ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль—декабрь 2001. Серия 2. Том 8, № 2, 42–51

УДК 519.87+519.854

ДВУХУРОВНЕВАЯ ЗАДАЧА О НАЗНАЧЕНИЯХ*)

Р. М. Ларин, А. В. Пяткин

Рассматривается двухуровневый вариант задачи о назначениях. Показывается, что эта задача NP-трудна даже в частном случае. Для ее решения предлагается метод ветвей и границ, позволяющий получать решение в общем случае (даже при неединственности решений задачи нижнего уровня).

В последние годы внимание исследователей привлекают двухуровневые задачи математического программирования. В данной статье рассматривается дискретная двухуровневая задача, близкая к задаче, поставленной и исследованной в [2].

1. Постановка задачи

Рассматривается ситуация, когда некоторый заказчик (верхний уровень) способен предложить n видов работ, которые выполняются m исполнителями (нижний уровень).

Заказчика характеризуют величины:

x_i — переменные выбора заказчика: $x_i = 1$, если работа вида i заказывается, и $x_i = 0$ в противном случае;

c_{ij} — затраты заказчика на выполнение работы вида i исполнителем j $(i \in I, j \in J, |I| = n, |J| = m)$.

Вектор x, вообще говоря, может быть любым вектором длины n, состоящим из нулей и единиц, т. е. заказчик вправе заказать любую совокупность работ или вовсе отказаться от их выполнения $(x = 0)$. Пусть $I(x) = \{i \in I \mid x_i = 1\}$ — множество заказанных работ в варианте выбора x. В рассматриваемой ниже постановке задачи считается, что заказчику выгодно заказать максимум работ, которые могут быть выполнены одновременно: $|I(x)| = m$.

*) Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проекты 99–01–00482, 99–01–00581 и 01–01–06200) и Федеральной целевой программы «Интеграция» (проект 274).

Исполнителей описывают параметры:

y_{ij} — переменные выбора исполнителей: $y_{ij} = 1$, если исполнитель j выполняет работу вида i, и $y_{ij} = 0$ в противном случае;

d_{ij} — доход исполнителя j от выполнения работы i $(i \in I, j \in J)$.

Предполагается, что каждый исполнитель выбирает только один вид работ из $I(x)$, предлагаемых заказчиком, и каждый вид работ выполняется только одним исполнителем.

Задача нижнего уровня заключается в таком назначении работ из предлагаемого варианта x, которое максимизирует суммарный доход исполнителей. Задача верхнего уровня состоит в минимизации затрат заказчика с учетом реакции исполнителей.

Ниже для единообразия рассматриваются обе задачи на минимум.

Пусть $I = \{1, \ldots, n\}$, $J = \{1, \ldots, m\}$, $x = (x_1, \ldots, x_n)$, $y = (y_{ij})_{i \in I, j \in J}$. Пусть заданы две вещественные матрицы $C = (c_{ij})_{i \in I, j \in J}$ и $D = (d_{ij})_{i \in I, j \in J}$. Для определенности считается, что $n \geqslant m$.

Рассматриваются две постановки задачи: *кооперативная* и *антикооперативная* [3]. *Двухуровневая кооперативная задача о назначении* формулируется в таком виде: найти

$$\min_{x \in X} \min_{y \in Y^*(x)} \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} \tag{1}$$

где $X = \{x \mid x_i \in \{0,1\} (i \in I), \sum\limits_{i \in I} x_i = m\}$, а $Y^*(x)$ — множество оптимальных решений задачи

$$\sum_{i \in I} \sum_{j \in J} d_{ij} y_{ij} \longrightarrow \min_{y \in Y(x)} . \tag{2}$$

Здесь $Y(x) = \{y \mid y_{ij} \in \{0,1\}, \sum\limits_{i \in I} y_{ij} = 1, \sum\limits_{j \in J} y_{ij} = x_i \ (i \in I, j \in J)\}$.

Задача (1) называется задачей *верхнего* уровня, а задача (2) — *нижнего* уровня. На нижнем уровне имеем классическую задачу о назначении. При этом верхний уровень рассчитывает на наилучший выбор нижнего уровня.

Антикооперативная задача состоит в нахождении

$$\min_{x \in X} \max_{y \in Y^*(x)} \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} \tag{3}$$

т. е. верхний уровень рассчитывает на наихудший результат.

2. NP-полнота двухуровневой задачи о назначениях

В случае $D = C$ имеем обычную задачу о назначениях с прямоугольной матрицей. Известно [6], что эта задача является полиномиально разрешимой. Докажем, что в случае, когда $D = -C$ и C — произвольная (0,1)-матрица с n строками и m столбцами, причем $n \geqslant m$, задача (1)–(2) является NP-трудной [4].

По матрице C построим двудольный граф $H = (V, U; E)$, поставив в соответствие i-й строке вершину $u_i \in U$, j-му столбцу вершину $v_j \in V$ и соединив u_i и v_j ребром тогда и только тогда, когда $c_{ij} = 1$. Тогда задача (1)–(2) при $D = C$ эквивалентна задаче поиска наибольшего паросочетания в двудольном графе H. Задача (1)–(2) при $D = -C$ также может быть сформулирована в терминах графа H. В этом случае требуется найти такое m-вершинное подмножество $U' \subset U$, что наибольшее паросочетание в индуцированном подграфе $H' = H(V, U')$ имеет минимальную мощность. Далее исследуется именно эта интерпретация двухуровневой задачи о назначениях.

Будем использовать следующие обозначения. Для произвольного графа $G = (V, E)$ через $p(G)$ обозначим мощность наибольшего паросочетания в G, а через $t(G)$ — число ациклических компонент связности в G. *Подразбиением ребра* будем называть размещение на нем дополнительной вершины, т. е. замену ребра на путь длины 2, а *подразбиением графа* G — подразбиение всех его ребер. Получающийся таким образом двудольный граф обозначим через H_G. Формально $V(H_G) = V(G)$, $U(H_G) = E(G)$ и каждому ребру $e = xy$ графа G соответствуют ребра xe и ye графа H_G. Очевидно, что операция подразбиения графа может быть осуществлена за полиномиальное время. Она сохраняет компоненты связности, число циклов в каждой компоненте и степени вершин из V. Ясно, что каждая вершина из U имеет степень 2.

Для доказательства NP-полноты задачи сформулируем ее в виде задачи распознавания. Такая формулировка двухуровневой задачи о назначении приводится ниже.

Задача 2-ЗОН. Даны двудольный граф $H = (V, U; E)$, где $|V| = m$, $|U| = n$, причем $m \leqslant n$, и положительное число $k \leqslant m$.

ВОПРОС: Существует ли такое подмножество $U' \subset U$ мощности m, что для индуцированного подграфа $H' = H(V, U')$ выполняется условие $p(H') \leqslant m - k$?

Доказательство NP-полноты этой задачи будем проводить в два этапа. Сначала сформулируем вспомогательную задачу П и докажем, что она полиномиально сводится к задаче 2-ЗОН. Затем докажем NP-полноту задачи П.

Задача П. Даны граф $G = (V, E)$ с m вершинами и $n \geqslant m$ ребрами и положительное число $k \leqslant m$.

ВОПРОС: Существует ли такое подмножество ребер $E' \subset E$ мощности m, что для графа $G' = (V, E')$ выполняется условие $t(G') \geqslant k$?

Теорема 1. *Задача* П *полиномиально сводится к задаче* 2-ЗОН.

ДОКАЗАТЕЛЬСТВО. Пусть $G = (V, E)$ — граф некоторой индивидуальной задачи П. Рассмотрим двудольный граф $H = H_G$ и для него решим задачу 2-ЗОН. По построению графа H_G выбору m вершин из доли U взаимно однозначно соответствует выбор m ребер графа G. Следовательно, имеем взаимно однозначное соответствие между индуцированными подграфами H' задачи 2-ЗОН и графами G' задачи П. Завершение доказательства теоремы 1 вытекает из следующего утверждения.

Лемма 1. *При любом* U' (*и соответственно* E') *выполняется равенство* $p(H') + t(G') = m$.

ДОКАЗАТЕЛЬСТВО. Пусть $t(G') = k$. Компоненты связности графа G' перенумеруем так, чтобы первыми оказались k деревьев. Компоненты связности обозначим через $G_1, G_2, \dots, G_k, G_{k+1}, \dots, G_l$, а число вершин в них — через $m_1, \dots, m_l$. Очевидно, что

$$\sum_{i=1}^{l} m_i = m.$$

Соответствующие компоненты связности графа H' обозначим через $H_1, \dots, H_l$. Вычислим мощность наибольшего паросочетания в H'. Для этого докажем следующие два утверждения.

1) Если $i \leqslant k$, то $p(H_i) = m_i - 1$. Действительно, так как G_i — дерево, то $|U(H_i)| = |E(G_i)| = m_i - 1$ и, значит, $p(H_i) \leqslant m_i - 1$. Построение паросочетания мощности $m_i - 1$ производится с помощью процедуры удаления висячих вершин в дереве G_i. Для висячей вершины x, инцидентной некоторому единственному ребру e, в паросочетание графа H_i добавим ребро xe и удалим из G_i вершину x и ребро e. Осуществляя эту операцию до тех пор, пока в G_i не останется ребер, получим в H_i паросочетание мощности $m_i - 1$.

2) Если $i > k$, то $p(H_i) = m_i$. Действительно, с одной стороны, $|V(H_i)| = |V(G_i)| = m_i$ и, значит, $p(H_i) \leqslant m_i$. С другой стороны, так как компонента G_i не является деревом, то в ней имеется не менее m_i ребер и по меньшей мере один цикл. Если число ребер в G_i больше m_i, то в G_i найдем цикл и удалим произвольное ребро из этого цикла. Очевидно, что это не нарушит связности графа G_i. Эту операцию будем производить до тех пор, пока число ребер в G_i не станет равным m_i, т. е. числу вершин. Такие графы называются *уницикличecкими*. Их структура такова: они состоят из единственного цикла C, к которому «приклеены»

всевозможные деревья. Вершины цикла назовем *внутренними*, а остальные вершины — *внешними*. Проведем индукцию по числу внешних вершин. Если таких вершин нет, то G_i является циклом с m_i вершинами, а значит, H_i является циклом с $2m_i$ вершинами и $p(H_i) = m_i$. Предположим, что для унициклических графов не более чем с r внешними вершинами утверждение доказано, и пусть G_i содержит $r+1$ внешних вершин. Пусть x — висячая внешняя вершина. Инцидентное ей ребро обозначим через e. Удалив из G_i вершину x и ребро e, получим унициклический граф G'_i с $m_i - 1$ вершинами, из которых r вершин будут внешними. По индукции в графе H'_i существует паросочетание M мощности $m_i - 1$. Но тогда $M \cup \{xe\}$ образует искомое паросочетание мощности m_i в H_i. Поэтому имеем

$$t(G') + p(H') = k + \sum_{i=1}^{l} p(H_i) = k + \sum_{i=1}^{k}(m_i - 1) + \sum_{i=k+1}^{l} m_i = k + m - k = m.$$

Лемма 1 доказана.

Теперь докажем NP-полноту задачи П. Для этого воспользуемся известной NP-полной задачей КЛИКА [5].

Задача КЛИКА. Даны m-вершинный граф $G = (V, E)$ и целое положительное число $k \leqslant m$.

ВОПРОС: Существует ли в G k-вершинная клика (полный подграф)?

Нам потребуется подзадача этой задачи, получающаяся из последней введением дополнительного ограничения на k.

Задача КЛИКА-1. Даны m-вершинный граф $G = (V, E)$ и целое число k такое, что $m/2 \leqslant k \leqslant m$.

ВОПРОС: Существует ли в G k-вершинная клика?

Лемма 2. *Подзадача* КЛИКА-1 *является* NP-*полной.*

Доказательство. Докажем сведение КЛИКА $\longrightarrow$ КЛИКА-1. Рассмотрим m-вершинный граф $G = (V, E)$ и положительное число $k \leqslant m$ некоторой индивидуальной задачи КЛИКА. Построим вспомогательный граф G_1, добавив к графу G еще m смежных друг с другом вершин и соединив каждую из них со всеми вершинами из V. Пусть ω — плотность (размер максимальной клики) графа G. Очевидно, что граф G_1 имеет $m_1 = 2m$ вершин, а его плотность есть $\omega_1 = \omega + m$. Положим $k_1 = k + m \geqslant m_1/2$ и применим к графу G_1 процедуру решения подзадачи КЛИКА-1. Имеем

$$\omega_1 \geqslant k_1 \Leftrightarrow \omega + m \geqslant k + m \Leftrightarrow \omega \geqslant k.$$

Таким образом, задача КЛИКА сводится к подзадаче КЛИКА-1. Лемма 2 доказана.

Для доказательства NP-полноты задачи П нам потребуется следующая

Лемма 3. *Пусть $G = (V, E)$ — m-вершинный граф с n ребрами и $k \geqslant 4$. Тогда в G есть клика размера k тогда и только тогда, когда существует такое подмножество ребер $E' \subset E$ мощности $(k^2 - k)/2$, что для графа $G' = (V, E')$ выполняется условие $t(G') = m - k$.*

Доказательство. Если в графе G есть клика размера k, то ее ребра образуют искомое множество E'.

Пусть E' — такое подмножество из $(k^2 - k)/2$ ребер, что в графе $G' = (V, E')$ ровно $m - k$ компонент являются деревьями. Суммарное число вершин в этих деревьях равно $m - k + l$, где l — общее число их ребер. Очевидно, что $0 \leqslant l \leqslant k$. Тогда граф с остальными $k - l$ вершинами имеет $(k^2 - k - 2l)/2$ ребер. Отсюда следует, что

$$\frac{(k-l)^2 - (k-l)}{2} \geqslant \frac{k^2 - k - 2l}{2},$$

т. е. $l^2 \geqslant 2kl - 3l$, и если $l > 0$, то $l \geqslant 2k - 3$. Отсюда и из условия $l \leqslant k$ следует, что $k \leqslant 3$, что противоречит условию $k \geqslant 4$. Значит, $l = 0$ и все деревья суть изолированные вершины. Но тогда граф с остальными k вершинами содержит $(k^2 - k)/2$ ребер, т. е. является кликой размера k. Лемма 3 доказана.

Теперь перейдем к доказательству NP-полноты задачи П.

Теорема 2. *Задача* П *является* NP-*полной.*

Доказательство. Предположим, что существует полиномиальный алгоритм решения задачи П. Покажем, что тогда задачу КЛИКА-1 можно решить за полиномиальное время.

Рассмотрим индивидуальную задачу КЛИКА-1, т. е. m-вершинный граф $G = (V, E)$ и целое число $k \geqslant m/2$. Положим

$$a = \frac{k^2 - k}{2} - m.$$

Легко проверить, что $a \geqslant 0$ и $k \geqslant 4$ при $m \geqslant 10$. Рассмотрим граф $G_a = (V_a, E)$, получающийся добавлением к графу G еще a изолированных вершин. Имеем

$$|V_a| = m_a = \frac{k^2 - k}{2}.$$

Положим $k_a = m_a - k$ и применим к G_a алгоритм решения задачи П, т. е. за полиномиальное время определим, существует ли в графе G_a

такое подмножество ребер мощности m_a, что k_a компонент связности в соответствующем графе являются деревьями. Но по лемме 3 ответ на этот вопрос положителен тогда и только тогда, когда в графе G_a существует клика размера k. Так как добавление изолированных вершин не увеличивает плотности графа, то мы определили, есть ли клика размера k в графе G, т. е. решили задачу КЛИКА-1. Значит, по лемме 2 задача П является NP-полной. Теорема 2 доказана.

Заметим, что при доказательстве теоремы 1 мы сводили задачу П к частному случаю задачи 2-ЗОН, когда все вершины большей доли двудольного графа H имеют степень 2. Эти графы соответствуют матрицам с двумя единицами в каждой строке. Поэтому задача (3) остается NP-трудной даже для (0,1)-матриц с двумя единицами в каждой строке.

3. Решение задачи методом ветвей и границ

Для единообразия изложения рассмотрим кооперативную постановку задачи. Алгоритм легко переносится и на случай максимизации.

Обозначим через

k — номер рассматриваемой вершины в дереве ветвлений ($0 \leqslant k \leqslant K$), где K — общее текущее число как проверенных, так и непроверенных вершин;

I_k — множество номеров строк, которые необходимо удалить из матриц C и D, чтобы получить подматрицы C_k и D_k;

$n_k = |I \backslash I_k|$ — число строк в матрицах C_k и D_k;

F_k^* — оптимальное значение целевой функции в задаче о назначении с матрицей C_k (нижняя оценка);

$I_k^* \equiv \{i_1^k, \ldots, i_m^k\}$ — соответствующий оптимальный набор строк из $I \backslash I_k$;

F_k' — гарантированное значение целевой функции верхнего уровня на выбранном множестве строк I_k^*: значение этой функции на некотором допустимом решении, определяемом на нижнем уровне. Если $Y^*(I_k^*)$ — множество оптимальных решений задачи нижнего уровня на множестве строк I_k^*, то в кооперативном случае

$$F_k' = \min_{y \in Y^*(I_k^*)} \sum_{i \in I_k^*} \sum_{j \in J} c_{ij} y_{ij};$$

V_k — множество ветвлений в вершине с номером k. В простейшем случае, который используется в описании алгоритма, $V_k = I_k^*$: на каждой из ветвей, выходящих из вершины k, удаляется одна из строк матриц C_k и D_k, вошедших в оптимальное решение I_k^*;

R — текущее значение рекорда (оценки сверху), которое хранится вместе с соответствующим допустимым решением $(i_1, \ldots, i_m)$.

Кроме того, введем вспомогательные множества:

$K_в$ — множество номеров висячих (непроверенных) вершин, т. е. вершин, которые являются перспективными при дальнейшем просмотре;

$K_з$ — множество номеров зачеркнутых (проверенных) вершин, т. е. вершин, которые уже оценены и ветвление из них не производится.

Для каждой вершины $k \in K_з$ хранится I_k, а для каждой вершины $k \in K_в$ необходимо хранить I_k, F_k^*, V_k, F_k'.

Начальная запись имеет вид $I_0 = \oslash$, $K_в = \{0\}$, $K_з = \oslash$, $K = 0$.

Алгоритм, реализующий метод ветвей и границ, состоит из последовательности следующих шагов.

Шаг 0. Находится F_0^*, $V_0 = I_0^*$, F_0'. Если $F_0^* = F_0'$, то оптимальное решение задачи получено. Иначе запоминаются $R = F_0'$ и множество строк I_0^*, соответствующее найденному решению.

Шаг 1. Выбирается непросмотренная вершина с наименьшей оценкой F_k^* . Пусть $F_k^* = \min\limits_{l \in K_в} F_l^*$. При этом имеем I_k, F_k^*, V_k, F_k'.

Шаг 2. Предварительный просмотр ветвей $V_k = \{i_{\alpha_1}^k, \ldots, i_{\alpha_v}^k\}$, где $v = |V_k|$. Обозначим через r номер рассматриваемой ветви $(1 \leqslant r \leqslant v)$. Ветви перебираются в порядке возрастания величины r, начиная с $r = 1$.

2.1. Пусть r задано. Имеем множество $I_k \cup \{i_{\alpha_r}^k\}$ для предполагаемой новой вершины (еще не имеющей номера).

Если а) существует $l \in K_з$ такое, что $I_l \subset I_k \cup \{i_{\alpha_r}^k\}$, или б) существует $l \in K_в$ такое, что $I_l = I_k \cup \{i_{\alpha_r}^k\}$, то полагается $V_k := V_k \backslash \{i_{\alpha_r}^k\}$.

2.2. Переход к следующей ветви: $r := r + 1$. Если $r \leqslant v$, то переход к п. 2.1.

Если же $r > v$ и $V_k = \oslash$, то вершина k вычеркивается, при этом полагается $K_в := K_в \backslash \{k\}$. Если встречался только случай а), то полагается $K_з := K_з \cup \{k\}$. Переход к шагу 1.

Если же $r > v$ и $V_k \neq \oslash$, то переход к шагу 3.

Шаг 3. Перебор ветвей V_k, начиная с $r = 1$. Пусть $V_k = \{i_{\beta_1}^k, \ldots, i_{\beta_{v'}}^k\}$, где $v' = |V_k|$, тогда $1 \leqslant r \leqslant v'$.

3.1. Полагается $K := K + 1$. Новой вершине присваивается номер $\bar{k} = K$. Имеем $I_{\bar{k}} = I_k \cup \{i_{\beta_r}^k\}$.

Если $|I_{\bar{k}}| = n - m$, то находится только $F_{\bar{k}}'$. Полагается $K_з := K_з \cup \{\bar{k}\}$. Если при этом $F_{\bar{k}}' < R$, то определяется новое значение рекорда $R := F_{\bar{k}}'$. Переход к шагу 4.

Если $|I_{\bar{k}}| < n - m$, то вычисляются $F_{\bar{k}}^*$ и $I_{\bar{k}}^*$.

3.2. Если $F_{\bar{k}}^* \geqslant R$, то полагается $K_з := K_з \cup \{\bar{k}\}$ и переход к п. 4.2, иначе полагается $V_{\bar{k}} = I_{\bar{k}}^*$ и вычисляется $F_{\bar{k}}'$.

3.3. Если $F_{\bar{k}}' < R$, то полагается $R := F_{\bar{k}}'$.

3.4. Если $F_{\bar{k}}' = F_{\bar{k}}^*$, то вершина $\bar{k}$ вычеркивается, т. е. полагается $K_з := K_з \cup \{\bar{k}\}$, иначе полагается $K_в := K_в \cup \{\bar{k}\}$. Переход к следующему шагу.

Шаг 4.

4.1. В случае уменьшения рекорда уточняются множества проверенных и непроверенных вершин: $K_в := K_в \backslash L$, $K_з := K_з \cup L$, где $L = \{l \in K_в \mid F^*_l \geqslant R\}$.

4.2. Полагается $r := r + 1$.

Если $r \leqslant v'$, то переход к п. 3.1, иначе полагается $K_в := K_в \backslash \{k\}$ и осуществляется переход к шагу 1 при условии, что $K_в \neq \varnothing$.

Если $K_в = \varnothing$, то процесс вычислений закончен. Оптимальное значение целевой функции равно R и $(i_1, \ldots, i_m)$ является решением.

Конец.

Отметим три момента в приведенном алгоритме, которые соответствуют классической схеме метода ветвей и границ [1].

1) Случай $F^*_k = F'_k$ соответствует ситуации, когда вершина переходит из непроверенных вершин в проверенные, так как в ней удалось найти наилучшее решение. Из этого равенства следует, что в вершине найдено оптимальное решение на множестве строк $I \backslash I_k$ и соответствующее допустимое множество дальше рассматривать не нужно.

Очевидно, что оценка F^*_k является наилучшей для верхнего уровня на множестве строк $I \backslash I_k$; при этом она достигается, если решение задачи нижнего уровня таково, что $F'_k = F^*_k$. Это зависит от вида матрицы D_k. В силу ее произвольности оценка, конечно, может и не достигаться.

2) Случай $F^*_k \geqslant R$ соответствует ситуации, когда нижняя оценка F^*_k для рассматриваемого подмножества строк $I \backslash I_k$ не меньше верхней оценки R. В этом случае вершина также удаляется из дерева ветвлений и включается в число проверенных.

3) Шаг 2 позволяет сузить множество ветвей без решения соответствующих задач о назначении. Если учесть, что в каждой непроверенной вершине решаются две задачи о назначении (на матрице C_k и на подматрице матрицы D_k), то такая операция представляется целесообразной. Она позволяет существенно уменьшить число таких задач.

Представляет интерес следующий факт, связанный с неединственностью решения задачи нижнего уровня.

Утверждение 1. *Величина F'_k может быть вычислена не более чем за $O(m^3)$ операций.*

Доказательство. Для кооперативного случая доказательство проведем путем сведения данной задачи к решению новой задачи о назначениях.

Решение задачи (2) на множестве строк I^*_k требует не более $O(m^3)$ операций. Если задача решается венгерским или аналогичным методом, то в результате получим преобразованную матрицу D^*_k, содержащую нули на местах, которые могут войти в оптимальные назначения задачи

(2). Элементы этой матрицы обозначим через d^*_{ij}. Построим вспомогательную матрицу C^*_k размерности $m \times m$ с элементами c^*_{ij} $(i \in I^*_k, j \in J)$ такую, что

1) $c^*_{ij} = c_{ij}$, если $d^*_{ij} = 0$;

2) $c^*_{ij} = +\infty$, если $d^*_{ij} > 0$.

После этого решаем задачу о назначениях с матрицей C^*_k. Решением вспомогательной задачи является величина F'_k, и для ее нахождения требуется не более $O(m^3)$ операций.

В антикооперативном случае условие 2 заменяется на условие: $c^*_{ij} = -\infty$, если $d^*_{ij} > 0$. Утверждение 1 доказано.

Отметим, что последнее обстоятельство выгодно отличает предлагаемый алгоритм от алгоритма из [7], разработанного для решения задач двухуровневого линейного булевого программирования, в котором предполагается единственность решения задачи нижнего уровня.

ЛИТЕРАТУРА

1. **Береснев В. Л., Дементьев В. Т.** Исследование операций. Новосибирск: Изд-во НГУ, 1979.
2. **Горбачевская Л. Е., Дементьев В. Т., Шамардин Ю. В.** Двухуровневая экстремальная задача выбора номенклатуры изделий. Новосибирск, 1997. 26 с. (Препринт / РАН. Сиб. отд-ние. Ин-т математики; № 41).
3. **Горбачевская Л. Е.** Алгоритмы и сложность решения двухуровневых задач стандартизации с коррекцией дохода // Дискрет. анализ и исслед. операций. Сер. 2. 1998. Т. 5, № 2. С. 20–33.
4. **Пяткин А. В.** NP-полнота двухуровневой задачи о назначениях. Новосибирск, 2000. 11 с. (Препринт / РАН. Сиб. отд-ние. Ин-т математики; № 71).
5. **Гэри М., Джонсон Д.** Вычислительные машины и трудноразрешимые задачи. М.: Мир, 1982.
6. **Кристофидес Н.** Теория графов: алгоритмический подход. М.: Мир, 1978.
7. **Bard J. F.** Practical bilevel optimization. Algorithms and applications. Dordrecht; Boston; London: Kluwer Acad. Publ., 1998. (Nonconvex Optimization and Its Applications; V. 30).

Адрес авторов:

Институт математики
им. С. Л. Соболева СО РАН,
пр. Академика Коптюга, 4,
630090 Новосибирск, Россия.
E-mail: orlab@math.nsc.ru

Статья поступила
26 июня 2000 г.,
переработанный вариант —
22 мая 2001 г.