УДК 519.832

Ю. В. Андрашко (Ужгородський нац. ун-т)

ЗВЕДЕННЯ КОНКУРЕНТНОЇ ЗАДАЧІ РОЗМІЩЕННЯ ДО ПОСЛІДОВНОСТІ ОДНОКРИТЕРІАЛЬНИХ ЗАДАЧ

The competitive placement problem with complete distribution of the market where two competitive firms open facilities consecutively for delivering the clients is considered. The algorithm for reduction of the original problem to the sequence of the single-criterion boolean programming problems is suggested.

В роботі розглядається конкурентна задача розміщення з повним розподілом ринку, на якому дві фірми послідовно розміщують підприємства для обслуговування клієнтів. Наведено алгоритм зведення вихідної задачі до послідовності однокритеріальних задач булевого програмування.

В даній роботі розглядається конкурентна задача розміщення з повним розподілом ринку, на якому присутні дві фірми [1–4].

Постановка задачі. Розглядається ринок однієї послуги. Відома множина клієнтів $J = \{1, \ldots, n\}$, що мають потребу в отриманні даної послуги. Також відома множина пунктів $I = \{1, \ldots, m\}$, де можуть бути відкриті підприємства для надання клієнтам послуги. Вважається, що підприємство задовольняє всі потреби клієнта, що обслуговується на ньому. Тобто, клієнт повністю обслуговується на єдиному підприємстві. Однак на одному підприємстві може обслуговуватись декілька клієнтів. Спочатку свою підмножину підприємств $S^1 \subset I$ відкриває перша фірма (Лідер). Потім, знаючи це рішення, друга фірма (Конкурент) відкриває власні підприємства $S^1 \subset I$, таку, що $S^1 \bigcap S^2 = \emptyset$.

Кожен клієнт вибирає з множини відкритих підприємств одне. Для визначення уподобань клієнтів введемо матрицю переваг G. Якщо $g_{lj} > g_{kj}$, то j-овий клієнт віддає перевагу пункту l перед k. При цьому клієнту байдуже, кому належить підприємство, відкрите в даному пункті.

Також відома f_i – вартість відкриття підприємства в кожному з пунктів. Лідера має певну кількість коштів F^1, що можуть бути витраченні для відкриття підприємств. Аналогічно, в Конкурента є певна кількість коштів F^2. Залежно від розміщення відкритих підприємств, множина клієнтів (ринок) буде якось розподілена між Лідером і Конкурентом. Обслуговування j-ого клієнта приносить прибуток u_j. Кожна фірма прагне максимізувати свою частку ринку та, відповідно, сумарний прибуток.

Математична модель. Введемо до розгляду змінні:

$$x_i = \begin{cases} 1, & \text{якщо Лідер відкриває підприємство в пункті i;} \\ 0, & \text{якщо Лідер не відкриває підприємство в пункті i;} \end{cases} i \in I;$$

$$y_i = \begin{cases} 1, & \text{якщо Конкурент відкриває підприємство в пункті i;} \\ 0, & \text{якщо Конкурент не відкриває підприємство в пункті i;} \end{cases}, i \in I;$$

$$u_j = \begin{cases} 1, & \text{якщо клієнта обслуговує Лідер;} \\ 0, & \text{якщо клієнта обслуговує Конкурент;} \end{cases} j \in J.$$

Знайти

$$W^1 = \sum_{j\in J} w_j u_j(x, y^*) \to max, \tag{1}$$

$$\tag{2}$$

де y^* – розв'язок

$$W^2 = \sum_{j\in J} w_j(1 - u_j(x, y)) \to max. \tag{3}$$

При умовах

$$\sum_{i\in I} f_i x_i \leqslant F^1; \tag{4}$$

$$\sum_{i\in I} f_i y_i \leqslant F^2; \tag{5}$$

$$I_j(x) = \{i \in I | g_{ij} > \max_{l\in I}\{g_{lj}|x_l = 1\}\}, j \in J; \tag{6}$$

$$u_j = 1 - \max_{i\in I_j(x)} y_i; j \in J; \tag{7}$$

$$u \in B^n; x, y \in B^m. \tag{8}$$

Цільова функція задачі визначає сумарний прибуток Лідера (1). Множина допустимих розв'язків описується за допомогою задачі Конкурента (3), (5)-(8). Вектор $x_i, i \in I$ та множини $I_j(x), j \in J$ в задачі Конкурента вважаються фіксованими та відомими.

Ідея алгоритму. Набором Конкурента назвемо булевий вектор, що визначає множину пунктів, де Конкурент відкриває підприємства. Допустимим набором назвемо такий набір, що задовольняє умовам (4) та (8). Розглянемо деяку непорожню множину допустимих наборів A. Припустимо, що Кокурент може відкривати підприємства тільки згідно деякого набору із множини A.

Для кожного $y \in A$ побудуємо множину утримання $I_j(y) = \{i \in I | g_{ij} \leqslant \min_{l\in I}\{g_{lj}|x_l = 1\}\}, j \in J$. Ці множини показують пункти розміщення підприємств у яких дозволяють Лідеру зберегти клієнта j, якщо Конкурент відкриє підприємства згідно набору y.

Введемо змінні:

$$v_{jy} = \begin{cases} 1, & \text{якщо клієнта j обслуговує Лідер;} \\ 0, & \text{якщо клієнта j обслуговує Конкурент;} \end{cases} j \in J, y \in A,$$

а Конкурент відкриває підприємства згідно набору y.

Нехай W^1 – сумарний прибуток Лідера.

Для фіксованої множини A задача Лідера може бути сформульована таким чином:

Знайти

$$W^1(A) \to max \tag{9}$$

при умовах:

$$\sum_{j\in J} w_j v_{jy} \geqslant W^1(A), y \in A; \tag{10}$$

$$\sum_{i\in I_j(y)} x_i \geqslant v_{jy}, j \in J, y \in A; \tag{11}$$

$$\sum_{i\in I} f_i x_i \leqslant F^1; \tag{12}$$

$$x \in B^m. \tag{13}$$

Цільова функція $W^1(A)$ визначає сумарний прибуток Лідера. Обмеження (10) гарантує, що Конкурент вибере найкращий набір з множини A у відповідь на набір відкритих Лідером підприємств x. А обмеження (11) визначає розподіл ринку між Лідером і Конкурентом.

Очевидно, якщо множина A містить всі допустимі набори y, то задачі (9)-(13) та (1)-(8) еквівалентні.

Зауважимо, що заміна булевих змінних v_{jy} на неперервні $0 \leqslant v_{jy} \leqslant 1$ породжує еквівалентну задачу. Якщо в оптимальному розв'язку задачі з неперервними змінними якась координата v_{jy} виявилася дробової, то збільшення її до одиниці не порушує обмеження (11) і сприяє збільшенню $W^1(A)$. Таким чином, в задачі залишиться тільки одна група булевих змінних x_i. Інші змінні можна вважати неперервними.

Якщо множина A досить мала, то для кожного $y \in A$ задачу Лідера (9)-(13) можна розв'язати класичними методами [5], як задачу частково-цілочислового програмування. Розв'язок задачі Лідера для будь-якої множини A є верхньою оцінкою оптимального значення цільової функції $\overline{W^1}(A)$ вихідної задачі (1)-(8).

Припустимо, що множина A містить всі допустимі набори y. Очевидно, що багато з елементів множини A породжують несуттєві обмеження (10) та (11). Побудуємо еквівалентну задачу, знайшовши таку підмножину A, елементи якої породжують тільки суттєві обмеження. Одним із способів знаходження такої підмножини є ітеративний процес, що наведено далі.

Нехай A – деяка множина допустимих наборів Конкурента. $x(A)$ – оптимальний розв'язок задачі (9)-(13), при $y \in A$. Тоді оптимальний розв'язок задачі Конкурента (3), (5)-(8) визначає нижню оцінку значення цільової функції $\underline{W^1}(A)$ вихідної задачі (1)-(8).

Алгоритм 1. Розв'язування конкурентної задачі розміщення з повним розподілом ринку:

1) Вибрати початкову множину A_0.

2) Розв'язати задачу (9)-(13), знайти $\overline{W^1}(A_k)$ та $x(A_k)$.

3) Розв'язати задачу (3), (5)-(8), знайти $\underline{W^1}(A_k)$ та $y(A_k)$.

4) Якщо $\overline{W^1}(A_k) = \underline{W^1}(A_k)$, то $x(A_k), y(A_k)$ – розв'язок задачі (1)-(8).

5) Інакше $A_{k+1} = A_k \cup y(A_k)$, $k = k + 1$. Повернутись до кроку 2.

Теорема 1. *Алгоритм 1 скінченний і завершується знаходженням розв'язку задачі (1)-(8).*

Доведення. Множина допустимих розв'язків задачі Конкурента (3), (5)-(8) є підмножиною булевих векторів довжини m, тому вона скінченна. Доведемо, що кожна ітерація алгоритму збільшує кількість елементів множини A_k. Припустимо, що $A_{k+1} = A_k$. Це можливо тільки тоді, коли набір y вже належить цій множині, тобто $y(A_k) \in A_k$. Але тоді, враховуючи обмеження (10), $\overline{W^1}(A_k) \leqslant \underline{W^1}(A_k)$. Врахувавши, що $\underline{W^1}(A) \leqslant W^1(A) \leqslant \overline{W^1}(A)$ для $\forall A$, ми отримуємо, що $W^1(A) = \overline{W^1}(A_k) = \underline{W^1}(A_k)$. Алгоритм завершується, знайшовши розв'язок задачі (9)-(13).

З еквівалентності задач (9)-(13) та (1)-(8), випливає, що $x(A_k)$ та $y(A_k)$ – точний розв'язок конкурентної задачі розміщення з повним розподілом ринку (1)-(8).

На жаль, отримати достатньо точну оцінку числа ітерацій такого методу не вдалось.

Запропонований алгоритм тестувався на прикладах з бібліотеки тестових завдань „Дискретні моделі розміщення" [4]. Чисельні експерименти показали, що кількість ітерацій, що необхідна для знаходження розв'язку залежить від вибору початкової множини A_0. Розглядались два способи вибору початкового значення: порожня множина, та множина, що складається з розв'язків задачі (14), (15).

$$\sum_{j \in J} \max_{i \in I} g_{ij} y_i \to max; \tag{14}$$

$$\sum_{i \in I} f_i y_i \leqslant F^2. \tag{15}$$

Приклад. Розглянемо застосування алгоритму на модельній задачі [3], для

Таблиця 1.

Кількість ітерацій при різних значеннях множини A_0.

Код задачі	Порожня множина	Розв'язок задачі (14), (15)
a50-01	1285	563
a50-02	1768	622
a50-03	1650	529
a50-04	2951	871
a50-05	1877	793

якої відомий точний розв'язок $x^* = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$, $y^* = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$.

Нехай пункти відкриття підприємств співпадають із місцем розміщенням клієнтів. Розміщення пунктів і клієнтів, а також прибуток, що приносить обслуговування відповідного клієнта, задано за допомогою графа (рис. 1). Привабливість пункту для клієнта обернено пропорційна відстані до нього. Кожна фірма може відкрити не більше, ніж два підприємства.

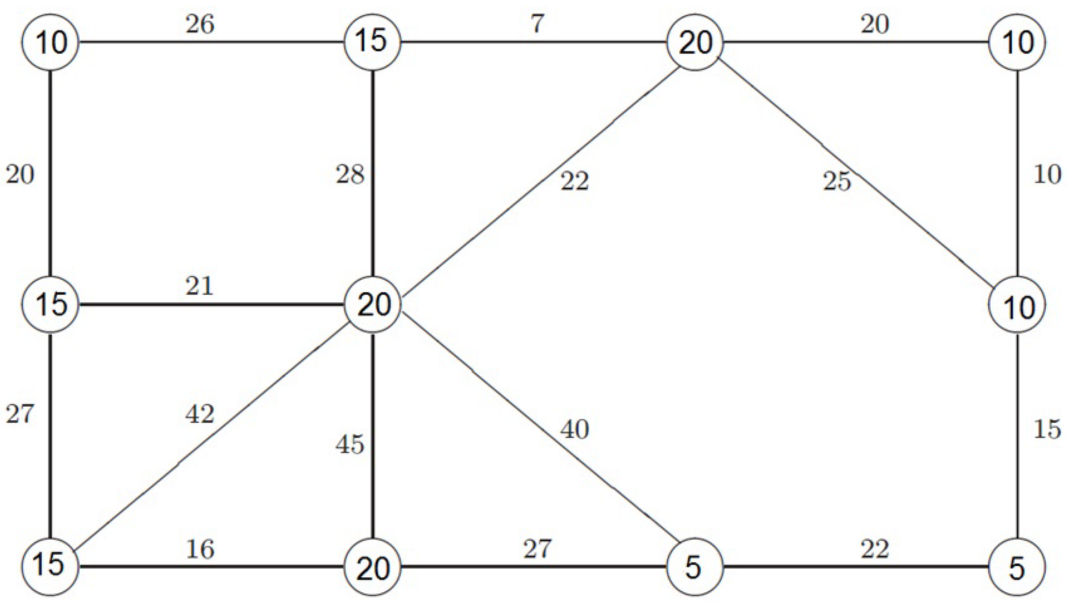

Рис. 1. Граф розміщення пунктів та клієнтів.

За домомогою алгоритму Флойда-Уоршелла знайдемо відстані між клієнтами і пунктами.

Врахуємо, що розв'язок задачі не залежить від абсолютних значеннь привабливості пунктів для клієнтів, а тільки від їх співвідношення. Для уникнення дробових і нескінченних величин будемо вважати привабливість g_{ij} рівною не $\frac{1}{d_{ij}}$, а $M - d_{ij}$, де d_{ij} – довжина найкоротшого шляху від клієнта i до пункту j, а M – деяке велике число таке, що $M \geqslant \max\limits_{i \in I, j \in J} d_{ij}$. Нехай $M = 100$, тоді

$$G = \begin{pmatrix} 100 & 74 & 67 & 47 & 80 & 59 & 42 & 53 & 37 & 19 & 27 \\ 74 & 100 & 93 & 73 & 54 & 72 & 68 & 30 & 27 & 32 & 53 \\ 67 & 93 & 100 & 80 & 57 & 78 & 75 & 36 & 33 & 38 & 60 \\ 47 & 73 & 80 & 100 & 37 & 58 & 90 & 16 & 26 & 53 & 75 \\ 80 & 54 & 57 & 37 & 100 & 79 & 32 & 73 & 52 & 39 & 17 \\ 59 & 72 & 78 & 58 & 79 & 100 & 53 & 58 & 55 & 60 & 38 \\ 42 & 68 & 75 & 90 & 32 & 53 & 100 & 20 & 36 & 63 & 85 \\ 53 & 30 & 36 & 16 & 73 & 58 & 20 & 100 & 84 & 57 & 35 \\ 37 & 27 & 33 & 26 & 52 & 55 & 36 & 84 & 100 & 73 & 51 \\ 19 & 32 & 38 & 53 & 39 & 60 & 63 & 57 & 73 & 100 & 78 \\ 27 & 53 & 60 & 75 & 17 & 38 & 85 & 35 & 51 & 78 & 100 \end{pmatrix};$$

$$w = \begin{pmatrix} 10 & 15 & 20 & 10 & 15 & 20 & 10 & 15 & 20 & 5 & 5 \end{pmatrix};$$

$$f = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix};$$

$$F^1 = F^2 = 2.$$

Для знаходження початковго значення множини наборів Конкурента побудуємо допоміжну задачу максимізації сумарної привабливості:

$$\max\{100y_1; 74y_2; 67y_3; 47y_4; 80y_5; 59y_6; 42y_7; 53y_8; 37y_9; 19y_{10}; 27y_{11}\}+$$
$$+\max\{74y_1; 100y_2; 93y_3; 73y_4; 54y_5; 72y_6; 68y_7; 30y_8; 27y_9; 32y_{10}; 53y_{11}\} + \ldots +$$
$$+\max\{27y_1; 53y_2; 60y_3; 75y_4; 17y_5; 38y_6; 85y_7; 35y_8; 51y_9; 78y_{10}; 100y_{11}\} \to max;$$
$$y_1 + y_2 + y_3 + y4 + y_5 + y_6 + y_7 + y_8 + y_9 + y_{10} + y_{11} \leqslant 2.$$

Знайдемо її розв'язок: $y = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$.

Нехай $A_0 = \{\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}\}$.

Знайдемо множини утримання: $I_1 = \{1\}; I_2 = \{1,2,3\}; I_3 = \{1,2,3,4,6,7\}$; $I_4 = \{1,2,3,4,6,7,10,11\}; I_5 = \{1,5\}; I_6 = \{1,2,3,5,6,10\}; I_7 = \{1,2,3,4,6,7,10,11\}$; $I_8 = \{8\}; I_9 = \{8,9\}; I_{10} = \{6,7,8,9,10,11\}; I_{11} = \{2,3,4,6,7,8,9,10,11\}$.

$x(A_0) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$ – розв'язок задачі Лідера:

$$W^1(A_0) \to max;$$
$$10v_1 + 15v_2 + 20v_3 + 10v_4 + 15v_5 + 20v_6 + 10v_7+$$
$$+15v_8 + 20v_9 + 5v_{10} + 5v_{11} \geqslant W^1(A_0);$$
$$x_1 \geqslant v_1;$$
$$x_1 + x_2 + x_3 \geqslant v_2;$$
$$x_1 + x_2 + x_3 + x_4 + x_6 + x_7 \geqslant v_3;$$
$$x_1 + x_2 + x_3 + x_4 + x_6 + x_7 + x_8 + x_{10} + x_{11} \geqslant v_4;$$
$$x_1 + x_5 \geqslant v_5;$$
$$x_1 + x_2 + x_3 + x_5 + x_6 + x_{10} \geqslant v_6;$$
$$x_1 + x_2 + x_3 + x_4 + x_6 + x_7 + x_8 + x_{10} + x_{11} \geqslant v_7;$$
$$x_8 \geqslant v_8;$$
$$x_8 + x_9 \geqslant v_9;$$
$$x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} \geqslant v_{10};$$
$$x_2 + x_3 + x_4 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} \geqslant v_{11};$$
$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} \leqslant 2;$$
$$x_i \in \{0,1\}, i = \overline{1,11}.$$

Тоді $\overline{W^1}(A) = 110$.

Знайдемо множини захоплення: $I_1 = \{1,2,5\}; I_2 = \{2\}; I_3 = \varnothing; I_4 = \{4,7\}$; $I_5 = \{1,5\}; I_6 = \varnothing; I_7 = \{4,7,11\}; I_8 = \{5,8,9\}; I_9 = \{8,9,10\}; I_{10} = \{7,9,10,11\}$; $I_{11} = \{4,7,10,11\}$.

$y(A_0) = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$ – розв'язок задачі Конкурента:

$$10u_1 + 15u_2 + 20u_3 + 10u_4 + 15u_5 + 20u_6 + 10u_7+$$
$$+15u_8 + 20u_9 + 5u_{10} + 5u_{11} \to min;$$
$$u_1 = 1 - \max\{y_1, y_2, y_5\};$$
$$u_2 = 1 - y_2;$$
$$u_3 = 1;$$
$$u_4 = 1 - \max\{y_4, y_7\};$$

$$u_5 = 1 - \max\{y_1, y_5\};$$
$$u_6 = 1;$$
$$u_7 = 1 - \max\{y_4, y_7, y_{11}\};$$
$$u_8 = 1 - \max\{y_5, y_8, y_9\};$$
$$u_9 = 1 - \max\{y_8, y_9, y_{10}\};$$
$$u_{10} = 1 - \max\{y_7, y_9, y_{10}, y_{11}\};$$
$$u_{11} = 1 - \max\{y_4, y_7, y_{10}, y_{11}\};$$
$$y_1 + y_2 + y_3 + y4 + y_5 + y_6 + y_7 + y_8 + y_9 + y_{10} + y_{11} \leqslant 2;$$
$$y_i \in \{0, 1\}, i = \overline{1, 11}.$$

$\underline{W^1}(A_0) = 60.$

Оскільки $\underline{W^1}(A_0) \neq \overline{W^1}(A_0)$, то продовжуємо виконання алгоритму.

$A_1 = \{\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix},$
$\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}\}.$

Розв'язавши задачу Лідера, знайдемо

$x(A_1) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$ та $\overline{W^1}(A_1) = 75.$

Розв'язавши задачу Конкурента, знайдемо

$y(A_1) = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$ та $\underline{W^1}(A_1) = 75.$

Оскільки $\underline{W^1}(A_1) = \overline{W^1}(A_1)$, то $x(A_1)$ та $y(A_1)$ – розв'язок задачі.

$x(A_1) = x^*$ та $y(A_1) = y^*$, отже задача розв'язана вірно.

Висновки. Наведений алгоритм дозволяє звести конкурентну задачу розміщення з повним розподілом ринку до послідовності однокритеріальних задач булевого програмування. Обчислювальна складність алгоритму залежить від багатьох факторів, зокрема від вибору початкової множини наборів Конкурента.

Для розв'язування отриманих задач булевого програмування невеликої розмірності можна застосовувати класичні методи, зокрема метод гілок та меж. Для задач більшої розмірності доцільно застосовувати евристичні алгоритми, що будуть наведені в наступних роботах.

1. *Santos–Penate D.R., Suarez–Vega R., Dorta–Gonzalez P.* The leader–follower location model // Networks and Spatial Economics. – 2007. – V. 7. – P. 45–61.
2. *Андрашко Ю.В., Кузка О.І.* Про деякі конкурентні задачі розміщення // Науковий вісник Ужгородського університету. Серія матем. і інформ./ Редкол.: В.В. Маринець та інші. – Ужгород: Видавництво УжНУ „Говерла“, 2013. – Вип. 24, №1. – С. 5-11.
3. *Береснев В.Л.* Верхние оценки для целевых функций дискретных задач конкурентного размещения предприятий // Дискретный анализ и исследование операций. – 2008. – Т. 15, № 4. – С. 3–24.
4. *Кочетов Ю.А.* Методы локального поиска для дискретных задач размещения. – Новосибирск: НГТУ, 2009. – 267 с.
5. *Сергиенко И.В., Каспшицкая М.Ф.* Модели и методы решения на ЭВМ комбинаторных задач оптимизации. – К.: Наукова думка, 1981. – 288 с.

Одержано 25.09.2013