostettuna on $100p_1 + 200p_2$, mikä ei saa ylittää vihannesten kilohintaa, joka on 3.

(c) Oheisen kuvaajan perusteella optimissa duaalin 2. ja 3. rajoitus aktiivisia, joten yhtälöpari $400p_1 + 200p_2 = 6$ ja $50p_1 + 600p_2 = 2$ antaa optimaalisen ratkaisun, joka on $p_1 = 8/575$ ja $p_2 = 1/460$ (EUR/g). Täydentyvyysehtojen perusteella primaalin ratkaisussa x_1 ei voi olla kannassa, sillä sitä vastaava rajoitus ei ole aktiivinen, joten primaalin ratkaisu saadaan yhtälöparista $400x_2+50x_3 = 100$ ja $200x_2+600x_3 = 200$, eli $x_1 = 0$, $x_2 = 5/23$ ja $x_3 = 6/23$.

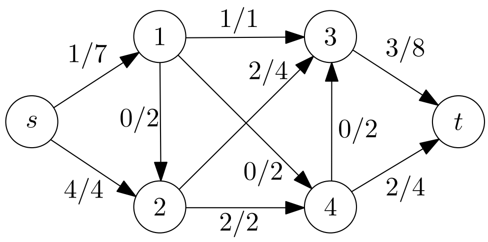


Tehtävä 4

Oheiseen kapasiteettirajoitettuun kuljetusverkkoon (ks. kuva 1) halutaan mahdollisimman suuri virtaus solmusta s solmuun t. Verkkoon on merkitty kapasiteettien lisäksi yksi käypä virtaus.

(a) Muodosta ongelmasta yleinen kapasiteettirajoitettu verkkovirtaustehtävä.

(b) Esitä maksimivirtaustehtävälle yksi iteraatio Ford-Fulkersonin-algoritmilla, jossa täydentävä polku etsitään nimikealgoritmilla, lähtien merkitystä virtauksesta. Terminoituuko algoritmi yhden iteraation jälkeen?



Kuva 1: Maksimivirtaustehtävä.

Vastaus

(a) Lisätään verkkoon kaari (t, s) jonka kustannus on -1 ja kapasiteetti on ∞. Tällöin tehtävä on

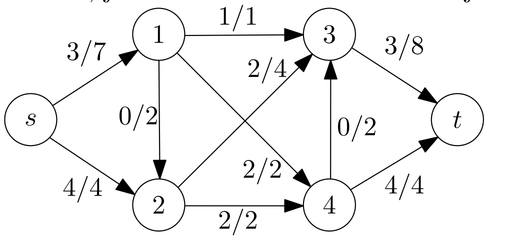
$$\begin{aligned} \min \quad & -f_{ts} \\ \text{s.e.} \quad & f_{ts} + f_{s1} + f_{s2} = 0 \\ & f_{s1} - f_{12} - f_{13} - f_{14} = 0 \\ & f_{s2} + f_{12} - f_{23} - f_{24} = 0 \\ & f_{13} + f_{23} + f_{43} - f_{3t} = 0 \\ & f_{14} + f_{24} - f_{43} - f_{4t} = 0 \\ & f_{3t} + f_{4t} - f_{ts} = 0 \\ & 0 \le f_{ij} \le u_{ij} \ , \end{aligned}$$

missä u_{ij} on kaaren (i, j) kapasiteetti.

(b) Etsitään täydentävää polkua nimikealgoritmilla:

1. $I = \{s\}$, poistetaan s, tutkitaan $(s, 1)$ $\Rightarrow$lisätään 1 ja $(s, 2)$ $\Rightarrow$ ei toimenpiteitä.
2. $I = \{1\}$, poistetaan 1, tutkitaan $(1, 3)$ $\Rightarrow$ei toimenpiteitä, $(1, 2)$ $\Rightarrow$ lisätään 2 ja $(1, 4)$ $\Rightarrow$ lisätään 4.
3. $I = \{2, 4\}$, poistetaan 4, tutkitaan $(4, 3)$ $\Rightarrow$lisätään 3, $(4, t)$ $\Rightarrow$ lisätään t.
4. $I = \{2, 3, t\}$. Koska $t \in I$, algoritmi päättyy. Täydentävä polku on $(s, 1, 4, t)$.

Pusketaan nimigealgoritmin löytämää polkua pitkin virtaa suurin mahdollinen määrä eli 2 yksikköä, jollon tilanne on oheisen kuvaajan mukainen.



Ford-Fulkersonin algoritimi terminoituu jos täydentävää polkua ei löydy tai löytyy täydentävä polku jota pitkin voidaan kasvattaa rajatta virtausta. Etsitään täydentävää polkua jälleen nimikealgoritmilla:

1. $I = \{s\}$, poistetaan s, tutkitaan $(s, 1)$ $\Rightarrow$lisätään 1 ja $(s, 2)$ $\Rightarrow$ ei toimenpiteitä.
2. $I = \{1\}$, poistetaan 1, tutkitaan $(1, 3)$ $\Rightarrow$ei toimenpiteitä, $(1, 2)$ $\Rightarrow$ lisätään 2 ja $(1, 4)$ $\Rightarrow$ ei toimenpiteitä.
3. $I = \{2\}$, poistetaan 2, tutkitaan $(2, 3)$ $\Rightarrow$lisätään 3, $(2, 4)$ $\Rightarrow$ ei toimenpiteitä.
4. $I = \{3\}$, poistetaan 3, tutkitaan $(4, 3)$ $\Rightarrow$lisätään 4, $(3, t)$ $\Rightarrow$ lisätään t.
5. $I = \{4, t\}$. Koska $t \in I$, algoritmi päättyy. Täydentävä polku on $(s, 1, 2, 3, t)$.

Koska täydentävä polku löytyy ja tätä polkua pitkin voidaan työntää vain rajallinen määrä virtausta, ei algoritmi terminoidu.

Tehtävä 5

Tarkastellaan kokonaislukutehtävää

$$\begin{aligned} \min \quad & x_{n+1} \\ \text{s.e.} \quad & 2x_1 + 2x_2 + \ldots, 2x_n + x_{n+1} = n \\ & x_i \in \{0, 1\} \forall i = 1, \ldots, n+1 \ . \end{aligned}$$

Näytä että mikä hyvänsä *Branch&Bound*-menetelmä, joka käyttää LP-relaksaatiota tehtävän alarajan laskemiseen ja jakaa osatehtävät asettamalla murtolukuarvoiset muuttujat joko arvoon 1 tai 0 vaatii eksponentiaalisen määrän osatehtäviä, kun n on pariton.

Vastaus

Olkoon $n = 2k + 1$ ja oletetaan että tehtävän LP-relaksaation ratkaisu on aina käypä kantaratkaisu. Ensimmäisen LP-relaksaatio antaa $x_{n+1} = 0$ ja tasan yksi jäljellä olevista muuttujista on $\frac{1}{2}$ (tämä siksi että $2 \sum_{i=1}^{n} x_i = 2k + 1$, joten kaikki

$x_i, i = 1, \ldots, n$ eivät voi olla 0 tai 1). Olkoon siis että muuttuja $x_1 = \frac{1}{2}$. Tällöin asetetaan että joko $x_1 = 0$ tai $x_1 = 1$, joita vastaavat seuraavat rajoitukset

$$2x_2 + \ldots + 2x_n + x_{n+1} = 2k + 1$$

ja

$$2x_2 + \ldots + 2x_n + x_{n+1} = 2k - 1 \ .$$

Koska kummallakin lisärajoituksilla saadaan LP-relaksaation optimikustannukseksi jälleen $x_{n+1} = 0$, ja tilanne on vastaava kuin alussa, voidaan todistusta jatkaa induktiivisesti. Tämä tarkoitttaa sitä että jokaisen muuttujan kohdalla on tehtävä jaettava kahteen osaan, joten ainakin 2^k LP-relaksaatiota on ratkaistava ennen kuin alkuperäisen tehtävän optimi löytyy. Näin ollen Branch & Bound tarvitsee eksponentiaalisen määrän aikaa tehtävän ratkaisemiseen. □