Täisarvuline planeerimine

2. november 2016. a.

Planeerimisülesannet nimetatakse täisarvuliseks, kui selles nõutakse, et kõik või osa muutujaid omandavad täisarvulisi väärtusi.

Juhul, kui täisarvulisuse nõue laineb kõikidele muutujatele, siis räägitakse ***puhttäisarvulisest*** ehk ***täielikult täisarvulisest*** planeerimisülesandest. Kui täisarvulisuse nõue on seatud muutujatele osaliselt, siis planeerimisülesannet nimetatakse ***osaliselt täisarvuliseks*** ülesandeks.

Muutujaid, mille väärtused peavad olema täisarvud, nimetatakse ***täisarvulisteks muutujateks***. Neid muutujaid, mille täisarvulisust ei nõuta, nimetatakse ***pidevateks muutujateks***.

Ülesannet, mis saadakse täisarvulise planeerimise ülesandest muutujate täisarvulisuse nõude ärajätmisel, nimetatakse sellele ülesandele vastavaks ***pidevaks ülesandeks***.

Järelikult täisarvulise lineaarse planeerimise ülesande võime esitada kujul

$$\begin{aligned} z = \mathbf{c} \cdot \mathbf{x} \quad &\to \quad \max, \\ A\mathbf{x} = \mathbf{b}, & \\ \mathbf{x} \geqslant 0 \quad &\text{ning} \quad x_j \in \mathbb{Z} \quad \text{ja } j \in T, \end{aligned} \tag{1}$$

kus T on kõikide muutujate indeksite hulga $\{1, 2, \dots, n\}$ mingi alamhulk.

Planeerimise ülesannet (1), kus $T = \{1, 2, \dots, n\}$, nimetatakse ***puhttäisravuliseks lineaarse planeerimise ülesandeks***.

Muudel juhtudel on tegemist ***osaliselt täisarvuliste lineaarsete planeerimisülesannetega***.

Harilikult sihifunktsiooni väärtusele täisarvulisuse nõuet ei seata.

Ignoreerime muutujate täisarvulisuse nõuet ja lahendame ülesande (1) asemel ülesande

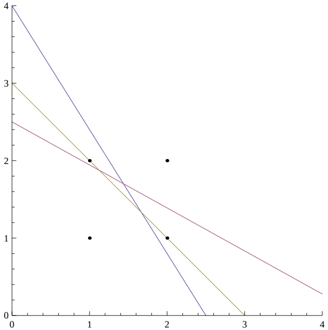
$$\begin{aligned} z = \mathbf{c} \cdot \mathbf{x} \quad &\to \quad \max, \\ A\mathbf{x} = \mathbf{b}&, \\ \mathbf{x} \geqslant 0&. \end{aligned} \tag{2}$$

Lahendada graafiliselt täisarvuline lineaarse planeerimise ülesanne:

$$\begin{aligned} z = x + y \qquad\qquad &\to \max, \\ 8x + \ \ 5y \leqslant 20&, \\ 10x + 18y \leqslant 45&, \\ x \geqslant 0, y \geqslant 0 \qquad &x, y \in \mathbb{Z}. \end{aligned}$$

Pideva ülesande lahend:

$$\left\{\frac{295}{94},\left\{x\to\frac{135}{94},y\to\frac{80}{47}\right\}\right\} \quad \text{ehk} \quad \{3.1383,\{x\to 1.43617,y\to 1.70213\}\}$$



Täisarvulise ülesande lahend:

$$\{2.,\{x\to 1,y\to 1\}\}$$

Juhul, kui ülesande (2) optimaalne lahend $\mathbf{x}^* = \begin{bmatrix} x_1^* & x_2^* & \dots & x_n^* \end{bmatrix}^T$ on ülesande (1) lubatavaks lahendiks, siis on $\mathbf{x}^*$ ka selle ülesande optimaalseks lahendiks.

Juhul, kui lahendil $\mathbf{x}^*$ leidub mittetäisarvuline komponent x_i^*, kus $i \in T$, siis konstrueeritakse uus lineaarne kitsendus, mis lõikab ülesande (2) lubatavate lahendite hulgast välja punkti $\mathbf{x}^*$ koos selle mingi ümbrusega, kuid säilitab kõik ülesande (1) lubatavad lahendid.

Selliste omadustega kitsendust nimetatakse ***lõikekitsenduseks***.

Lõikekitsendus lisatakse ülesandele (2) ja see lahendatakse uuesti.

Nii jätkatakse seni, kuni saadakse ülesande (1) lubatav lahend, mis on ühtlasi selle ülesande optimaalseks lahendiks.

Lõikekitsendusi kasutavaid lahendusmeetodeid nimetatakse ***lõikemeetoditeks***. Lõikekitsendusega ülesande lahendamiseks saab kasutada eelnavalt lahendatud ülesande optimaalsel kujul olevat simplekstabelit. Seetõttu seisneb lõikekitsendusega ülesande lahendamine mõne simplekssammu teostamist.

Lõikemeetodi realiseerimisel oli põhiliseks raskuseks niisuguste lõikekitsenduste konstrueerimine, mille korral saadav lahendusalgoritm oleks lõplik. Esimesena, 1958. aastal, pakkus sellise lõikekitsenduse välja ameerika matemaatik Ralph Edward Gomory (s 1929).

Olgu (1) puhttäisarvuline ülesanne ja sellele vastava ülesande (2) simplekstabel optimaalne.

Siis saame sihifunktsiooni ja kitsendused esitada kujul:

$$\begin{aligned} z &= b_0 + \sum_{j \in J} c_j(-x_j), \\ x_i &= b_i + \sum_{j \in J} a_{ij}(-x_j) \quad (i \in I), \end{aligned} \tag{3}$$

kus I on optimaalsele baasilahendile vastav baasiindeksite hulk, J - baasiväliste indeksite hulk ja $T = I \cup J = \{1, 2, \ldots, n\}$.

Siit saadava optimaalse baasilahendi $\mathbf{x}^* = \begin{bmatrix} x_1^* & x_2^* & \ldots & x_n^* \end{bmatrix}^T$ tõttu

$$c_j \geqslant 0 \ \ (j \in J) \quad \text{ja} \quad b_i \geqslant 0 \ \ (i \in I).$$

Vaatleme juhtu, kui $\mathbf{x}^*$ ei ole ülesande (1) lubatavaks lahendiks. Siis vähemalt üks lahendi $\mathbf{x}^*$ komponentidest ei ole täisarvuline. Olgu selleks $x_k^* = b_k$ ja süsteemist (3) leiame vastava kitsenduse

$$x_k = b_k + \sum_{j \in J} a_{kj}(-x_j). \tag{4}$$

$[a]$ - arvu täisosa, st suurim täisarv, mis ei ületa arvu a.

$\{a\} = a - [a]$ - arvu murdosa.
Näiteks:

$$[3.8] = 3 \qquad [-3.8] = -4$$
$$\{3.8\} = 0,8 \qquad \{-3.8\} = 0,2.$$

$$\begin{aligned} b_k &= [b_k] + \{b_k\} \quad \text{kus} \quad 0 < \{b_k\} < 1, \\ a_{kj} &= [a_{kj}] + \{a_{kj}\} \quad \text{kus} \quad 0 \leqslant \{a_{kj}\} < 1. \end{aligned} \tag{5}$$

Asendame need kitsendusse (4) saame

$$x_k = [b_k] + \{b_k\} + \sum_{j \in J} ([a_{kj}] + \{a_{kj}\})(-x_j). \tag{6}$$

Toome murdosad vasakule poole

$$-\{b_k\} + \sum_{j \in J} \{a_{kj}\} x_j = -x_k + [b_k] + \sum_{j \in J} [a_{kj}](-x_j). \tag{7}$$

Ülesande (1) muutujate lubatud väärtuste korral on selle võrduse parem pool täisarv, järelikult ka vasak pool peab olema samuti täisarv. Nõudest $x_j \geqslant 0$ ja tingimustest (5) järeldub

$$\sum_{j \in J} \{a_{kj}\} x_j \geqslant 0 \quad \text{ning} \quad -1 < -\{b_k\} < 0.$$

Järelikult seose (7) vasaku poole jaoks kehtib hinnang

$$-\{b_k\} + \sum_{j\in J}\{a_{kj}\}x_j > -1 + \sum_{j\in J}\{a_{kj}\}x_j > -1. \tag{8}$$

Kuna see peab olema täisarv, siis kehtib veelgi rangem hinnang

$$-\{b_k\} + \sum_{j\in J}\{a_{kj}\}x_j \geqslant 0. \tag{9}$$

Tähistame selle võrratuse vasaku poole

$$x_{n+1} = -\{b_k\} + \sum_{j\in J}\{a_{kj}\}x_j \geqslant 0, \tag{10}$$

kus x_{n+1} on eelnava põhjal mittenegatiivne täisarv. Selle lineaarse tingimuse võime esitada kujul

$$x_{n+1} = ([b_k] - b_k) - \sum_{j\in J}([a_{kj}] - a_{kj})x_j, \quad x_{n+1} \in \mathbb{Z}^+. \tag{11}$$

$$x_{n+1} = ([b_k] - b_k) - \sum_{j \in J} ([a_{kj}] - a_{kj}) x_j, \quad x_{n+1} \in \mathbb{Z}^+. \qquad (11)$$

See on uus lineaarne kitsendus, mis on kitsendustega (3) sama tüüpi, kuid lisandunud on mittenegatiivne täisarvuline abimuutuja x_{n+1}. Järelikult lisandunud abimuutuja võime praegu lugeda baasimuutujaks nagu esialgseid muutujaid x_i, kus $(i \in I)$.

Kitsendus (11) sobib lõikekitsenduseks, sest see on konstrueeritud nii, et ta täidetud ülesande (1) iga lubatud lahendi korral, st ei lõika ülesande (2) lubatavate lahendite hulgast välja ühtegi täisarvuliste komponentidega lahendit.

Kitsendus (11) ei ole täidetud ülesande (2) optimaalse lahendi $\mathbf{x}^* = \begin{bmatrix} x_1^* & x_2^* & \dots & x_n^* \end{bmatrix}^T$ korral. Tõepoolest, optimaalse lahendi $\mathbf{x}^*$ baasivälised komponendid $x_j = 0 \ \ (j \in J)$ ja seega

$$x_{n+1} = [b_k] - b_k = -\{b_k\} < 0.$$

$$x_{n+1} = [b_k] - b_k = -\{b_k\} < 0.$$

Tekkinud olukorda võime tõlgendada nii, et kitsendus (11) lõikab ülesande (2) lubatavate lahendite hulgast välja selle optimaalse baasilahendi $\mathbf{x}^*$. Järelikult kitsendus (11) on tõepoolest lõikekitsendus.

Nüüd lisame ülesande (2) optimaalsel kujul olevale simplekstabelile lõikekitsendusele vastava rea ja muutujale x_{n+1} vastava veeru.

Saadud simpelekstabel ei ole lubatav, sest vabaliikmete veerus on negatiivne element $b_{m+1} = -\{b_k\} < 0$, kuid see simplekstabel on duaalselt lubatav. Rakendades duaalset simpleksmeetodit, teisendame selle tabeli optimaalsele kujule. Kui uus optimaalne lahend ei ole täisarvuline, siis konstrueerime uue lõikekitsenduse ja kordame eespool kirjeldatud samme kuni jõuame optimaalse täisarvulise lahendini.

Juhul kui jõuame simplekstabelini, mille lõikekitsendusele vastava juhtrea ainsaks negatiivseks elemendiks on vabaliige, siis on antud kitsendus vastuoluline ja ülesandel puuduvad lubatud lahendid.