

PROGRAMARE LINIARĂ CU CONDIȚII LOGICE

DE

F. RADÓ

Comunicare prezentată de academician T. POPOVICIU în ședința din 29 iunie 1963

1. În lucrarea [1] am formulat împreună cu L. Némethi următoarea problemă :

Se consideră mulțimea vectorilor $x = (x_1, x_2, \dots, x_n)$ cu componente reale, care verifică condițiile

$$x_j \geq 0, \quad j = 1, 2, \dots, n, \quad (1)$$

$$D_i(x) = \sum_{j=1}^n d_{ij} x_j - d_i \geq 0, \quad i = 1, 2, \dots, m \quad (2)$$

$$A_k(x) = \sum_{j=1}^n a_{kj} x_j - a_k < 0 \rightarrow B_k(x) = \sum_{j=1}^n b_{kj} x_j - b_k \geq 0, \quad k=1, \dots, p. \quad (3)$$

Să se determine vectorul x , pentru care funcția

$$f(x) = \sum_{j=1}^n c_j x_j \quad (4)$$

ia valoarea minimă.

Diferite probleme de economie conduc la acest model matematic, care generalizează programarea liniară. Astfel în lucrarea [1] o problemă de minim relativă la timpul de fabricație al unui produs ce trece printr-un șir de mașini în ordine determinată a fost pusă sub forma de mai sus. De asemenea, multe probleme de ordonanțare și de lansare a fabricației conduc la același model matematic.

Problema particulară din [1] a fost rezolvată cu metode speciale utilizându-se artificii bazate pe particularitățile problemei. Nota de față conține un algoritm care rezolvă în general problema de programare liniară cu condiții logice.

Condițiile (3) pot fi puse sub formă de disjuncții :

$$A_k(x) \geq 0 \vee B_k(x) \geq 0, \quad k = 1, 2, \dots, p. \quad (3')$$

Să considerăm cele 2^p probleme de programare liniară, avînd condițiile (1), (2) și una dintre condițiile $A_k(x) \geq 0$, $B_k(x) \geq 0$ pentru fiecare $k = 1, 2, \dots, p$ și funcția de minimizat (4). O metodă de a rezolva problema propusă constă în rezolvarea tuturor celor 2^p probleme de programare liniară și a căuta pentru care din ele f are valoarea cea mai mică. Dacă însă avem multe condiții (3), metoda aceasta nu poate fi utilizată în practică.

Algoritmul prezentat mai jos se bazează de asemenea pe rezolvări succesive de probleme de programare liniară, dar numărul lor este esențial mai mic și mai are avantajul că la fiecare dintre aceste probleme condițiile se deosebesc de cele precedente numai prin adăugarea unei condiții noi. Utilizînd metoda simplex duală, tabloul care reprezintă soluția unei probleme de programare liniară, după un număr redus de transformări (de multe ori după o singură transformare), devine soluția problemei următoare.

2. Să notăm cu P problema propusă și cu P_0 problema de programare liniară, avînd condițiile (1), (2) și funcția (4). Fie $\{k_1, k_2, \dots, k_r\}$ o submulțime a mulțimii de numere naturale $\{1, 2, \dots, p\}$, iar $\alpha_1, \alpha_2, \dots, \alpha_r$ un șir format din numerele 1 și 2; notăm cu

$$P_{k_1, k_2, \dots, k_r}^{\alpha_1, \alpha_2, \dots, \alpha_r} \quad (5)$$

problema de programare liniară, compusă din funcția (4), condițiile (1), (2) și următoarele r condiții :

$$A_{k_v}(x) \geq 0, \quad \text{dacă} \quad \alpha_v = 1,$$

respectiv

$$B_{k_v}(x) \geq 0, \quad \text{dacă} \quad \alpha_v = 2.$$

Fie x' o soluție optimă a problemei (5); se notează cu

$$m_{k_1, k_2, \dots, k_r}^{\alpha_1, \alpha_2, \dots, \alpha_r} \quad (6)$$

valoarea minimă $f(x')$. Dacă problema (5) este incompatibilă, punem ∞ pentru expresia (6), iar dacă ea are o soluție improprie, punem $-\infty$ pentru (6). Vectorul x' verifică evident condițiile (3') pentru $k = k_1, k_2, \dots, k_r$; dacă el verifică și toate celelalte condiții (3'), se marchează acest fapt prin sublinierea vectorului x' scriind $\underline{x'}$. De asemenea subliniem vectorul ce reprezintă soluția improprie a problemei (5), dacă el satisface toate condițiile (3'). Se notează cu m respectiv m_0 , min f la problemele P și P_0 (punînd ∞ și $-\infty$ pentru cazul incompatibil respectiv impropriu).

Descriem un algoritm care conduce la o soluție a problemei P , dacă ea are soluție, și care indică incompatibilitatea respectiv existența soluției improprii, în caz contrar.

Construim printr-un proces iterativ trei șiruri : șirul (*) format din probleme de programare liniară de tipul (5); odată cu formarea unui termen al șirului (*) se formează și cîte un termen în celelalte două șiruri și anume în șirul (**) se pune minimul problemei din (*), iar în șirul (***) o soluție de bază optimă. Anumiți termeni din (***) sînt subliniați; subliniem de asemenea termenii corespunzători din (**).

Primul termen al șirului (*) este P_0 .

Să presupunem că am construit un număr de termeni în cele 3 șiruri. Căutăm numărul cel mai mic în (**); dacă mai mulți termeni au valoarea cea mai mică, alegem unul dintre ei și să presupunem întîi că el nu este subliniat și nu este ∞ . Fie pentru fixarea ideilor acest termen chiar (6), termenul corespunzător în șirul (*) chiar (5), iar termenul corespunzător în (***) vectorul x' . Înlocuim vectorul x' în condițiile (3'). Ținînd seamă că x' nu este subliniat, există cel puțin un indice k_{r+1} , astfel ca

$$A_{k_{r+1}}(x') < 0 \quad \text{și} \quad B_{k_{r+1}}(x') < 0; \quad (7)$$

indicele k_{r+1} poate fi ales oricum dintre cei pentru care condițiile (7) au loc, dar se obține o tehnică de calcul mai rapidă, dacă se ia pentru k_{r+1} , indicele k care face minimă expresia

$$\max [A_k(x'), B_k(x')]. \quad (8)$$

Suprimăm din șirul (*) termenul (5) și îi adăugăm termenii

$$P_{k_1, \dots, k_r, k_{r+1}}^{\alpha_1, \dots, \alpha_r, 1} \quad \text{și} \quad P_{k_1, \dots, k_r, k_{r+1}}^{\alpha_1, \dots, \alpha_r, 2}. \quad (9)$$

Suprimăm din șirurile (**) și (***) termenii care corespund la termenul suprimat din (*) și adăugăm la ele minimele celor două probleme (9), respectiv cîte o soluție de bază optimă a lor. Verificăm dacă acestea satisfac toate condițiile (3'), și în caz afirmativ subliniem termenii respectiv în șirurile (**) și (**).

Dacă este subliniat cel puțin unul dintre termenii șirului (**), care iau valoarea cea mai mică, sau dacă toți termenii șirului (**) sînt ∞ , s-a terminat construcția celor trei șiruri. În primul caz, termenul subliniat în șirul (**) este minimul problemei P și vectorul corespunzător din șirul (***) este o soluție optimă a problemei P (proprie sau improprie după cum minimul nu este sau este $-\infty$), iar în al doilea caz problema P este incompatibilă. Procesul iterativ se termină după un număr finit de pași.

3. Algoritmul descris se generalizează pentru problema de minimizare a funcției (4), variabilele fiind supuse la condițiile (1), (2) și

$$A_k^{(1)}(x) \geq 0 \vee A_k^{(2)}(x) \geq 0 \vee \dots \vee A_k^{(s_k)}(x) \geq 0, \quad k = 1, 2, \dots, p. \quad (3'')$$

De asemenea, pot figura printre restricțiile problemei un număr de inegalități liniare, legate oricum prin operațiile calculului propozițional, căci punînd o asemenea condiție sub formă de conjuncție de disjuncții, apar o mulțime de condiții (3'').

ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ С ЛОГИЧЕСКИМИ УСЛОВИЯМИ

РЕЗЮМЕ

В работе [1] сформулирована задача: Определить из множества векторов $x = (x_1, x_2, \dots, x_n)$ с действительными составляющими, удовлетворяющими условиям (1) (2) и (3), вектор, для которого функция (4) принимает минимальное значение.

Различные экономические задачи ведут к этой математической модели, обобщающей задачу линейного программирования.

В настоящей заметке дается алгоритм решения высказанной выше задачи.

PROGRAMMATION LINÉAIRE À CONDITIONS LOGIQUES

RÉSUMÉ

Dans le travail [1], on a énoncé le problème : « Soit à déterminer, dans l'ensemble de vecteurs $x = (x_1, x_2, \dots, x_n)$ à composantes réelles satisfaisant aux conditions (1), (2), (3), le vecteur pour lequel la fonction (4) prend sa valeur minimum ».

Divers problèmes d'économie conduisent à ce modèle mathématique, qui généralise le problème de programmation linéaire.

Dans le présent travail on propose un algorithme qui résout le problème énoncé ci-dessus.

BIBLIOGRAFIE

1. L. NÉMETHI, F. RADÓ, *Ein Wartezeitproblem in der Programmierung der Produktion*, Mathematica (sub tipar).