

ОБ ОДНОМ АЛГОРИТМЕ РЕШЕНИЯ  
ТРЕХИНДЕКСНОЙ АКСИАЛЬНОЙ ЗАДАЧИ  
О НАЗНАЧЕНИЯХ НА ОДНОЦИКЛИЧЕСКИХ  
ПОДСТАНОВКАХ\*)

*Э. Х. Гимади, Н. М. Коркишко*

Рассматривается трехиндексная аксиальная задача о назначениях на одноциклических подстановках. Приводится условие разрешимости задачи. Представлен алгоритм приближенного решения задачи с временной сложностью  $O(n^2)$ . Проведен анализ работы алгоритма, доказана корректность его работы. Указываются условия асимптотической точности алгоритма для решения задачи на случайных входах.

**Введение**

*Трехиндексная аксиальная задача о назначениях* [4] состоит в таком выборе  $n$  элементов кубической матрицы  $(c_{ijk})$  порядка  $n$ , что в каждом ее сечении имеется ровно один выбранный элемент и при этом сумма таких элементов минимальна. (Под сечением матрицы понимается множество  $n^2$  ее элементов с фиксированным значением одного из индексов  $i, j$  или  $k$ ,  $1 \leq i, j, k \leq n$ .)

Задача может быть сформулирована в виде следующей задачи линейного целочисленного программирования: найти минимум функции

$$\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n c_{ijk} x_{ijk}$$

---

\*) Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проект 02-01-01153), научной программы «Университеты России» (проект 04.01.012) и INTAS (грант 00-217).

при следующих ограничениях:

$$\begin{aligned}\sum_{j=1}^n \sum_{k=1}^n x_{ijk} &= 1, \quad 1 \leq i \leq n; \\ \sum_{i=1}^n \sum_{k=1}^n x_{ijk} &= 1, \quad 1 \leq j \leq n; \\ \sum_{i=1}^n \sum_{j=1}^n x_{ijk} &= 1, \quad 1 \leq k \leq n; \\ x_{ijk} &\in \{0, 1\}, \quad 1 \leq i, j, k \leq n,\end{aligned}$$

или в алгебраической форме:

$$\text{минимизировать} \quad \sum_{i=1}^n c_{i, \pi(i), \sigma\pi(i)} \quad (1)$$

по всем подстановкам  $\pi, \sigma$  степени  $n$ .

В отличие от двухиндексного случая трехиндексная аксиальная задача о назначении NP-трудна [5–7].

*Трехиндексная аксиальная задача о назначении на одноциклических подстановках* отличается от классической задачи (1) дополнительным требованием одноциклическости каждой из трех подстановок

$$\pi, \sigma, \sigma\pi \in P_n, \quad (2)$$

где  $P_n$  — множество всех одноциклических подстановок степени  $n$ .

Двухиндексная задача о назначениях на одноциклических подстановках совпадает с классической задачей коммивояжера. Отсюда следует, что для рассматриваемой в статье задачи (1)–(2) проблематично построение полиномиальной приближенной схемы (PTAS) [7–9].

В отличие от двухиндексной задачи в трехиндексной задаче существуют входы, на которых нет решения.

Задачу будем называть *разрешимой*, если для нее существуют подстановки  $\pi, \sigma$ , удовлетворяющие условию (2). В противном случае задача называется *неразрешимой*. Критерий разрешимости рассматриваемой задачи состоит в следующем.

**Теорема 1** ([1, 3]). *Трехиндексная задача о назначениях на одноциклических подстановках степени  $n$  разрешима тогда и только тогда, когда  $n$  нечетно.*

В [3] для решения задачи (1)–(2) представлен приближенный алгоритм с временной сложностью  $O(n^3)$ . Алгоритм основан на таком согласованном преобразовании подстановок  $\pi$  и  $\sigma$ , которое почти всегда

(с вероятностью, стремящейся к 1 при  $n \rightarrow \infty$ ) приводит их (а также подстановку  $\sigma\pi$ ) к одноциклическому виду. При этом для отыскания исходной подстановки  $\sigma$  используется точный алгоритм решения задачи о назначении на специально сформированной квадратной матрице порядка  $n$ . Там же приведены условия асимптотической точности алгоритма решения задачи на случайных входных данных. К сожалению, идею этого алгоритма не удастся применить для случая числа индексов, большего трех.

Ниже опишем алгоритм, который в отличие от [3] всегда строит допустимое решение задачи и имеет существенно меньшую временную сложность. Кроме того, идею этого алгоритма можно, в принципе, применять и для случая большего числа индексов.

### 1. Приближенный алгоритм для решения трехиндексной аксиальной задачи о назначении на одноциклических подстановках

Перейдем к описанию алгоритма  $A$  для нахождения приближенного решения задачи (1)–(2). Так же, как и в [3], в начале работы алгоритма  $A$  формируется специальная квадратная матрица  $D = (d_{jk})$  порядка  $n$ , состоящая из части элементов исходной матрицы  $(c_{ijk})$ , где  $i = \pi^{-1}(j)$ :

$$d_{jk} = c_{\pi^{-1}(j)jk}, \quad 1 \leq j, k \leq n.$$

Здесь в качестве подстановки  $\pi \in P_n$  берется подстановка  $\pi = (1, 2, \dots, n)$ , которая далее в отличие от [3] остается без изменения. Кроме того, для отыскания одноциклической подстановки  $\sigma$  используется не точный, а менее трудоемкий приближенный метод, основанный на выборе минимального элемента в текущей строке с учетом дополнительного требования одноциклическости подстановки  $\sigma\pi$ .

Элементам подстановки  $\sigma\pi$  поставим в соответствие вершины ориентированного графа  $G$ , а переходам от элемента к элементу — дуги графа  $G$ . На начальном шаге работы алгоритма множество дуг графа  $G$  пусто. Далее граф представляется в виде (растущего от шага к шагу) частичного тура, т. е. такого подмножества дуг, которое может быть дополнено до гамильтонова обхода.

Пусть в ходе работы алгоритма в графе  $G$  возникает некоторый максимальный (по включению) путь  $(s, \dots, k)$ . Тогда в его начальной вершине  $s$  будем хранить информацию о его концевой вершине  $k$  в виде пометки  $p(s) = k$ , а в концевой вершине  $k$  — информацию о начальной вершине  $s$  в виде пометки  $q(k) = s$ .

Далее разность  $i - 1$  будем полагать равной  $n$  при  $i = 1$  и сумму  $i + 1$  равной 1 при  $i = n$ .

**Алгоритм А.**

*Этап 1.* Выбирается одноциклическая подстановка  $\pi = (1, 2, \dots, n)$  и формируется квадратная матрица  $D$ , как указано выше.

*Этап 2.* В матрице  $D$  исключаются из дальнейшего рассмотрения (запрещаются) диагональные  $d_{ii}$  и «поддиагональные»  $d_{i,i-1}$  элементы,  $i = 1, \dots, n$ , для того чтобы не появлялись непереставляемые элементы в подстановках  $\sigma$  и  $\sigma\pi$ .

*Этап 3.* Он состоит из  $n - 4$  шагов. На шаге  $t$ ,  $1 \leq t \leq n - 4$ , имеется подстановка  $\sigma^{(t)} = (i_1^{(t)}, i_2^{(t)}, \dots, i_n^{(t)})$ .

**Шаг 1.**

Шаг 1 начинается с подстановки  $\sigma^{(1)} = \pi$ . Первая строка выбирается в качестве текущей, и в ней просматриваются незапрещенные элементы матрицы  $D$ . Пусть  $k$  — номер столбца, в котором находится минимальный из просмотренных элементов. Тогда элементы 2 и  $k$  меняются местами и получается подстановка  $\sigma^{(2)}$ .

В результате таких действий в подстановке  $\sigma^{(1)}$  элемент 1 перейдет в  $k$ , а в подстановке  $\sigma^{(1)}\pi$  элемент  $n$  перейдет в  $k$ . При этом в графе  $G$  появится дуга  $(n, k)$ . Помечаются ее концы:  $p(n) = k$ ,  $q(k) = n$ . Чтобы избежать заикливания в подстановке  $\sigma\pi$ , в матрице  $D$  запрещается элемент  $d_{k+1,n}$ .

**Шаг  $t$ ,  $2 \leq t \leq n - 4$ .**

Перед шагом  $t$  имеется подстановка  $\sigma^{(t)} = (i_1^{(t)}, i_2^{(t)}, \dots, i_n^{(t)})$ . На шаге  $t$  в качестве текущей берется строка  $j = i_t^{(t)}$  и просматриваются элементы  $d_{jk}$ , расположенные в столбцах с номерами  $i_{i+1}^{(t)}, \dots, i_n^{(t)}$ . Эти столбцы и соответствующие им элементы текущей строки называются *свободными*. Пусть  $k$  — номер столбца, в котором находится минимальный из просмотренных и незапрещенных элементов. Поменяв в  $\sigma^{(t)}$  местами элементы  $i_{i+1}^{(t)}$  и  $k$ , получим подстановку  $\sigma^{(t+1)}$ .

В графе  $G$  появилась новая дуга  $(s, k)$ , где  $s = j - 1$ .

Если концевые вершины  $s$  и  $k$  этой дуги не помечены, то осуществляется их пометка:  $p(s) = k$ ,  $q(k) = s$ , а элемент  $d_{k+1,s}$  запрещается.

Если помечена только вершина  $s$ , то корректируются пометки  $q(k) = q(s)$  и  $p(q(s)) = k$  и запрещается элемент  $d_{k+1,q(s)}$ .

Если помечена только вершина  $k$ , то полагается  $p(s) = p(k)$  и  $q(p(k)) = s$  и запрещается элемент  $d_{p(k)+1,s}$ .

Если помечены обе вершины  $s$  и  $k$ , то полагается  $p(q(s)) = p(k)$  и  $q(p(k)) = q(s)$ . Элемент  $d_{p(k)+1,q(s)}$  запрещается.

Переход к следующему шагу.

Заметим, что после  $n - 4$  шагов работы алгоритма А граф  $G$  представляет собой частичный тур, содержащий  $n - 3$  вершины. Другими

словами, результат умножения первых  $n - 3$  элементов подстановки  $\sigma$  на одноциклическую подстановку  $\pi$  не содержит циклов.

*Этап 4.* Среди шести подстановок, получаемых из подстановки  $\sigma$  с помощью всевозможных перестановок трех последних элементов  $i_{n-2}$ ,  $i_{n-1}$  и  $i_n$ , выбирается подстановка  $\sigma'$ , удовлетворяющая условию (2).

На этом работа алгоритма  $A$  заканчивается. Получено допустимое решение  $\pi, \sigma'$  со значением целевой функции

$$f_A = \sum_{i=1}^n c_{i, \pi(i), \sigma' \pi(i)}.$$

## 2. Корректность работы алгоритма $A$

Покажем, что число шагов, выполняемых на этапе 3, выбрано корректно, т. е. на шаге  $t$ ,  $2 \leq t \leq n - 4$ , в очередной текущей строке (имеющей, очевидно, не менее четырех свободных элементов) должен быть хотя бы один свободный незапрещенный элемент.

**Утверждение 1.** *Перед каждым шагом  $t$ ,  $2 \leq t \leq n - 4$ , алгоритма  $A$  в текущей строке  $j = i_t^{(t)}$  матрицы  $D$  среди свободных элементов имеется не более трех запрещенных.*

**Доказательство.** Запрещенным среди свободных элементов в текущей строке является диагональный элемент и, возможно, «поддиагональный». Покажем, что кроме них среди свободных элементов может оказаться еще не более одного запрещенного элемента.

Предположим противное. Пусть в  $j$ -й строке матрицы  $D$  среди свободных элементов запрещены два элемента  $d_{jk}$  и  $d_{js}$ . Элемент  $d_{jk}$  запрещен, поскольку в графе  $G$  существует путь с началом  $k$  и концом  $j - 1$ . Аналогично элемент  $d_{js}$  мог быть запрещен только из-за того, что в графе  $G$  существует путь с началом  $s$  и концом  $j - 1$ . Следовательно, в графе  $G$  имеются два пути, заканчивающиеся одной и той же вершиной  $j - 1$ . Это противоречит тому, что на каждом шаге этапа 3 граф  $G$  является частичным туром. Утверждение 1 доказано.

Убедимся в корректности работы алгоритма на этапе 4.

**Утверждение 2.** *Среди шести циклических подстановок, полученных из подстановки  $\sigma$  с помощью всевозможных перестановок трех последних элементов  $i_{n-2}$ ,  $i_{n-1}$  и  $i_n$ , найдется подстановка  $\sigma'$ , удовлетворяющая условию (2).*

**Доказательство.** Напомним, что  $n$  — нечетное, а подстановки  $\sigma$  и  $\pi$  — одноциклические. Выясним четность числа циклов в подстановке

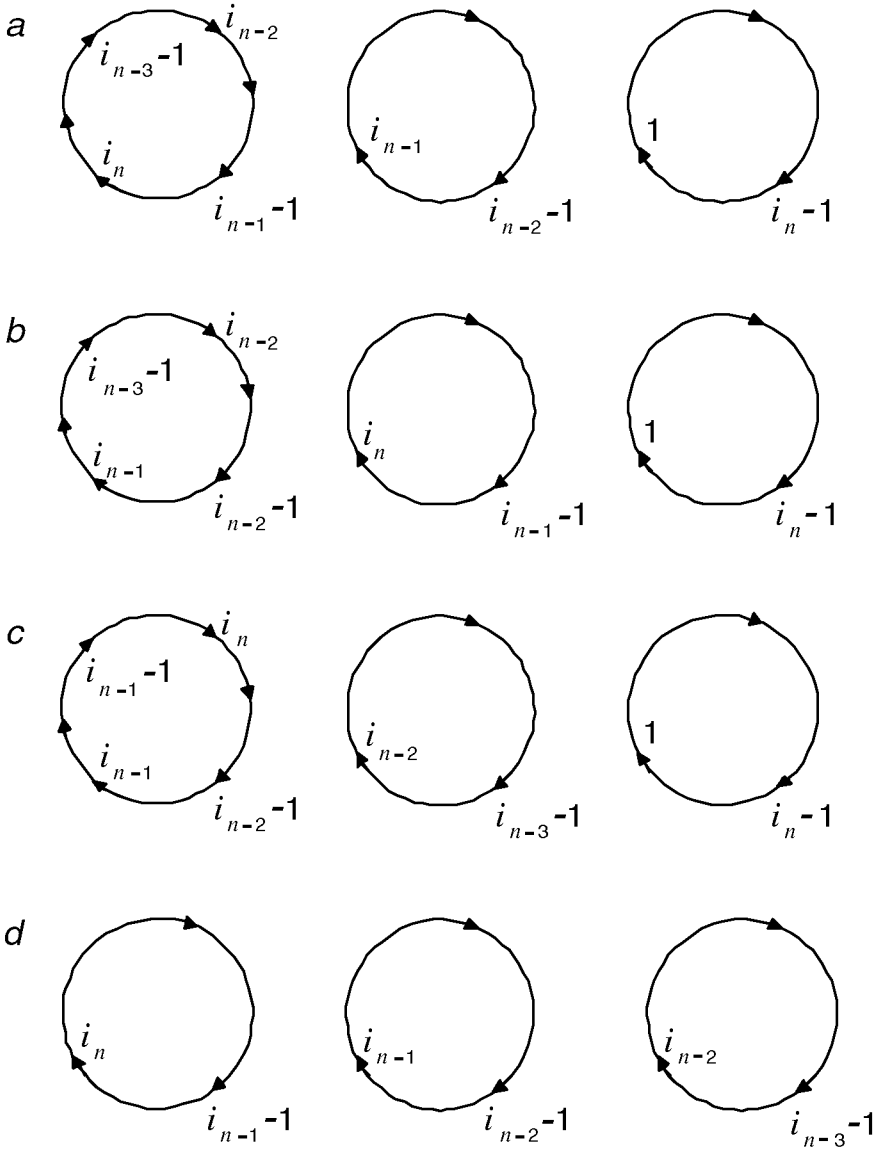


Рис. 1

$\sigma\pi$ . Подстановка  $\pi$  представима в виде произведения  $n-1$  транспозиций (число которых четно в силу нечетности  $n$ ). Поэтому при умножении одноклической подстановки  $\sigma$  на четное число транспозиций нечетность числа циклов сохраняется. Таким образом, подстановка  $\sigma\pi$  состоит из нечетного числа циклов.

По построению произведение первых  $n - 3$  элементов подстановки  $\sigma$  на подстановку  $\pi$  не содержит циклов и отличается от произведения  $\sigma$  на  $\pi$  на три элемента. Следовательно, подстановка  $\sigma\pi$  не может содержать более трех циклов, т. е. она состоит либо из одного, либо из трех циклов. Если число циклов равно 1, то  $\sigma' = \sigma$ .

Пусть подстановка  $\sigma\pi$  состоит из трех циклов. Элементы  $i_1 = 1$ ,  $i_{n-2}$ ,  $i_{n-1}$  и  $i_n$  подстановки  $\sigma$  могут распределиться по трем циклам четырьмя различными способами (рис. 1).

Более подробно проанализируем первый случай.

*Случай а.* Элементы  $i_{n-2}$  и  $i_n$  принадлежат первому циклу, элемент  $i_{n-1}$  — второму, а  $i_1 = 1$  — третьему циклу. В этом случае подстановка  $\sigma' = (1, i_2, \dots, i_{n-3}, i_{n-2}, i_n, i_{n-1})$  удовлетворяет условию (2). Проверим это.

Подстановка  $\sigma\pi$  имеет вид

$$\begin{pmatrix} n & i_2 - 1 & \dots & i_{n-3} - 1 & i_{n-2} - 1 & i_{n-1} - 1 & i_n - 1 \\ i_2 & i_3 & \dots & i_{n-2} & i_{n-1} & i_n & 1 \end{pmatrix}.$$

Напомним, что подстановка  $\sigma\pi$  состоит из трех независимых циклов.

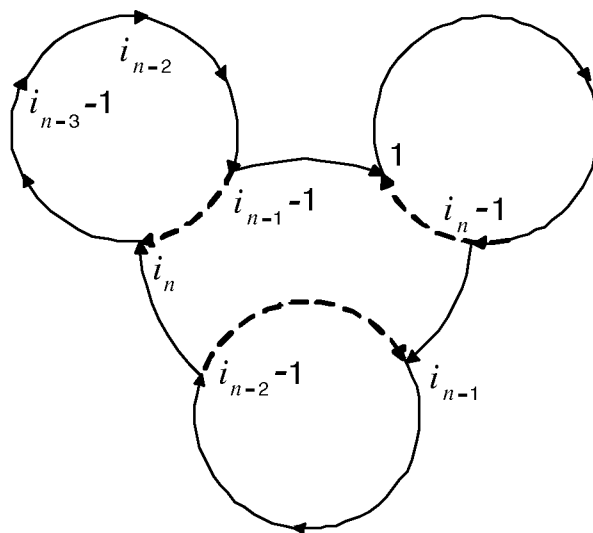


Рис. 2

В графическом изображении эти три цикла представляют собой 2-фактор. Удалением из каждого цикла по одной дуге и добавлением трех соединяющих дуг получаем замкнутый обход  $n$  вершин (рис. 2). Этот обход соответствует одноциклической подстановке  $\sigma'\pi$ , имеющей

следующий вид:

$$\begin{pmatrix} n & i_2 - 1 & \dots & i_{n-3} - 1 & i_n - 1 & i_{n-2} - 1 & i_{n-1} - 1 \\ i_2 & i_3 & \dots & i_{n-2} & i_{n-1} & i_n & 1 \end{pmatrix}.$$

Результаты анализа  $b, c$  и  $d$  приведем в сокращенном виде.

*Случай b.* Элементы  $i_{n-2}$  и  $i_{n-1}$  принадлежат одному циклу, элемент  $i_n$  — другому циклу. В этом случае подстановка

$$\sigma' = (1, i_2, \dots, i_{n-3}, i_{n-2}, i_n, i_{n-1})$$

удовлетворяет условию (2).

*Случай c.* Элементы  $i_{n-1}$  и  $i_n$  принадлежат одному циклу, элемент  $i_{n-2}$  — другому циклу. В этом случае подстановка

$$\sigma' = (1, i_2, \dots, i_{n-3}, i_n, i_{n-1}, i_{n-2})$$

удовлетворяет условию (2).

*Случай d.* Элементы  $i_{n-2}$ ,  $i_{n-1}$ ,  $i_n$  принадлежат разным циклам. В этом случае подстановка

$$\sigma' = (1, i_2, \dots, i_{n-3}, i_{n-2}, i_n, i_{n-1})$$

удовлетворяет условию (2).

Итак, во всех рассматриваемых случаях требуемые подстановки приведены. Утверждение 2 доказано.

### 3. Анализ оценок качества алгоритма $A$

Оценим временную сложность алгоритма  $A$ . Ясно, что каждый шаг этапа 3 имеет не более чем линейную временную сложность. Поэтому временная сложность этапа 3 алгоритма равна  $O(n^2)$ . Очевидно, что отыскание подстановки  $\sigma'$  на этапе 4 осуществляется за время  $O(n)$ . Таким образом, временная сложность алгоритма  $A$  в целом равна  $O(n^2)$ .

Перейдем к вероятностному анализу точности алгоритма для решения задачи на случайных входах, определяемых множеством  $M_n$  матриц  $(c_{ijk})$  размера  $n \times n \times n$ , где элементы  $c_{ijk}$  — независимые случайные величины, выбираемые из отрезка  $[a_n, b_n]$ , где  $b_n \geq a_n > 0$ , с одинаковой функцией распределения.

Обозначим через  $f_A(x)$  приближенное (полученное посредством алгоритма  $A$ ) и через  $f(x)$  оптимальное значение целевой функции задачи на входе  $x$ .

Будем говорить, что алгоритм  $A$  имеет оценки  $(\varepsilon_A, \delta_A)$  на случайных входах  $x$  рассматриваемой задачи, если выполнено неравенство

$$\Pr\{f_A(x) > (1 + \varepsilon_A)f(x)\} \leq \delta_A,$$



где  $\varepsilon_A$  есть оценка относительной погрешности решения, получаемого алгоритмом  $A$ ,  $\delta_A$  — вероятность несрабатывания алгоритма  $A$  (т. е. величина  $\delta_A$  равна доле случаев, когда алгоритм не гарантирует погрешность, не превосходящую  $\varepsilon_A$ ). Представляется интересным поведение оценок  $\delta_A$  и  $\varepsilon_A$  при увеличении размерности задачи.

Алгоритм  $A$  называется *асимптотически точным* на классе рассматриваемых задач, если существуют оценки  $\varepsilon_A$  и  $\delta_A$ , стремящиеся к нулю с ростом размерности задачи.

Для задачи отыскания минимального гамильтонова обхода вершин графа в статье [2] получены некоторые условия асимптотической точности приближенного алгоритма типа  $A$  (действующего по принципу выбора минимального элемента на всех шагах, кроме последнего). Например, когда расстояния между вершинами в графе равномерно распределены в интервале  $[a_n, b_n]$ , предлагаемый в [2] алгоритм асимптотически точен при  $b_n/a_n \leq n/(\psi_n \ln n)$ , где  $\psi_n$  — произвольная функция такая, что  $\psi_n \rightarrow \infty$  при  $n \rightarrow \infty$ . В нашем случае отказ от принципа выбора минимального элемента происходит на четырех последних шагах алгоритма  $A$  (на этапе 3). Однако легко видеть, что для решения рассматриваемой задачи посредством описанного выше алгоритма  $A$  оценки временной сложности и условия асимптотической точности, полученные в [2] (см. также [10, гл. 7]), остаются справедливыми.

**Замечание 1.** Можно построить модификацию алгоритма с существенно лучшей сходимостью к нулю вероятности несрабатывания, но с большей временной сложностью, а именно  $O(n^3)$ . Для этого выполняем алгоритм для каждой из  $n$  подстановок  $\pi \in P_n$  вида

$$\begin{pmatrix} 1 & 2 & \cdots & n-1 & n \\ r & r+1 & \cdots & r-2 & r-1 \end{pmatrix}, \quad r = 1, \dots, n.$$

При каждом таком выполнении алгоритм имеет дело с независимой случайной двумерной матрицей  $D^{(r)} = (d_{jk}^{(r)})$ , где  $d_{jk}^{(r)} = c_{ijk}$ ,  $i = \pi^{-1}(j)$ ,  $1 \leq j, k \leq n$ . Выбирая решение с минимальным значением целевой функции  $f_A = \min\{f_A^{(r)} \mid 1 \leq r \leq n\}$ , получим вероятность несрабатывания алгоритма, не превышающую величины  $(\delta_A)^n \rightarrow 0$  при  $n \rightarrow \infty$ .

**Замечание 2.** Используя принцип выбора максимального элемента, аналогичный приближенный алгоритм может быть построен для максимизационного варианта трехиндексной аксиальной задачи о назначении на одноциклических подстановках. Для такого варианта задачи также остаются справедливыми результаты, полученные для вероятностного анализа задачи отыскания максимального гамильтонова обхода вершин графа (см., например, [10, гл. 11]). В этом случае алгоритм асимптотически точен безотносительно каких-либо условий на отношение  $b_n/a_n$ .

В заключение авторы благодарят В. Д. Мазурова за ценные советы по обоснованию корректности предложенного алгоритма.

## ЛИТЕРАТУРА

1. **Гимади Э. Х., Кайран Н. М., Сердюков А. И.** О разрешимости многоиндексной аксиальной задачи о назначениях на одноциклических подстановках // Изв. вузов. Математика. 2000. № 12. С. 31–26.
2. **Гимади Э. Х., Перепелица В. А.** Асимптотический подход к решению задач коммивояжера // Управляемые системы: Сб. науч. тр. Вып. 12. Новосибирск: Ин-т математики СО АН СССР, 1974. С. 35–45.
3. **Гимади Э. Х., Сердюков А. И.** Аксиальные трехиндексные задачи о назначении и коммивояжера: быстрые приближенные алгоритмы и их вероятностный анализ // Изв. вузов. Математика. 1999. № 12. С. 19–25.
4. **Емеличев В. А., Ковалев М. М., Кравцов М. М.** Многогранники, графы, оптимизация. М.: Наука, 1981.
5. **Fon-Der-Flaass D. G.** Arrays of distinct representatives — a very simple NP-complete problem // Discrete Math. 1997. V. 171, N 1–3. P. 295–298.
6. **Frieze A. M.** Complexity of a 3-dimensional assignment problem // European J. Oper. Res. 1983. V. 13, N 2. P. 161–164.
7. **Garey M. R., Johnson D. S.** Computers and intractability. A guide to the theory of NP-completeness. San Francisco: W. H. Freeman and Co., 1979.
8. **Papadimitriou C. H., Yannakakis M.** Optimization, approximation, and complexity classes // J. Comput. System Sci. 1991. V. 43, N 3. P. 425–440.
9. **Sahni S., Gonzales T. P.** P-complete approximation problem // J. Assoc. Comput. Mach. 1976. V. 23, N 3. P. 555–565.
10. **Gutin G., Punnen A. P.** (Eds.) The traveling salesman problem and its variations. Dordrecht; Boston; London: Kluwer Acad. Publ., 2002.

Адрес авторов:

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

Статья поступила

3 февраля 2003 г.,  
переработанный вариант —  
31 марта 2003 г.