

ЛЕКСИКОГРАФИЧЕСКИЙ 0,5–ПРИБЛИЖЕННЫЙ АЛГОРИТМ ДЛЯ ЗАДАЧИ О МНОГИХ РАНЦАХ

А. Б. Хуторецкий, С. В. Бредихин, А. А. Замятин

Предложен алгоритм для задачи о многих ранцах (МКР). Алгоритм использует упорядочение ранцев по неубыванию размера и два упорядочения объектов: по невозрастанию полезности и невозрастанию отношения полезности к размеру объекта. Эти упорядочения порождают два отношения лексикографического порядка на множестве $A \times B$ (здесь A — множество ранцев и B — множество неделимых объектов). По каждому лексикографическому упорядочению алгоритм строит допустимое решение МКР, просматривая пары $(a, b) \in A \times B$ в соответствующем порядке и помещая объект b в ранец a , если объект еще не размещен и в ранце достаточно места. Из двух полученных решений алгоритм выбирает лучшее. Алгоритм имеет относительную точность 0,5 и трудоемкость $O(mn)$ (без учета сортировок), где m и n — мощности множеств A и B соответственно.

Ключевые слова: задача о многих ранцах, лексикографическое упорядочение, приближенный алгоритм, относительная точность.

DOI 10.17377/SIBJIM.2018.21.210

ВВЕДЕНИЕ

Задача о многих ранцах (Multiple Knapsack Problem, МКР) заключается в следующем (см., например, [1, 2]): есть множество A ранцев, $|A| = m$, и множество B неделимых объектов, $|B| = n$; нужно разместить в ранцах объекты максимальной суммарной полезности. Такая задача возникает, например, при распределении вычислительного ресурса в многопроцессорной системе.

Задача NP -полна [3, теорема 15.8], вследствие чего представляют интерес приближенные алгоритмы ее решения. В [4] доказано, что если $P \neq NP$, то не существует полностью приближенной полиномиальной схемы (FPTAS) для МКР даже при $m = 2$. Там же построена приближенная полиномиальная схема (PTAS) для МКР, позволяющая построить $(1 - \varepsilon)$ -приближенное решение с трудоемкостью $O(n^{O(\ln(1/\varepsilon)/\varepsilon^8)})$ (эта оценка трудоемкости приведена в работе [5]).

Просматривая пары $(a, b) \in A \times B$ в некотором порядке и помещая объект b в ранец a , если объект еще не размещен и в ранце достаточно места, можно построить допустимое решение МКР. Различные упорядочения пар порождают семейство жадных алгоритмов для МКР; назовем их *укладки*. Определение укладки легко модифицировать применительно к задачам с делимыми объектами (как, например, линейная релаксация МКР): для каждой очередной пары (a, b) в ранец b укладывается максимальная допустимая (с учетом предшествующей загрузки) часть объекта a . Отношения полного порядка $<_A$ и $<_B$ на множествах A и B соответственно определяют лексикографическое упорядочение на $A \times B$ и, следовательно, некоторую укладку. Частным случаем МКР (при $m = 1$) является задача о ранце (Knapsack Problem, КР). Для линейной релаксации КР, предполагая, что размер любого объекта не больше размера ранца, оптимальное решение (x^{LP} в обозначениях из [2]) можно получить укладкой с упорядочением объектов по невозрастанию эффективности (отношения полезности к размеру) [6].

Если алгоритм $\mathbf{A}$, примененный к задаче P из некоторого класса $\mathcal{P}$ задач максимизации, находит допустимое решение со значением целевой функции $\mathbf{A}(P)$, то его относительная точность (worst-case performance ratio [1, с. 9]) для класса задач $\mathcal{P}$ — это наибольшее число ε такое, что $\mathbf{A}(P) \geq \varepsilon \text{Opt}(P)$ для всех $P \in \mathcal{P}$ (здесь $\text{Opt}(P)$ — оптимальное значение целевой функции в задаче P). Алгоритм с относительной точностью ε называют ε -приближенным. Известный 0, 5-приближенный алгоритм для КР [1] выбирает лучшее из двух допустимых решений. Первое включает все объекты, которые целиком входят в x^{LP} , второе помещает в ранец только объект максимальной полезности. В [7] предложено улучшение этого алгоритма: второе размещение строится укладкой с упорядочением объектов по невозрастанию полезности.

Пусть $Q(a) > 0$ и $q(b) > 0$ — размер ранца a и объекта b соответственно. В книге [2, с. 299] рассматривается линейная релаксация МКР с дополнительным условием: нельзя поместить в ранец a ненулевую часть объекта b , если $Q(a) < q(b)$. Там же утверждается (без доказательства), что если упорядочить объекты по невозрастанию эффективности, а ранцы — по неубыванию размера, то укладка, использующая только пары (a, b) , для которых $q(b) \leq Q(a)$, дает оптимальное решение x^* этой задачи. Мы докажем (теорема 1) более общий результат: если объекты упорядочены по невозрастанию эффективности и множество допустимых пар согласовано с упорядочением ранцев, то укладка дает оптимальное решение линейной релаксации МКР. Объекты, вошедшие в размещение x^* целиком, и объекты, вошедшие в него частично, образуют два допустимых решения МКР: x_1 и x_2 . Алгоритм, выбирающий лучшее из этих решений, является 0, 5-приближенным [2, теорема 10.4.2]. Трудоемкость алгоритма есть $O(mn)$ (без учета сортировок).

Цель статьи — построение нового алгоритма с такими же оценками точности и трудоемкости. Мотивация состоит в том, что для конкретной задачи о многих ранцах разные алгоритмы дают, как правило, разные решения, из которых можно выбрать лучшее.

Мы рассмотрим для МКР две укладки с упорядочением ранцев по неубыванию размера. Первая упорядочивает объекты по невозрастанию эффективности, вторая — по невозрастанию полезности. Эти укладки порождают размещения x_3 и x_4 , которые в общем случае отличаются от размещений x_1 и x_2 (для КР размещения x_1 и x_3 совпадают). Следствие 5 (ниже) доказывает, что комбинированный алгоритм, выбирающий лучшее из размещений x_3 и x_4 , является 0, 5-приближенным. Таким образом, предложен новый 0, 5-приближенный алгоритм для МКР с трудоемкостью $O(mn)$ (без учета сортировок), который сочетает идеи указанных выше работ и обобщает на случай $m > 1$ алгоритм из [7]. Алгоритм для МКР, имеющий лучшие характеристики, нам неизвестен.

1. ФОРМАЛИЗАЦИЯ ЗАДАЧИ

Координаты векторов x, y, y_0, y_k, z из $\mathbb{R}^{mn}$, сопоставленные парам $(a, b) \in A \times B$, будем обозначать $x(a, b), y(a, b), y_0(a, b), y_k(a, b), z(a, b)$ соответственно. Будем рассматривать МКР в следующей записи:

$$V(x) = \sum_{b \in B} v(b) \sum_{a \in A} x(a, b) \rightarrow \max \quad (1)$$

при условиях

$$\sum_{b \in B} q(b)x(a, b) \leq Q(a) \quad \text{для всех } a, \quad (2)$$

$$\sum_{a \in A} x(a, b) \leq 1 \quad \text{для всех } b, \quad (3)$$

$$x(a, b) \in \{0, 1\} \quad \text{для всех } a, b. \quad (4)$$

Здесь $v(b) > 0$ — полезность объекта b , а переменная $x(a, b)$ — «доля» объекта b , помещенная в ранец a . Пусть $K(P)$ — задача (1)–(4) с набором параметров $P = \langle A, B, q(\cdot), v(\cdot), Q(\cdot) \rangle$; $X(P)$ — множество всех допустимых решений этой задачи. Вектор $x \in X(P)$ указывает допустимое размещение: объект b «уложен» в ранец a , если и только если $x(a, b) = 1$.

2. ЛЕКСИКОГРАФИЧЕСКИЕ АЛГОРИТМЫ

Для конечного множества $M \neq \emptyset$, на котором задано отношение полного порядка $\prec$, обозначим $\min(\prec, M)$ первый (в смысле отношения $\prec$) элемент из M . Опишем два лексикографических алгоритма (GI и TGI) для МКР.

2.1. Алгоритм GI (Greedy heuristic for Indivisible objects).

вход: задача $K(P)$, отношение полного порядка $\prec$ на множестве $A \times B$;

выход: вектор $x \in X(P)$;

begin {просмотр пар $(a, b) \in A \times B$ в соответствии с порядком $\prec$ }

(а) $Q'(a) := Q(a)$ для всех a ; $q'(b) := q(b)$ для всех b ; $M := \emptyset$; {инициализация; $Q'(a)$ — неиспользованный размер ранца a ; $q'(b)$ — неразмещенная часть объекта b ; M — множество пар (a, b) , для которых значение $x(a, b)$ определено}

(б) **while** $M \neq A \times B$ **do** {построение вектора x }

найти $(a_0, b_0) = \min(\prec, (A \times B) \setminus M)$; {начало шага (a_0, b_0) }

if $q'(b_0) = q(b_0)$ **and** $q(b_0) \leq Q'(a_0)$ {объект b_0 еще не размещен и может быть уложен в ранец a_0 }

(с) **then** $x(a_0, b_0) := 1$ {объект b_0 уложен в ранец a_0 } **else** $x(a_0, b_0) := 0$

endif;

$Q'(a_0) := Q'(a_0) - x(a_0, b_0)q(b_0)$; $q'(b_0) := q'(b_0) - x(a_0, b_0)q(b_0)$;

{значение $x(a_0, b_0)$ определено, конец шага (a_0, b_0) }

endwhile

end.

Ясно, что в начале шага (a_0, b_0) алгоритма имеем

$$Q'(a_0) = Q(a_0) - \sum_{(a_0, b) \prec (a_0, b_0)} q(b)x(a_0, b); \quad (5)$$

$$q'(b_0) = q(b_0) \left[1 - \sum_{(a, b_0) \prec (a_0, b_0)} x(a, b_0) \right]. \quad (6)$$

Для $a \in A$ положим $B(a) = \{b \in B \mid q(b) \leq Q(a)\}$ — множество всех объектов, допускающих размещение в ранце a . Используя идею из [7, с. 629], сформулируем «усеченный» вариант алгоритма GI, допускающий размещение объекта b_0 в ранце a_0 только если все объекты $b \in B(a_0)$ такие, что $(a_0, b) \prec (a_0, b_0)$, размещены в ранцах $a \preccurlyeq a_0$.

2.2. Алгоритм TGI (Truncated Greedy heuristic for Indivisible objects).

вход: задача $K(P)$, отношение полного порядка $\prec$ на множестве $A \times B$;

выход: вектор $y \in X(P)$;

begin {просмотр пар $(a, b) \in A \times B$ в соответствии с порядком $\prec$ }

(а) $Q'(a) := Q(a)$ для всех a ; $q'(b) := q(b)$ для всех b ; $A' := A$; $M := \emptyset$; {инициализация; $Q'(a)$ — неиспользованный размер ранца a ; $q'(b)$ — неразмещенная часть объекта b ; A' — множество ранцев, которые можно пополють; M — множество пар, для которых определено значение $y(a, b)$ }

(б) **while** $M \neq A \times B$ **do** {построение вектора y }

найти $(a_0, b_0) = \min(\prec, (A \times B) \setminus M)$; {начало шага (a_0, b_0) }

if $(a_0 \notin A' \text{ or } b_0 \notin B(a_0) \text{ or } q'(b_0) = 0)$

```

(c)   then  $y(a_0, b_0) := 0$  else
      if  $q(b_0) \leq Q'(a_0)$  {объект  $b_0$  может быть уложен в ранец  $a_0$ }
(d)   then  $y(a_0, b_0) := 1$  {помещаем объект  $b_0$  в ранец  $a_0$ }
(e)   else begin  $y(a_0, b_0) := 0$ ;  $A' := A' \setminus \{a_0\}$  end {ранец  $a_0$  больше не будет
      пополняться}
      endif
    endif
   $M := M \cup \{(a_0, b_0)\}$ ; {значение  $y(a_0, b_0)$  определено}
   $Q'(a_0) := Q'(a_0) - y(a_0, b_0)q(b_0)$ ;  $q'(b_0) := q'(b_0) - q(b_0)y(a_0, b_0)$ ; {конец шага
   $(a_0, b_0)$ }
endwhile
end.

```

Ясно, что в начале шага (a_0, b_0) алгоритма TGI выполняются равенства (5), (6) при $x = y$.

2.3. Свойства алгоритмов GI и TGI. Пусть $GI(P, \prec)$ и $TGI(P, \prec)$ — реализации алгоритмов GI и TGI соответственно для задачи $K(P)$, использующие упорядочение $\prec$ множества $A \times B$. Допустимые решения задачи $K(P)$, построенные алгоритмами $GI(P, \prec)$ и $TGI(P, \prec)$, обозначим $x(P, \prec)$ и $y(P, \prec)$ соответственно. Для $x \in X(P)$ и $(a_0, b) \in A \times B$ положим

$$r(x, a_0, b) = \sum_{(a, b) \preceq (a_0, b)} x(a, b).$$

Из (3) следует, что $r(x, a_0, b) = 1$, если при размещении x объект b попадает в ранец a , для которого $(a, b) \preceq (a_0, b)$, иначе $r(x, a_0, b) = 0$.

Лемма 1. Пусть $y = y(P, \prec)$.

(i) $a_0 \notin A'$ на шаге (a_0, b_0) алгоритма $TGI(P, \prec)$, если и только если существует объект b_1 такой, что $(a_0, b_1) \prec (a_0, b_0)$, $b_1 \in B(a_0)$ и $r(y, a_0, b_1) = 0$.

(ii) Если $a_0 \notin A'$ на шаге (a_0, b_0) алгоритма $TGI(P, \prec)$, то существует объект b такой, что $y(a_0, b) = 1$.

(iii) $V(x(P, \prec)) \geq V(y(P, \prec))$.

ДОКАЗАТЕЛЬСТВО. Если на шаге (a_0, b_0) алгоритма $TGI(P, \prec)$ имеем $a_0 \notin A'$, то на некотором предшествующем шаге (a_0, b_1) алгоритм выполнил строку с меткой (е), $b_1 \in B(a_0)$ и $r(y, a_0, b_1) = 0$. Обратно, пусть $(a_0, b_1) \prec (a_0, b_0)$, $b_1 \in B(a_0)$ и $r(y, a_0, b_1) = 0$. Тогда $q'(b_1) = q(b_1) > 0$ в начале шаге (a_0, b_1) и $y(a_0, b_1) = 0$. Фиксация $y(a_0, b_1) = 0$ может произойти только на метках (с) и (е) алгоритма TGI, причем $a_0 \notin A'$ на шаге (a_0, b_1) в первом случае и после этого шага — во втором случае. В любом случае $a_0 \notin A'$ на шаге (a_0, b_0) . Утверждение (i) доказано.

Если $a_0 \notin A'$ на шаге (a_0, b_0) , то на некотором шаге $(a_0, b) \prec (a_0, b_0)$ алгоритм TGI вышел на метку (е). Тогда $q(b) \leq Q(a_0)$, $q'(b) = q(b)$ и $q(b) > Q'(a_0)$, откуда $Q'(a_0) < Q(a_0)$. Теперь (ii) следует из (5).

Пусть $x = x(P, \prec)$. Допустим, что $r(x, a, b) \geq y(a, b)$ для всех пар $(a, b) \prec (a_0, b_0)$ (предположение индукции). Это верно и для пары (a_0, b_0) при $y(a_0, b_0) = 0$. Пусть $y(a_0, b_0) = 1$. Тогда $b_0 \in B(a_0)$ и $a_0 \in A'$ на шаге (a_0, b_0) алгоритма $TGI(P, \prec)$. Покажем, что $r(x, a_0, b_0) = 1$. Положим $B'(a_0) = \{b \in B(a_0) \mid (a_0, b) \prec (a_0, b_0)\}$ и выберем $b \in B'(a_0)$. Из утверждения (i) следует, что $r(y, a_0, b) = 1$. Если $y(a_0, b) = 0$, то существует a такое, что $(a, b) \prec (a_0, b)$ и $y(a, b) = 1$. Тогда $r(x, a, b) = 1$ по предположению индукции и $x(a_0, b) = 0$ по (3). Следовательно, $y(a_0, b) \geq x(a_0, b)$ для всех $b \in B'(a_0)$, откуда

$$\sum_{(a_0, b) \prec (a_0, b_0)} x(a_0, b)q(b) \leq \sum_{(a_0, b) \prec (a_0, b_0)} y(a_0, b)q(b) \leq Q(a_0) - q(b_0). \quad (7)$$

Последнее неравенство в (7) следует из $y \in X(P)$ и $y(a_0, b_0) = 1$. Если найдется пара $(a, b_0) \prec (a_0, b_0)$, для которой $x(a, b_0) = 1$, то $r(x, a_0, b_0) = 1$. Пусть $x(a, b_0) = 0$ для всех $(a, b_0) \prec (a_0, b_0)$. Тогда на шаге (a_0, b_0) алгоритма GI($P, \prec$) имеем $q'(b_0) = q(b_0)$, а из (5) и (7) следует, что $q(b_0) \leq Q'(a_0)$. Значит, алгоритм GI на шаге (a_0, b_0) выйдет на метку (с) и установит $x(a_0, b_0) = 1$, откуда $r(x, a_0, b_0) = 1 \geq y(a, b)$. По индукции $r(x, a, b) \geq y(a, b)$ для всех a и b , откуда следует утверждение (iii). $\square$

Следствие 1. Пусть $y = y(P, \prec)$. Если $y(a_0, b_0) = 1$, то $r(y, a_0, b) = 1$ для всех b таких, что $(a_0, b) \prec (a_0, b_0)$ и $b \in B(a_0)$.

ДОКАЗАТЕЛЬСТВО. Пусть

$$y(a_0, b_0) = 1, \quad b \in B(a_0), \quad (a_0, b) \prec (a_0, b_0), \quad r(y, a_0, b) = 0.$$

Тогда $a_0 \notin A'$ на шаге (a_0, b_0) алгоритма TGI по утверждению (i) леммы 1, алгоритм выйдет на метку (с) и установит $y(a_0, b_0) = 0$; противоречие. $\square$

3. МКР с ДЕЛИМЫМИ ОБЪЕКТАМИ

Линейную релаксацию задачи (1)–(4) получим, заменив (4) на

$$x(a, b) \geq 0 \text{ для всех } a, b. \quad (8)$$

Из (3) следует $x(a, b) \leq 1$. Условие (8) означает, что в ранец a может быть уложена доля $x(a, b)$ объекта b , имеющая размер $x(a, b)q(b)$ и стоимость $x(a, b)v(b)$. В случае одного ранца оптимальное решение задачи (1)–(3), (8) строится укладкой, использующей упорядочение объектов по эффективности [6].

Предположим, что некоторые назначения запрещены: указано множество $C \subseteq A \times B$ такое, что

$$x(a, b) = 0, \text{ если } (a, b) \notin C. \quad (9)$$

Набор параметров P и множество C определяют задачу (1)–(3), (8), (9); обозначим ее $L(P, C)$. Пусть $Z(P, C)$ и $Z^*(P, C)$ — множества всех допустимых и соответственно всех оптимальных решений задачи $L(P, C)$. Для $x \in Z(P, C)$ и $(a_0, b_0) \in A \times B$ положим $S(x, a_0) = \sum_b q(b)x(a, b)$, $R(x, b_0) = \sum_a x(a, b)$.

Лемма 2. Если $x \in Z^*(P, C)$ и $(a_0, b_0) \in C$, то либо $S(x, a_0) = Q(a_0)$, либо $R(x, b_0) = 1$.

ДОКАЗАТЕЛЬСТВО. Для $(a_0, b_0) \in C$ и $x \in Z^*(P, C)$ предположим противное: $S(x, a_0) = Q(a_0) - \delta_1$ и $R(x, b_0) = 1 - \delta_2$, причем $\delta_1 > 0$ и $\delta_2 > 0$. Положим $\delta = \min\{\delta_1/q(b_0), \delta_2\} > 0$. Определим $y \in \mathbb{R}^{mn}$ следующим образом: $y(a_0, b_0) = x(a_0, b_0) + \delta$ и $y(a, b) = x(a, b)$ для всех $(a, b) \neq (a_0, b_0)$. Легко проверить, что $y \in Z(P, C)$, причем $V(y) = V(x) + \delta v(b_0) > V(x)$, что противоречит выбору x . $\square$

Сформулируем лексикографический алгоритм для задачи $L(P, C)$.

Алгоритм GD (Greedy heuristic for Divisible objects).

вход: задача $L(P, C)$, отношение полного порядка $\prec$ на $A \times B$;

выход: вектор $z \in Z(P, C)$;

begin {алгоритм просматривает пары $(a, b) \in A \times B$ в соответствии с упорядочением $\prec$ }

(а) $Q'(a) := Q(a)$ для всех a ; $q'(b) := q(b)$ для всех b ; $M := \emptyset$; {инициализация; $Q'(a)$ — неиспользованный размер ранца a ; $q'(b)$ — размер неразмещенной части объекта b ; M — множество пар (a, b) , для которых значение $z(a, b)$ определено}

(б) **while** $M \neq A \times B$ **do** {построение вектора z }

найти $(a_0, b_0) = \min(\prec, A \times B) \setminus M$; {начало шага (a_0, b_0) }
if $(a_0, b_0) \in C$
 (с) **then** $z(a_0, b_0) := \min\{q'(b_0), Q'(a_0)\}/q(b_0)$; {из неразмещенной части
 объекта b_0 помещаем в ранец a_0 максимум того, что в него еще
 можно уложить}
else $z(a_0, b_0) := 0$
endif
 $M := M \setminus \{(a_0, b_0)\}$; {значение $z(a_0, b_0)$ определено}
 $Q'(a_0) := Q'(a_0) - q(b_0)z(a_0, b_0)$; $q'(b_0) := q'(b_0) - q(b_0)z(a_0, b_0)$; {конец шага
 (a_0, b_0) }
endwhile
end.

Из описания алгоритма GD ясно, что в начале шага (a_0, b_0) выполнены равенства (5), (6) при $x = z$. Пусть $\text{GD}(P, C, \prec)$ — реализация алгоритма GD для задачи $L(P, C)$, использующая упорядочение $\prec$ множества $A \times B$; $z(P, C, \prec)$ — допустимое решение задачи, построенное алгоритмом $\text{GD}(P, C, \prec)$.

Лемма 3. Пусть $z = z(P, C, \prec)$ и $(a_0, b_0) \in A \times B$. Если $y \in Z(P, C)$ и $z(a, b) = y(a, b)$ для всех $(a, b) \prec (a_0, b_0)$, то $z(a_0, b_0) \geq y(a_0, b_0)$.

ДОКАЗАТЕЛЬСТВО. Пусть $z(a, b) = y(a, b)$ для всех $(a, b) \prec (a_0, b_0)$ и $y(a_0, b_0) > z(a_0, b_0) \geq 0$. Тогда $(a_0, b_0) \in C$. Из (5), (6) и $y \in Z(P, C)$ следует, что в начале шага (a_0, b_0) алгоритма $\text{GD}(P, C, \prec)$ имеем

$$\begin{aligned}
 Q'(a_0) &= Q(a_0) - \sum_{(a_0, b) \prec (a_0, b_0)} q(b)z(a_0, b) \\
 &= Q(a_0) - \sum_{(a_0, b) \prec (a_0, b_0)} q(b)y(a_0, b) \geq q(b_0)y(a_0, b_0) > q(b_0)z(a_0, b_0); \\
 q'(b_0) &= q(b_0)[1 - \sum_{(a, b_0) \prec (a_0, b_0)} z(a, b_0)] \\
 &= q(b_0)[1 - \sum_{(a, b_0) \prec (a_0, b_0)} y(a, b_0)] \geq q(b_0)y(a_0, b_0) > q(b_0)z(a_0, b_0).
 \end{aligned}$$

Но по построению $q(b_0)z(a_0, b_0) = \min\{q'(b_0), Q'(a_0)\}$; противоречие. $\square$

Далее будем считать, что множества A и B упорядочены отношениями $\prec_A$ и $\prec_B$ соответственно. Пусть $\prec_{A, B}$ — лексикографическое упорядочение множества $A \times B$, порожденное отношениями $\prec_A$ и $\prec_B$. Для $b \in B$ положим $d(b) = v(b)/q(b)$ (эффективность объекта b). Отношение $\prec_B$ назовем *d-упорядочением*, если оно упорядочивает объекты по невозрастанию эффективности: $b_1 \prec_B b_2$ влечет $d(b_1) \geq d(b_2)$ для любых b_1, b_2 . Множество $C \subseteq A \times B$ согласовано с отношением полного порядка $\prec_A$ на A , если для любых a_1, a_2, b из $(a_1, b) \in C$ и $a_1 \prec_A a_2$ следует $(a_2, b) \in C$.

Ниже, в теореме 1, сформулированы достаточные условия оптимальности вектора $z(P, C, \prec)$ в задаче $L(P, C)$. При $(a_0, b_0) \in A \times B$ и $z \in Z(P, C)$ будем использовать следующие обозначения:

$$\begin{aligned}
 N_1(z, a_0, b_0) &= |\{b \in B \mid z(a_0, b) > 0 \text{ и } b_0 \prec_B b\}|, \\
 N_2(z, a_0, b_0) &= |\{a \in A \mid z(a, b_0) > 0 \text{ и } a_0 \prec_A a\}|, \\
 N_3(z, a_0, b_0) &= |\{a \in A \mid a \preceq_A a_0 \text{ и } S(z, a) < Q(a)\}|, \\
 N_4(z, a_0, b_0) &= |\{b \in B \mid b \preceq_B b_0 \text{ и } R(z, b) < 1\}|,
 \end{aligned}$$

$$N(z, a_0, b_0) = \sum_{k=1}^4 N_k(z, a_0, b_0).$$

Лемма 4. Допустим, что отношение $\prec_B$ является d -упорядочением и множество C согласовано с отношением $\prec_A$. Положим $x = z(P, C, \prec_{A,B})$. Если $y \in Z^*(P, C)$ и $x(a, b) = y(a, b)$ для всех $(a, b) \prec_{A,B} (a_0, b_0)$, то существует вектор $z \in Z^*(P, C)$ такой, что из $(a, b) \prec_{A,B} (a_0, b_0)$ следует $z(a, b) = x(a, b)$, и либо $z(a_0, b_0) = x(a_0, b_0)$, либо $N(z, a_0, b_0) < N(y, a_0, b_0)$.

ДОКАЗАТЕЛЬСТВО. Допустим, что условия леммы выполнены. Если $x(a_0, b_0) = y(a_0, b_0)$, то утверждение леммы выполнено при $z = y$, поэтому предположим, что $x(a_0, b_0) \neq y(a_0, b_0)$. Тогда $(a_0, b_0) \in C$ (так как $x(a_0, b_0) = y(a_0, b_0) = 0$ при $(a_0, b_0) \notin C$) и $x(a_0, b_0) > y(a_0, b_0)$ по лемме 3. Итак, мы предполагаем, что

$$y(a_0, b_0) < x(a_0, b_0) \text{ и } y(a_0, b) = x(a_0, b) \text{ при } b \prec_B b_0. \quad (10)$$

Положим $\delta_1 = x(a_0, b_0) - y(a_0, b_0) > 0$. Из леммы 2 следует, что $S(y, a_0) = Q(a_0)$ или $R(y, b_0) = 1$. Рассмотрим следующие случаи.

1. $S(y, a_0) = Q(a_0)$, $R(y, b_0) = 1 - \delta_2$, где $\delta_2 > 0$. Тогда с учетом (2) $S(x, a_0) \leq S(y, a_0)$, и из (10) следует, что найдется объект b_1 , удовлетворяющий условиям $b_0 \prec_B b_1$ и $y(a_0, b_1) > x(a_0, b_1)$. Положим $\delta_3 = y(a_0, b_1)q(b_1)/q(b_0) > 0$ и $\delta = \min\{\delta_1, \delta_2, \delta_3\} > 0$. Определим вектор z : $z(a, b) = y(a, b)$, если $(a, b) \notin \{(a_0, b_0), (a_0, b_1)\}$, $z(a_0, b_0) = y(a_0, b_0) + \delta$, $z(a_0, b_1) = y(a_0, b_1) - \delta q(b_0)/q(b_1)$. Нетрудно проверить, что $z \in Z(P, C)$. Но $V(z) = V(y) + \delta[v(b_0) - v(b_1)q(b_0)/q(b_1)] \geq V(y)$, так как $b_0 \prec_B b_1$ влечет $v(b_0)/q(b_0) \geq v(b_1)/q(b_1)$, следовательно, $z \in Z^*(P, C)$. При этом $z(a, b) = x(a, b)$ для $(a, b) \prec_{A,B} (a_0, b_0)$, $N_k(z, a_0, b_0) \leq N_k(y, a_0, b_0)$ для $k \in \{1, 4\}$ и $N_k(z, a_0, b_0) = N_k(y, a_0, b_0)$ для $k \in \{2, 3\}$. Если $\delta = \delta_1$, то $z(a_0, b_0) = x(a_0, b_0)$. Если $\delta = \delta_2$, то $N_4(z, a_0, b_0) = N_4(y, a_0, b_0) - 1$, так как $R(z, b_0) = 1$. Если $\delta = \delta_3$, то $z(a_0, b_1) = 0$ и $N_1(z, a_0, b_0) = N_1(y, a_0, b_0) - 1$. Следовательно, $z(a_0, b_0) = x(a_0, b_0)$ или $N(z, a_0, b_0) < N(y, a_0, b_0)$.

2. $S(y, a_0) = Q(a_0) - \delta_2$, где $\delta_2 > 0$, и $R(y, b_0) = 1$. С учетом (3) $R(x, b_0) \leq R(y, b_0)$, и из (10) следует, что найдется a_1 , для которого $a_0 \prec_A a_1$ и $y(a_1, b_0) > x(a_1, b_0)$. Пусть $\delta = \min\{\delta_2/q(b_0), \delta_1, y(a_1, b_0)\} > 0$. Определим вектор z : $z(a, b) = y(a, b)$, если $(a, b) \notin \{(a_0, b_0), (a_1, b_0)\}$; $z(a_0, b_0) = y(a_0, b_0) + \delta$; $z(a_1, b_0) = y(a_1, b_0) - \delta$. Легко проверить, что $z \in Z(P, C)$ и $V(z) = V(y)$, откуда $z \in Z^*(P, C)$. При этом $z(a, b) = x(a, b)$ для $(a, b) \prec_{A,B} (a_0, b_0)$, $N_k(z, a_0, b_0) = N_k(y, a_0, b_0)$ для $k \in \{1, 4\}$ и $N_k(z, a_0, b_0) \leq N_k(y, a_0, b_0)$ для $k \in \{2, 3\}$. В случае $\delta = \delta_2/q(b_0)$ имеем $S(z, a_0) = Q(a_0)$, откуда $N_3(z, a_0, b_0) = N_3(y, a_0, b_0) - 1$. Если $\delta = \delta_1$, то $z(a_0, b_0) = x(a_0, b_0)$. Если же $\delta = y(a_1, b_0)$, то $z(a_1, b_0) = 0$, откуда $N_2(z, a_0, b_0) = N_2(y, a_0, b_0) - 1$. Следовательно, $z(a_0, b_0) = x(a_0, b_0)$ или $N(z, a_0, b_0) < N(y, a_0, b_0)$.

3. $S(y, a_0) = Q(a_0)$ и $R(y, b_0) = 1$. В этом случае $S(y, a_0) \geq S(x, a_0)$, $R(y, b_0) \geq R(x, b_0)$. Ввиду (10) найдутся a_1 и b_1 такие, что $a_0 \prec_A a_1$, $b_0 \prec_B b_1$, $y(a_0, b_1) > x(a_0, b_1) \geq 0$ и $y(a_1, b_0) > x(a_1, b_0) \geq 0$. Заметим, что $(a_1, b_1) \in C$, так как множество C согласовано с отношением $\prec_A$, $a_0 \prec_A a_1$ и $(a_0, b_1) \in C$ (поскольку $y(a_0, b_1) > 0$). Выберем $\delta = \min\{x(a_0, b_0) - y(a_0, b_0), y(a_1, b_0), y(a_0, b_1)\} > 0$ и определим вектор z : $z(a_0, b_0) = y(a_0, b_0) + \delta$; $z(a_1, b_0) = y(a_1, b_0) - \delta$; $z(a_0, b_1) = y(a_0, b_1) - \delta$; $z(a_1, b_1) = y(a_1, b_1) + \delta$; если $(a, b) \notin \{(a_0, b_0), (a_1, b_0), (a_0, b_1), (a_1, b_1)\}$, то $z(a, b) = y(a, b)$. Легко видеть, что $z \in Z(P, C)$ и $V(z) = V(y)$, откуда $z \in Z^*(P, C)$. Кроме того, $z(a, b) = x(a, b)$ при $(a, b) \prec_{A,B} (a_0, b_0)$, $N_k(z, a_0, b_0) = N_k(y, a_0, b_0)$ при $k \in \{3, 4\}$ и $N_k(z, a_0, b_0) \leq N_k(y, a_0, b_0)$ при $k \in \{1, 2\}$. Если $\delta = x(a_0, b_0) - y(a_0, b_0)$, то $z(a_0, b_0) = x(a_0, b_0)$. Если $\delta = y(a_1, b_0)$, то $z(a_1, b_0) = 0$, откуда $N_2(z, a_0, b_0) < N_2(y, a_0, b_0)$. Если $\delta = y(a_0, b_1)$, то $z(a_0, b_1) = 0$ и $N_1(z, a_0, b_0) < N_1(y, a_0, b_0)$. И в этом случае $z(a_0, b_0) = x(a_0, b_0)$ или $N(z, a_0, b_0) < N(y, a_0, b_0)$. $\square$

Теорема 1. Если множество C согласовано с отношением $\prec_A$ и на B задано d -упорядочение $\prec_B$, то алгоритм $\text{GD}(P, C, \prec_{A,B})$ строит оптимальное решение задачи $L(P, C)$.

ДОКАЗАТЕЛЬСТВО. Положим $x = z(P, C, \prec_{A,B})$. Если $y \in Z^*(P, C)$, $y \neq x$ и (a_0, b_0) — первая (в упорядочении $\prec_{A,B}$) пара (a, b) , для которой $x(a, b) \neq y(a, b)$, то обозначим $\varphi(x, y)$ вектор z , существование которого утверждает лемма 4. Выберем произвольно $y_1 \in Z^*(P, C)$ и построим последовательность $Y = (y_1, y_2, \dots)$ векторов из $Z^*(P, C)$ следующим образом: если вектор y_k определен и $y_k = x$, то построение закончено, иначе $y_{k+1} = \varphi(x, y_k)$. Допустим, что последовательность Y бесконечна. Тогда $y_k \neq x$ для любого $k \geq 1$. Пусть (a_k, b_k) — первая в упорядочении $\prec_{A,B}$ пара (a, b) такая, что $y_k(a, b) \neq x(a, b)$. По лемме 4 либо $(a_{k+1}, b_{k+1}) = (a_k, b_k)$ и $N(y_{k+1}, a_{k+1}, b_{k+1}) < N(y_k, a_k, b_k)$, либо $y_k(a, b) = x(a, b)$ для всех $(a, b) \preceq_{A,B} (a_k, b_k)$ и, следовательно, $(a_k, b_k) \prec_{A,B} (a_{k+1}, b_{k+1})$. Это невозможно, так как $N(y_k, a_k, b_k)$ — натуральные числа и $A \times B$ — конечное множество. Значит, последовательность Y конечна. Понятно, что ее последний элемент совпадает с x , поэтому $x = z(P, C, \prec_{A,B}) \in Z^*(P, C)$. $\square$

Отношение полного порядка $\prec_A$ на A назовем Q -упорядочением, если $a_1 \prec_A a_2$ влечет $Q(a_1) \leq Q(a_2)$.

Следствие 2. Если на A задано Q -упорядочение $\prec_A$, а на B — d -упорядочение $\prec_B$ и $C = \{(a, b) \mid q(b) \leq Q(a)\}$, то алгоритм $\text{GD}(P, C, \prec_{A,B})$ решает задачу $L(P, C)$.

ДОКАЗАТЕЛЬСТВО. Очевидно, что множество C согласовано с отношением $\prec_A$. Остается применить теорему 1. $\square$

4. КОМБИНИРОВАННЫЕ АЛГОРИТМЫ

Сформулируем комбинированные алгоритмы $\max \text{GI}$ и $\max \text{TGI}$, которые, как будет доказано, являются 0, 5-приближенными для МКР. Отношение полного порядка $\prec_B$ на B назовем v -упорядочением, если $b_1 \prec_B b_2$ влечет $v(b_1) \geq v(b_2)$.

Алгоритм $\max \text{GI}$

вход: задача $K(P)$; Q -упорядочение $\prec_A$ на A ; d -упорядочение $\prec_d$ на B ; v -упорядочение $\prec_v$ на B ;

выход: допустимое решение x задачи $K(P)$;

begin

(а) $\prec_B := \prec_d$; **выполнить** алгоритм $\text{GI}(P, \prec_{A,B})$; $x_1 := x(P, \prec_{A,B})$;

$\prec_B := \prec_v$; **выполнить** алгоритм $\text{GI}(P, \prec_{A,B})$; $x_2 := x(P, \prec_{A,B})$;

if $V(x_1) \geq V(x_2)$ **then** $x := x_1$ **else** $x := x_2$

end.

Известно [1, с. 28; 7, теорема 5], что при одном ранце алгоритм $\max \text{GI}$ имеет оценку относительной точности 0, 5. Идея этого алгоритма использована в [1, с. 166–167] при построении $1/(m+1)$ -приближенного алгоритма для МКР.

Алгоритм $\max \text{TGI}$ получается из алгоритма $\max \text{GI}$ заменой строки с меткой (а) следующей строкой:

$\prec_B := \prec_d$; **выполнить** алгоритм $\text{TGI}(P, \prec_{A,B})$; $x_1 := y(P, \prec_{A,B})$.

4.1. Схема доказательства основного результата. Утверждение (iii) леммы 1 показывает, что из 0, 5-приближенности алгоритма $\max \text{TGI}$ следует 0, 5-приближенность алгоритма $\max \text{GI}$. Пусть $\prec_A$ — Q -упорядочение и $\prec_B$ — d -упорядочение. Чтобы доказать 0, 5-приближенность алгоритма $\max \text{TGI}$ для задачи $K(P)$, мы, используя вектор $y = y(P, \prec_{A,B})$, построим множество B' «дополнительных» объектов. Затем сформируем расширенное множество объектов $B_1 = B \cup B'$, доопределим функции $q(\cdot)$, $v(\cdot)$ на множестве B_1 и расширим отношение $\prec_B$ до d -упорядочения $\prec_{B_1}$ множества B_1 . Пусть $P_1 =$

$\langle A, B_1, q(\cdot), v(\cdot), Q(\cdot) \rangle$. Мы построим множество $C \subseteq A \times B_1$, согласованное с отношением $\prec_A$ и включающее все пары (a, b) из $A \times B$ такие, что $q(b) \leq Q(a)$. По теореме 1 вектор z , построенный алгоритмом $\text{GD}(P_1, C, \prec_{A, B_1})$, — оптимальное решение задачи $L(P_1, C)$. Пусть V_1^* и V^* — оптимальные значения целевых функций в задачах $L(P_1, C)$ и $K(P)$ соответственно, а вектор x построен алгоритмом $\text{GI}(P, \prec_{A, B})$ с использованием v -упорядочения множества B в качестве $\prec_B$. Мы покажем, что $y = (z(a, b) \mid (a, b) \in A \times B)$, $V(x) \geq \sum_{b \in B'} v(b)$ и $V(x) + V(y) \geq V_1^* \geq V^*$. Отсюда $\max\{V(x), V(y)\} \geq 0,5V^*$.

4.2. Построение задачи $L(P_1, C)$. Выберем Q -упорядочение $\prec_A$ на A и d -упорядочение $\prec_B$ на B . Для $x \in X(P)$ и $a \in A$ положим $B^+(x, a) = \{b \mid x(a, b) = 1\}$ и $B^-(x, a) = \{b \in B(a) \mid r(x, a, b) = 0\}$. Если $B^-(x, a) \neq \emptyset$, положим $\beta(a) = \min(\prec_B, B^-(x, a))$.

Лемма 5. Пусть $y = y(P, \prec_{A, B})$. Тогда:

- (i) если $B^-(y, a) \neq \emptyset$, то $q(\beta(a)) > Q(a) - S(y, a)$;
- (ii) если $B^-(y, a_1) \neq \emptyset$, $a_1 \prec_A a$ и $r(y, a, \beta(a_1)) = 0$, то $B^+(y, a) \neq \emptyset$ и $B^+(y, a) \cap B(a_1) = \emptyset$.

ДОКАЗАТЕЛЬСТВО. (i) Пусть $B^-(y, a) \neq \emptyset$. По определению $\beta(a) \in B(a)$, $r(y, a, \beta(a)) = 0$ и $r(y, a, b) = 1$ для всех $b \prec_B \beta(a)$ таких, что $b \in B(a)$. Кроме того, $y(a, b) = 0$ для всех b таких, что $\beta(a) \prec_B b$ (по следствию 1). Тогда на шаге $(a, \beta(a))$ алгоритма TGI имеем $a \in A'$ (по утверждению (i) леммы 1) и $Q'(a) = Q(a) - S(y, a)$. Если $q(\beta(a)) \leq Q(a) - S(y, a)$, то алгоритм TGI на метке (d) установит $y(a, \beta(a)) = 1$, что противоречит $r(y, a, \beta(a)) = 0$.

(ii) Предположим, что $B^-(y, a_1) \neq \emptyset$, $a_1 \prec_A a$ и $r(y, a, \beta(a_1)) = 0$. Из $r(y, a, \beta(a_1)) = 0$ следует $y(a, \beta(a_1)) = 0$. Алгоритм TGI установит $y(a, \beta(a_1)) = 0$, только если на шаге $(a, \beta(a_1))$ выполнено одно из следующих условий: (a) $a \notin A'$, (b) $q(\beta(a_1)) > Q'(a)$, (c) $\beta(a_1) \notin B(a)$, (d) $q'(\beta(a_1)) = 0$. Условия (c) и (d) не выполняются, так как $\beta(a_1) \in B(a_1) \subseteq B(a)$ по определению и $r(y, a, \beta(a_1)) = 0$ по предположению. $B^+(y, a) \neq \emptyset$ следует из условия (a) по утверждению (ii) леммы 1 и из условия (b) — по (5). Пусть $b \in B^+(y, a)$. Тогда $r(y, a, b) = 1$, $b \neq \beta(a_1)$ (так как $r(y, a, \beta(a_1)) = 0$ по условию) и $r(y, a_1, b) = 0$ по (3). Учитывая определение $\beta(a_1)$, имеем либо $b \notin B(a_1)$, либо $\beta(a_1) \prec_B b$. Но $\beta(a_1) \prec_B b$ влекло бы $r(x, a, \beta(a_1)) = 1$ по следствию 1, что противоречит предположению. Значит, $b \notin B(a_1)$. $\square$

Пусть $y = y(P, \prec_{A, B})$. Для всех a таких, что $B^-(y, a) \neq \emptyset$, определим объекты $\delta(a)$ следующим образом. Допустим, что объекты $\delta(a)$ уже выбраны для всех $a \prec_A a_0$ таких, что $B^-(y, a) \neq \emptyset$. Если $\beta(a_0) \notin \{\delta(a) \mid a \prec_A a_0\}$, то положим $\delta(a_0) = \beta(a_0)$. Если $\beta(a_0) = \delta(a_1)$ для какого-то $a_1 \prec_A a_0$, найдем ранец a такой, что $a_1 \prec_A a \preceq_A a_0$ и $B^+(y, a) \cap \{\delta(a') \mid a_1 \prec_A a' \prec_A a_0\} = \emptyset$, и выберем $\delta(a_0)$ из $B^+(y, a)$. Докажем корректность предшествующего определения.

Лемма 6. Если $y = y(P, \prec_{A, B})$ и $B^-(y, a) \neq \emptyset$, то объект $\delta(a)$ определен и либо $\delta(a) = \beta(a)$, либо $r(y, a, \delta(a)) = 1$.

ДОКАЗАТЕЛЬСТВО. Пусть $y = y(P, \prec_{A, B})$ и $B^-(y, a_0) \neq \emptyset$. Допустим (гипотеза индукции), что значения $\delta(a)$ определены для всех $a \prec_A a_0$ таких, что $B^-(y, a) \neq \emptyset$, и либо $\delta(a) = \beta(a)$, либо $r(y, a, \delta(a)) = 1$. Если $\beta(a_0) \notin \{\delta(a) \mid a \prec_A a_0\}$, то $\delta(a_0) = \beta(a_0)$. Если $\beta(a_0) = \delta(a_1)$ для какого-то $a_1 \prec_A a_0$, то $r(y, a_1, \delta(a_1)) \leq r(y, a_0, \beta(a_0)) = 0$, и, учитывая предположение индукции, $\delta(a_1) = \beta(a_1)$. Пусть $M_1 = \{a \mid a_1 \prec_A a \preceq_A a_0\}$ и $M_2 = \{\delta(a) \mid a_1 \prec_A a \prec_A a_0\}$. Понятно, что $|M_1| > |M_2|$. Множества $B^+(y, a)$ попарно не пересекаются, поэтому найдется ранец $a \in M_1$, для которого $B^+(y, a) \cap M_2 = \emptyset$, при

этом $B^+(y, a) \neq \emptyset$ по утверждению (ii) леммы 5. Следовательно, удастся выбрать $\delta(a_0) \in B^+(y, a)$ в соответствии с определением. Из $a \preceq_A a_0$ следует $r(y, a_0, \delta(a_0)) = 1$. $\square$

Следующая лемма описывает используемые в дальнейшем свойства набора объектов $\delta(a)$.

Лемма 7. Если $y = y(P, \prec_{A,B})$ и $B^-(y, a) \neq \emptyset$, то:

- (i) $\delta(a) \in B(a)$;
- (ii) объекты $\delta(a)$ попарно различны;
- (iii) $\delta(a) \preceq_B \beta(a)$;
- (iv) $q(\delta(a)) > Q(a) - S(y, a)$.

ДОКАЗАТЕЛЬСТВО. (i) По лемме 6 либо $\delta(a) = \beta(a)$, либо $r(y, a, \delta(a)) = 1$. В любом случае $\delta(a) \in B(a)$.

(ii) Пусть $B^-(y, a_0) \neq \emptyset$. Покажем, что $\delta(a_0) \notin \{\delta(a') \mid a' \prec_A a_0\}$. По определению $\delta(a_0)$ это верно, если $\delta(a_0) = \beta(a_0)$. Если же $\delta(a_0) \neq \beta(a_0)$, то найдутся a_1 и a_2 такие, что

$$\delta(a_0) \notin \{\delta(a) \mid a_1 \prec_A a \prec_A a_0\}, \quad (11)$$

$$a_1 \prec_A a_2 \preceq_A a_0, \quad \beta(a_0) = \delta(a_1), \quad \delta(a_0) \in B^+(y, a_2). \quad (12)$$

Отсюда $r(y, a_1, \delta(a_1)) \leq r(y, a_2, \delta(a_1)) \leq r(y, a_0, \beta(a_0)) = 0$ и $\delta(a_1) = \beta(a_1)$ по лемме 6. Следовательно, $r(y, a, \beta(a_1)) \leq r(y, a_0, \beta(a_1)) = r(y, a_0, \beta(a_0)) = 0$ и, по утверждению (ii) леммы 5, $B^+(y, a_2) \cap B(a_1) = \emptyset$, поэтому $\delta(a_0) \notin B(a_1)$. С учетом утверждения (i), $\delta(a) \in B(a) \subseteq B(a_1)$ для $a \preceq_A a_1$, поэтому $\delta(a_0) \notin \{\delta(a) \mid a \preceq_A a_1\}$. Отсюда, с учетом (11), следует утверждение (ii).

(iii) Предположим, что $\delta(a) \neq \beta(a)$. Тогда, по определению $\delta(a)$, существуют a_1 и a_2 , удовлетворяющие условиям (11), (12) при $a_0 = a$. По утверждению (i) из $\beta(a) = \delta(a_1)$ следует, что $\beta(a) \in B(a_1) \subseteq B(a_2)$, при этом $r(y, a_2, \beta(a)) \leq r(y, a, \beta(a)) = 0$. Применяя следствие 1 (с заменой a_0 , b_0 и b на a_2 , $\delta(a)$ и $\beta(a)$ соответственно), получим $\delta(a) \prec_B \beta(a)$.

(iv) Если $\delta(a) = \beta(a)$, то утверждение (iv) эквивалентно утверждению (i) леммы 5. Если же $\delta(a) \neq \beta(a)$, то существуют a_1 и a_2 , удовлетворяющие условиям (11), (12) при $a_0 = a$. Тогда из $r(y, a, \beta(a)) = 0$ следует $r(y, a_1, \delta(a_1)) = r(y, a_2, \beta(a)) = 0$, и $\beta(a) = \delta(a_1) = \beta(a_1)$ по лемме 6. Значит, $r(y, a_2, \beta(a_1)) \leq r(y, a, \beta(a)) = 0$. Тогда $B^+(y, a_2) \cap B(a_1) = \emptyset$ по утверждению (ii) леммы 5, и из $\delta(a) \in B^+(y, a_2)$ следует $\delta(a) \notin B(a_1)$. Отсюда и из $\beta(a) = \delta(a_1) \in B(a_1)$, используя утверждение (i) леммы 5, получаем $q(\delta(a)) > Q(a_1) \geq q(\beta(a)) > Q(a) - S(y, a)$. $\square$

Пусть $y = y(P, \prec_{A,B})$, $A_1 = \{a \in A \mid B^-(y, a) \neq \emptyset\}$. Для каждого $a \in A_1$ введем новый объект $\gamma(a)$ с параметрами $q(\gamma(a)) = Q(a) - S(y, a)$ и $v(\gamma(a)) = q(\gamma(a)) \cdot d(\delta(a))$. Определение $v(\gamma(a))$ корректно, так как для $a \in A_1$ объект $\delta(a)$ существует (по лемме 6).

Положим $B' = \{\gamma(a) \mid a \in A_1\}$ и $B_1 = B \cup B'$. Функции $q(\cdot)$ и $v(\cdot)$ теперь определены на B_1 . Отношение $\prec_{B_1}$ на B_1 установим следующим образом: на B отношение $\prec_{B_1}$ совпадает с $\prec_B$, и объект $\gamma(a)$ непосредственно предшествует $\delta(a)$.

Если $A_1 = \emptyset$, то вектор y является оптимальным решением задачи $K(P)$, поскольку все объекты размещены. Учитывая это, будем считать, что $A_1 \neq \emptyset$. Очевидно, что отношение $\prec_{B_1}$ вполне упорядочивает множество B_1 . Покажем, что это d -упорядочение.

Лемма 8. Отношение $\prec_{B_1}$ упорядочивает B_1 по невозрастанию эффективности объектов.

ДОКАЗАТЕЛЬСТВО. Пусть $d(b) = v(b)/q(b)$ для $b \in B_1$. Покажем, что

$$b_1 \prec_{B_1} b_2 \rightarrow d(b_1) \geq d(b_2) \text{ для всех } b_1 \text{ и } b_2 \text{ из } B_1. \quad (13)$$

Очевидно, что условие (13) выполнено для b_1 и b_2 из B . Если же $b = \gamma(a)$, то $d(b) = v(\gamma(a))/q(\gamma(a)) = d(\delta(a))$, поэтому размещение $\gamma(a)$ непосредственно перед $\delta(a)$ сохраняет условие (13). $\square$

Итак, имеем Q -упорядочение $\prec_A$ на A , расширенное множество объектов B_1 с d -упорядочением $\prec_{B_1}$ и определенные на B_1 функции $q(\cdot)$ и $v(\cdot)$. Положим $C = \{(a, b) \in A \times B \mid q(b) \leq Q(a)\} \cup \{(a, \gamma(a')) \mid a' \preceq_A a\}$ и $P_1 = \langle A, B_1, q(\cdot), v(\cdot), Q(\cdot) \rangle$. Теперь определена задача $L(P_1, C)$.

4.3. Свойства оптимального решения задачи $L(P_1, C)$. Пусть $Z(P_1, C)$ и $V_1(\cdot)$ — множество допустимых решений и целевая функция задачи $L(P_1, C)$ соответственно; V^* и V_1^* — оптимальные значения целевых функций в задачах $K(P)$ и $L(P_1, C)$ соответственно.

Положим $u_0 = z(P_1, C, \prec_{A, B_1}) = (u_0(a, b) \mid (a, b) \in A \times B_1)$.

Лемма 9. *Выполняются соотношения $V^* \leq V_1^* = V_1(u_0)$.*

ДОКАЗАТЕЛЬСТВО. Пусть $x \in X(P)$ и $u(x) = (u(a, b) \mid (a, b) \in A \times B_1)$, где $u(a, b) = x(a, b)$ при $b \in B$ и $u(a, b) = 0$ при $b \in B'$. Очевидно, что $u(x) \in Z(P_1, C)$, $V_1(u(x)) = V(x)$ и, следовательно, $V^* \leq V_1^*$.

Пусть $(a, b) \in C$ и $a \prec_A a_1$. Если $b \in B$, то $q(b) \leq Q(a) \leq Q(a_1)$, поэтому $(a_1, b) \in C$. Если же $b = \gamma(a')$, то из $(a, b) \in C$ по определению множества C следует, что $a' \preceq_A a$. Тогда $a' \prec_A a_1$ и $(a_1, b) \in C$ по определению. Следовательно, множество C согласовано с отношением $\prec_A$. С учетом леммы 8 из теоремы 1 следует, что u_0 — оптимальное решение задачи $L(P_1, C)$. $\square$

Для $x \in X(P)$, $u = (u(a, b) \mid (a, b) \in A \times B_1)$ и $a_0 \in A$ обозначим

$$s(x, a_0, b_0) = \sum_{\{b \in B \mid b \prec_{B_1} b_0\}} q(b)x(a_0, b), \quad s_1(u, a_0, b_0) = \sum_{\{b \in B_1 \mid b \prec_{B_1} b_0\}} q(b)u(a_0, b).$$

Положим $y_0 = y(P, \prec_{A, B})$.

Теорема 2. (i) *Для всех $(a, b) \in A \times B$ выполняется равенство $u_0(a, b) = y_0(a, b)$;*

(ii) *Для всех $a \in A_1$ выполняется равенство $u_0(a, \gamma(a)) = 1$.*

ДОКАЗАТЕЛЬСТВО. На шаге (a_0, b_0) алгоритма $GD(P_1, C, \prec_{A, B_1})$ значения $u_0(a, b)$ уже известны для всех пар $(a, b) \prec_{A, B_1} (a_0, b_0)$. Допустим (предположение индукции), что для всех таких пар выполнены следующие условия: если $b \in B$, то $u_0(a, b) = y_0(a, b)$; если $a \in A_1$ и $b = \gamma(a)$, то $u_0(a, b) = 1$. При этом предположении

$$u_0(a_0, \gamma(a)) = 0, \quad \text{если } \gamma(a) \preceq_{B_1} b_0, a \neq a_0. \quad (14)$$

Действительно, пусть $\gamma(a) \preceq_{B_1} b_0$, $a \neq a_0$. Если $a_0 \prec_A a$, то $(a_0, \gamma(a)) \notin C$ и $u_0(a_0, \gamma(a)) = 0$ в соответствии с алгоритмом GD . Если же $a \prec_A a_0$, то $(a, \gamma(a)) \prec_{A, B_1} (a_0, b_0)$, $u_0(a, \gamma(a)) = 1$ по предположению индукции и $u_0(a_0, \gamma(a)) = 0$ по (3).

(i) Пусть $b_0 \in B$. Если $b \prec_{B_1} b_0$ и $b \in B$, то $(a_0, b) \prec_{A, B_1} (a_0, b_0)$ и по предположению индукции $u_0(a_0, b) = y_0(a_0, b)$. Выше мы определили объекты $\gamma(a)$ только для $a \in A_1$. Далее удобно считать, что для каждого $a \in A \setminus A_1$ тоже определен объект $\gamma(a)$ такой, что $q(\gamma(a)) = 0$. Тогда, учитывая (14), можем записать

$$s_1(u_0, a_0, b_0) = q(\gamma(a_0)) \cdot u_0(a_0, \gamma(a_0)) + s(y_0, a_0, b_0). \quad (15)$$

Отсюда $s_1(u_0, a_0, b_0) \leq q(\gamma(a_0)) + s(y_0, a_0, b_0)$. По определению $q(\gamma(a_0)) = 0$, если $a_0 \notin A_1$, и $q(\gamma(a_0)) = Q(a_0) - S(y_0, a_0)$, если $a_0 \in A_1$. Кроме того, $Q(a_0) \geq S(y_0, a_0)$ по (2). Тогда при любом $a \in A$ имеем

$$Q(a_0) - s_1(u_0, a_0, b_0) \geq S(y_0, a_0) - s(y_0, a_0, b_0). \quad (16)$$

Если $b_0 \notin B(a_0)$, то $u_0(a_0, b_0) = y_0(a_0, b_0) = 0$ по построению. Если $y_0(a, b_0) = 1$ для некоторого $a \prec_A a_0$, то $u(a, b_0) = 1$ по предположению индукции и, по условию (3), $u_0(a_0, b_0) = y_0(a_0, b_0) = 0$. Рассмотрим случай $b_0 \in B(a_0)$ и $y_0(a, b_0) = 0$ для всех $a \prec_A a_0$. Тогда с учетом предположения индукции

$$\sum_{a \prec_A a_0} y_0(a, b_0) = \sum_{a \prec_A a_0} u_0(a, b_0) = 0. \quad (17)$$

Предположим, что $y_0(a_0, b_0) = 1$. Тогда $b_0 \in B(a_0)$, $(a_0, b_0) \in C$ и $S(y_0, a_0) \geq s(y_0, a_0, b_0) + q(b_0)$. С учетом (16) $Q(a_0) - s_1(u_0, a_0, b_0) \geq q(b_0)$. В сочетании с (17) это гарантирует, что алгоритм GD на шаге (a_0, b_0) установит $u_0(a_0, b_0) = 1 = y_0(a_0, b_0)$.

Теперь допустим, что $y_0(a_0, b_0) = 0$. Тогда $r(y_0, a_0, b_0) = 0$, и из $b_0 \in B(a_0)$ следует, что $b_0 \in B^-(y_0, a_0)$, поэтому $a_0 \in A_1$ и определены объекты $\beta(a_0)$, $\delta(a_0)$. Из определения отношения $\prec_{B1}$, утверждения (iii) леммы 7 и определения $\beta(a)$ получим $\gamma(a_0) \prec_{B1} \delta(a_0) \preceq_B \beta(a_0) \preceq_B b_0$. Тогда $(a_0, \gamma(a_0)) \prec_{A, B1} (a_0, b_0)$ и $u_0(a_0, \gamma(a_0)) = 1$ по предположению индукции. Из (15) следует, что $s_1(u_0, a_0, b_0) = q(\gamma(a_0)) + s(y_0, a_0, b_0)$, тогда $Q(a_0) - s_1(u_0, a_0, b_0) = S(y_0, a_0) - s(y_0, a_0, b_0)$. Но $y_0(a_0, b_0) = 0$ влечет $S(y_0, a_0) = s(y_0, a_0, b_0)$ (следствие 1), поэтому $Q(a_0) = s_1(u_0, a_0, b_0)$. Строка с меткой (с) алгоритма GD на шаге (a_0, b_0) даст $u_0(a_0, b_0) = 0$.

(ii) Допустим, что $a_0 \in A_1$ и $b_0 = \gamma(a_0)$. Если $a \prec_A a_0$, то $(a, b_0) \notin C$ и $u_0(a, b_0) = 0$. Используя (14) и предположение индукции, получим $s_1(u_0, a_0, b_0) = \sum_{\{b \in B \mid b \prec_B b_0\}} q(b)u_0(a_0, b) = s(y_0, a_0, b_0) \leq S(y_0, a_0)$. По определению $q(\gamma(a_0)) = Q(a_0) - S(y_0, a_0) \leq Q(a_0) - s_1(u_0, a_0, b_0)$. Тогда строка с меткой (с) алгоритма GD на шаге (a_0, b_0) даст $u_0(a_0, b_0) = 1$. $\square$

Следствие 3. Справедливо неравенство $V_1^* \leq V(y_0) + \sum_{a \in A_1} v(\delta(a))$.

Доказательство. Для каждого $a \in A_1$ по определению

$$v(\gamma(a)) = q(\gamma(a)) \cdot d(\delta(a)) = \frac{Q(a) - S(y, a)}{q(\delta(a))} v(\delta(a)),$$

поэтому из утверждения (iv) леммы 7 следует, что $v(\gamma(a)) < v(\delta(a))$. Используя лемму 9, теорему 2 и утверждение (iv) леммы 7, получим

$$\begin{aligned} V_1^* = V_1(u_0) &= \sum_{b \in B} v(b) \sum_{a \in A} u_0(a, b) + \sum_{b \in B'} v(b) \sum_{a \in A} u_0(a, b) \\ &= \sum_{b \in B} v(b) \sum_{a \in A} y_0(a, b) + \sum_{a \in A_1} v(\gamma(a)) \leq V(y_0) + \sum_{a \in A_1} v(\delta(a)). \quad \square \end{aligned}$$

4.4. Основной результат. Пусть на A задано Q -упорядочение $\prec_A$, а на B — v -упорядочение $\prec_B$. Для каждого $a \in A_1$ построим объект $\varphi(a)$ следующим образом: если объекты $\varphi(a') \in B(a')$ определены для всех $a' \in A_1$ таких, что $a' \prec_A a$, то положим

$$\begin{aligned} \Phi(a) &= \{\varphi(a') \mid a' \in A_1, a' \prec_A a\}, \quad \varphi(a) = \min(\prec_B, B(a) \setminus \Phi(a)), \\ D(a) &= \{\delta(a') \mid a' \in A_1, a' \prec_A a\}. \end{aligned}$$

Следующая лемма доказывает корректность определения $\varphi(a)$.

Лемма 10. Если $a \in A_1$, то значение $\varphi(a)$ определено.

ДОКАЗАТЕЛЬСТВО. Понятно, что $|\Phi(a)| \leq |\{a' \in A_1 \mid a' \prec_A a\}|$. Из утверждения (ii) леммы 7 следует, что $|D(a)| = |\{a' \in A_1 \mid a' \prec_A a\}|$ и $\delta(a) \notin D(a)$. Из утверждения (i) леммы 7 и определения Q -упорядочения следует, что $D(a) \cup \{\delta(a)\} \subseteq B(a)$. Тогда $|\Phi(a)| \leq |D(a)| < |B(a)|$ и $B(a) \setminus \Phi(a) \neq \emptyset$. $\square$

Лемма 11. Пусть $\Phi = \{\varphi(a) \mid a \in A_1\}$, $D = \{\delta(a) \mid a \in A_1\}$. Существует изоморфизм $\pi: D \mapsto \Phi$ такой, что $v(\pi(b)) \geq v(b)$ для всех $b \in D$.

ДОКАЗАТЕЛЬСТВО. Пусть определен изоморфизм $\pi: D(a_0) \mapsto \Phi(a_0)$, удовлетворяющий условию $v(\pi(b)) \geq v(b)$ для всех $b \in D(a_0)$ (гипотеза индукции). Положим $\Phi_1(a_0) = \{\varphi(a) \mid a \prec_A a_0, \varphi(a) \prec_B \varphi(a_0)\}$ и покажем, что $\Phi_1(a_0) = \{b \in B(a_0) \mid b \prec_B \varphi(a_0)\}$. Действительно, если $b \in B(a_0)$ и $b \prec_B \varphi(a_0)$, то $b = \varphi(a)$ для какого-то $a \prec_A a_0$ по определению $\varphi(a_0)$, откуда $b \in \Phi_1(a_0)$. Обратно, если $b \in \Phi_1(a_0)$, то найдется $a \prec_A a_0$, для которого $b = \varphi(a) \prec_B \varphi(a_0)$, причем $b \in B(a) \subseteq B(a_0)$. Следовательно, $\Phi_1(a_0) = \{b \in B(a_0) \mid b \prec_B \varphi(a_0)\}$. Обозначим $D_1(a_0) = \pi^{-1}(\Phi_1(a_0))$, $D_1^+ = D_1(a_0) \cup \{\delta(a_0)\}$ и $\Phi_1^+ = \Phi_1(a_0) \cup \{\varphi(a_0)\}$. Множества $\Phi_1(a_0)$ и $D_1(a_0)$ равномощны, положим $k = |\Phi_1(a_0)| = |D_1(a_0)|$. Перечислим множества D_1^+ и Φ_1^+ в соответствии с упорядочением $\prec_B$: $D_1^+ = \{b_1^1, \dots, b_{k+1}^1\}$ и $\Phi_1^+ = \{b_1^2, \dots, b_{k+1}^2\}$. Объекты $\delta(a)$ попарно различны по утверждению (ii) леммы 7, поэтому множество D_1^+ включает $k+1$ элемент из $B(a_0)$. Объекты $\varphi(a)$ попарно различны по построению, поэтому множество Φ_1^+ включает $k+1$ первых (в упорядочении $\prec_B$) элементов из $B(a_0)$. Тогда $v(b_i^1) \leq v(b_i^2)$ для любого $i \in \{1, \dots, k+1\}$.

Не изменяя $\pi(b)$ для $b \in D(a_0) \setminus D_1(a_0)$, переопределим соответствие π на $D_1(a_0)$ и доопределим его на $\delta(a_0)$: $\pi(b_i^1) = b_i^2$. Теперь π отображает множество $D(a_0) \cup \{\delta(a_0)\}$ на $\Phi(a_0) \cup \{\varphi(a_0)\}$. Взаимная однозначность отображения сохраняется. $\square$

Следствие 4. Справедливо неравенство $\sum_{a \in A_1} v(\delta(a)) \leq \sum_{a \in A_1} v(\varphi(a))$.

ДОКАЗАТЕЛЬСТВО. Пусть $\pi: D \mapsto \Phi$ — отображение, построенное в лемме 11. Тогда $\sum_{a \in A_1} v(\delta(a)) = \sum_{b \in D} v(b) \leq \sum_{b \in D} v(\varphi(b)) = \sum_{b \in \Phi} v(b)$. $\square$

Покажем, что все объекты $\varphi(a)$ войдут в размещение $x(P, \prec_{A,B})$.

Лемма 12. Если $x = x(P, \prec_{A,B})$, то $r(x, a, \varphi(a)) = 1$ для всех $a \in A_1$.

ДОКАЗАТЕЛЬСТВО. Пусть $r(x, a, \varphi(a)) = 1$ для всех a из A_1 таких, что $a \prec_A a_0$ (гипотеза индукции). Обозначим $\Phi^-(a_0) = B(a_0) \setminus \Phi(a_0)$, $M(a_0) = \{b \in B(a_0) \mid \sum_{a \prec a_0} x(a, b) = 0\}$ и $b_0 = \min(\prec_B, M(a_0))$. Пусть $r(x, a_0, \varphi(a_0)) = 0$.

Тогда $\varphi(a_0) \in M(a_0)$. Из индукционной гипотезы следует, что $b_0 \in \Phi^-(a_0)$. Но $\varphi(a_0) = \min(\prec_B, \Phi^-(a_0))$, поэтому $\varphi(a_0) = b_0$. Отсюда $x(a, \varphi(a_0)) = 0$ для всех $a \prec_A a_0$ и $x(a_0, b) = 0$ для всех $b \prec_B b_0$ из $B(a_0)$ (так как $\sum_{a \prec a_0} x(a, b) = 1$

для такого b). Тогда из (5) и (6) следует, что на шаге $(a_0, \varphi(a_0))$ алгоритма GI($P, \prec_{A,B}$) будет $q'(\varphi(a_0)) = 0$ и $Q'(a_0) = Q(a_0)$, вследствие чего алгоритм выйдет на строку с меткой (с) и установит $x(a_0, \varphi(a_0)) = 1$, а это противоречит предположению $r(x, a_0, \varphi(a_0)) = 0$. $\square$

Теорема 3. Алгоритм maxTGI строит 0,5-приближенное решение задачи о многих ранцах.

ДОКАЗАТЕЛЬСТВО. Применяя последовательно лемму 9, следствие 3, следствие 4 и лемму 12, получим

$$V^* \leq V_1^* \leq V(y_0) + \sum_{a \in A_1} v(\delta(a)) \leq V(y_0) + \sum_{a \in A_1} v(\varphi(a)) \leq V(y_0) + V(x_0).$$

Здесь V^* — оптимальное значение целевой функции в МКР (1)–(4), а $V(y_0)$ и $V(x_0)$ — значения этой функции на векторах, построенных соответственно алгоритмом TGI с d -упорядочением на B и алгоритмом GI с v -упорядочением на B , при Q -упорядочении множества A в обоих случаях. $\square$

Наконец, по утверждению (iii) леммы 1 из 0, 5-приближенности алгоритма $\max\text{TGI}$ следует 0, 5-приближенность алгоритма $\max\text{GI}$.

Следствие 5. Алгоритм $\max\text{GI}$ строит 0, 5-приближенное решение задачи о многих ранцах.

ЗАКЛЮЧЕНИЕ

Построен новый 0, 5-приближенный алгоритм для задачи о нескольких ранцах, который может быть использован, например, при распределении вычислительного ресурса в многопроцессорной системе. Алгоритм имеет трудоемкость $O(mn)$ (без учета сортировок) и обобщает на случай $m > 1$ алгоритм из [7]. Такие же оценки относительной точности и трудоемкости имеет алгоритм, описанный в [2, с. 299]. Указанные алгоритмы порождают, как правило, различные распределения, поэтому в приложениях целесообразно применять оба алгоритма и выбирать лучший из полученных результатов.

Авторы благодарны участникам семинара «Дискретные экстремальные задачи» ИМ СО РАН за полезные замечания и рекомендации.

ЛИТЕРАТУРА

1. Martello S., Toth P. Knapsack Problems: Algorithms and Computer Implementations. Chichester; N. Y.: John Wiley & Sons, 1990.
2. Kellerer H., Pferschy U., Pisinger D. Knapsack Problems. Berlin; Heidelberg: Springer-Verl., 2004.
3. Papadimitriou C., Steiglitz K. Combinatorial Optimization: Algorithms and Complexity. Englewood Cliffs: Prentice Hall, 1982.
4. Chekuri C., Khanna S. A polynomial time approximation scheme for the multiple knapsack problem // SIAM J. Computing. 2003. V. 5, Issue 3. P. 713–728.
5. Jansen K. Parameterized approximation scheme for the multiple knapsack problem // SIAM J. Computing. 2009. V. 39, Issue. 4. P. 1392–1412.
6. Dantzig G. D. Discrete variable extremum problems // Operations Research. 1957. V. 5, N 2. P. 266–277.
7. Mualem A., Nisan N. Truthful approximation mechanisms for restricted combinatorial auctions // Games and Economic Behavior. 2008. V. 64, Issue 2. P. 612–631.

Статья поступила 12 октября 2017 г.

Хуторецкий Александр Борисович
Новосибирский государственный университет
ул. Пирогова, 2

Бредихин Сергей Всеволодович
Институт вычислительной математики и математической геофизики СО РАН
просп. Акад. Лаврентьева, 6

Замятин Александр Александрович
Новосибирский государственный университет
630090 г. Новосибирск

E-mail: hab@dus.nsc.ru; bred@nsc.ru; stekbox@gmail.com