УДК 519.168

**А.Н. ТРОФИМЧУК, В.А. ВАСЯНИН**

# ЗАДАЧА ВЫБОРА ПРОПУСКНЫХ СПОСОБНОСТЕЙ ДУГ С ОГРАНИЧЕНИЕМ НА ВРЕМЯ ЗАДЕРЖКИ ПОТОКОВ

**Аннотация.** Рассмотрена задача выбора пропускных способностей дуг из заданного набора, актуальная при распределении потоков в многопродуктовых коммуникационных сетях с ограничением на время задержки потоков. Доказано, что такая задача является NP-трудной. Приведены алгоритмы приближенного решения задачи и результаты их экспериментального сравнения с точным переборным алгоритмом на основе генерации последовательности двоично-отраженных кодов Грея. Отмечено, что получение точного решения возможно с использованием псевдополиномиальных алгоритмов для 0–1 задачи о ранце с мультивыбором.

**Ключевые слова:** потоки в сетях, время задержки потоков, задачи комбинаторной оптимизации.

## ВВЕДЕНИЕ

При проектировании и анализе функционирования сетей передачи данных и транспортных сетей часто необходимо решать задачу выбора пропускных способностей дуг (ЗВПС) из заданного набора дискретных целочисленных значений при ограничении на максимальное время задержки потоков. Задержки потоков $t_{kl}$ на дугах определяются как $t_{kl} = f_{kl} / (w_{kl} - f_{kl})$, $kl \in E$, а ограничение на время $t_{av}$ задержки потоков в сети имеет вид $t_{av} = 1/U_{\Sigma} \sum_{kl \in E} f_{kl} / (w_{kl} - f_{kl}) \le T_{\max}$. Здесь $f_{kl} \in Z^+$ — фиксированное значение потока по дуге $kl \in E$, $E$ — множество дуг сети, $w_{kl} \in Z^+$ — пропускная способность дуги $kl \in E$, $T_{\max}$ — максимальное время задержки потоков в сети, $U_{\Sigma} = \sum_{ij \in S} u_{ij}$ — суммарный поток в сети, $u_{ij} \in Z^+$ — величина потока из узла $i$ в узел $j$, $S$ — множество пар индексов корреспондирующихся узлов в сети. При приближении величины потока на дугах к их пропускным способностям задержки увеличиваются и, следовательно, могут возникать перегрузки в сети.

Для проектирования сетей передачи данных ЗВПС подробно исследовалась в [1–4]. Суть задачи заключается в том, что при фиксированных потоках требуется так выбрать пропускные способности дуг из заданного набора целых чисел, чтобы выполнялось ограничение на время задержки потоков и достигался минимум некоторой целевой функции. Такая задача возникает и в транспортных сетях при распределении потоков по критерию минимума стоимости сети и заданном ограничении на время задержки потоков [5]. Управляя параметром $T_{\max}$ для максимальной задержки, администратор сети передачи данных или диспетчер транспортной сети может обеспечить необходимый ему резерв пропускной способности каналов связи или грузоподъемности транспортных средств при прогнозируемых колебаниях величины потоков на заданных промежутках времени. Уменьшение параметра $T_{\max}$ (увеличение резерва) приводит к удорожанию сети, но уменьшает вероятность перераспределения потоков и технического переоснащения каналов связи или парка транспортных средств при увеличении потоков и угрозе возникновения перегрузок в сети. Увеличение параметра $T_{\max}$ дает возможность уменьшить пропускную способность каналов связи или грузоподъемность транспортных средств и стоимость сети, но увеличивает риск перераспределения потоков и вероятность модернизации сети.

Актуальными являются вопросы принадлежности ЗВПС к классу NP-трудных и разработки приближенных полиномиальных по времени алгоритмов ее решения. Кроме того, представляет интерес определение границ размерности сети, для которых возможно практическое использование алгоритмов полного перебора для получения точного решения задачи.

**МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ЗАДАЧИ**

Пусть задана связная ориентированная сеть $G(N, E)$ с множеством узлов $N$, $n = |N|$, и множеством дуг $E$, $e = |E|$, где $n$ и $e$ соответственно количество узлов и дуг сети. Будем считать, что сеть такова, что для каждой прямой дуги $kl$ ($k < l$) существует обратная $lk$ ($l > k$). Для сети передачи данных дуга представляет коммутированную линию связи, состоящую из одного элементарного канала или пучка таких каналов. Для транспортной сети дуга отождествляется с маршрутом транспортного средства, концевые узлы которого совпадают с начальным и конечным узлами дуги. Сеть может содержать петли и параллельные дуги, поскольку допускаются циклические и повторяющиеся маршруты, а также маршруты с одинаковыми концевыми узлами. На сети задана целочисленная матрица потоков $U = ||u_{ij}||_{n \times n}$, где $u_{ij}$ — величина потока (количество единиц потока) из узла $i$ в узел $j$ в некоторых транспортных блоках заданного размера. Пусть $w_{kl}$, $kl \in E$, — искомые пропускные способности дуг сети в транспортных блоках, $w_{kl} \in \{w_1, w_2, \ldots, w_\alpha\}$, $w_i$, $i = \overline{1, \alpha}$, — упорядоченные по возрастанию целые положительные числа; $d_{kl}$, $kl \in E$, — длины дуг; $C_{kl}(w_{kl}, d_{kl}) \in R^+$, $kl \in E$, — дискретные стоимости дуг, такие, что $C_{kl}(w_i, d_{kl}) \leq$ $\leq C_{kl}(w_{i+1}, d_{kl})$, $i = \overline{1, \alpha - 1}$; $f_{kl} = \sum\limits_{ij \in S} u_{ij}^{kl}$, $kl \in E$, — фиксированные суммарные потоки в транспортных блоках, протекающие по дугам сети, где $u_{ij}^{kl}$ — поток транспортных блоков из $i$ в $j$, проходящий по дуге $kl$.

Требуется найти минимальное значение функции стоимости сети

$$\min_{w_{kl}} \sum_{kl \in E} C_{kl}(w_{kl}, d_{kl}), \; w_{kl} \in \{w_1, w_2, \ldots, w_\alpha\} \tag{1}$$

при ограничении

$$\frac{1}{U_\Sigma} \sum_{kl \in E} \frac{f_{kl}}{w_{kl} - f_{kl}} \leq T_{\max}, \; w_{kl} > f_{kl}, \; kl \in E. \tag{2}$$

Задачу (1), (2) представим в виде задачи о ранце с булевыми переменными и мультивыбором (0–1 Multiple-choice Knapsack Problem, 0–1 MCKP), которая, как известно, принадлежит к классу NP-трудных задач [6]. Пусть $c_{ij} \in R^+$ — дискретные стоимости дуг $i$ с пропускной способностью $w_{ij} \in \{w_1, w_2, \ldots, w_\alpha\} \in Z^+$ и длиной $d_i$, $j = \overline{1, \alpha}$, $i = \overline{1, e}$; $t_{ij} = f_i / (w_{ij} - f_i)$, $w_{ij} > f_i$, $j = \overline{1, \alpha}$, $i = \overline{1, e}$, — задержки потоков на дугах; $f_i$ — поток по дуге $i$, $i = \overline{1, e}$. Положим $x_{ij} = 1$, если для дуги $i$ выбрана пропускная способность $w_{ij}$, $j = \overline{1, \alpha}$, $i = \overline{1, e}$, и $x_{ij} = 0$ в противном случае.

Требуется найти

$$\min \sum_{i=1}^{e} \sum_{j=1}^{\alpha} c_{ij} x_{ij} \tag{3}$$

при ограничениях

$$\frac{1}{U_\Sigma} \sum_{i=1}^{e} \sum_{j=1}^{\alpha} t_{ij} x_{ij} \leq T_{\max}, \tag{4}$$

$$\sum_{j=1}^{\alpha} x_{ij} = 1, \; i = \overline{1, e}, \tag{5}$$

$$x_{ij} \in \{0, 1\}. \tag{6}$$

Здесь искомые пропускные способностив $w_{ij}$ соответствуют $x_{ij}^*$ — оптимальному решению задачи (3)–(6).

Нетрудно видеть, что любую индивидуальную ЗВПС, сформулированную в виде (1), (2), можно за время $O(e\alpha)$ преобразовать в соответствующий экземпляр задачи (3)–(6). Для этого необходимо построить две матрицы размером $e\times\alpha$, строки которых соответствуют дугам, столбцы — набору дискретных пропускных способностей, а в качестве элементов матриц принимаются стоимости дуг $c_{ij}$ и задержки на дугах $t_{ij}$. Справедливо и обратное преобразование.

**Утверждение 1.** Оптимизационная ЗВПС, сформулированная в виде (1), (2), является NP-трудной.

**Доказательство.** В [6] рассматривается NP-полная задача выбора пропускных способностей, сформулированная в следующем виде распознавания. Задано множество дуг $E$, множество пропускных способностей $W\subseteq Z^+$, функция стоимости $c:E\times W\to Z^+$, функция штрафа за задержку $t:E\times W\to Z^+$, такие, что для всех упорядоченных по неубыванию пропускных способностей $w_i\in\in\{w_1,w_2,\dots,w_\alpha\}\in W$ и $i<j$ выполняются неравенства $c_l(w_i)\le c_l(w_j)$ и $t_l(w_i)\ge t_l(w_j)$, $l\in E$. В задаче ставится вопрос, существует ли такой выбор пропускных способностей $w_i$, что общая стоимость $\sum_{l\in E}c_l\ (w)\le C$, а общий штраф за задержку $\sum_{l\in E}t_l\ (w)\le T$, где $C$ и $T$ — заданные положительные целые числа.

Задача (1), (2), сформулированная как задача распознавания, принадлежит к классу NP, так как очевидно, что для нее существует сертификат (удостоверение) длиной $e$, представляющий собой набор оптимальных пропускных способностей дуг, который может быть проверен за линейное время. В свою очередь, любую индивидуальную задачу распознавания, сформулированную в [6], можно за полиномиальное время преобразовать (по Карпу [7]) в задачу распознавания (1), (2), что и доказывает NP-полноту ЗВПС в виде распознавания и ее NP-трудность как оптимизационной задачи. Другое доказательство NP-полноты и NP-трудности ЗВПС следует из полиномиального преобразования задачи, сформулированной в виде (3)–(6), в задачу (1), (2).

### АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧИ

Рассмотрим два приближенных алгоритма решения ЗВПС. Первый алгоритм (А) основан на работах [1–3], в которых предложено для решения ЗВПС в виде (1), (2) аппроксимировать дискретные стоимости дуг $C_{kl}(w_{kl},d_{kl})$ непрерывными линейными функциями

$$C_{kl}(w_{kl},d_{kl})=c_{kl}^0+c_{kl}^1 w_{kl}, \qquad (7)$$

где $c_{kl}^0$, $c_{kl}^1$ — полученные коэффициенты аппроксимации. Затем, используя метод множителей Лагранжа, следует аналитически решить релаксированную задачу и получить значения оптимальных пропускных способностей $w_{kl}^*$ [2]:

$$w_{kl}^*=f_{kl}+\frac{f_{kl}}{U_\Sigma T_{\max}}\frac{\sum\limits_{rs\in E}\sqrt{c_{rs}^1 f_{rs}}}{\sqrt{c_{kl}^1 f_{kl}}}. \qquad (8)$$

Алгоритм позволяет быстро найти окрестности непрерывных точек оптимума $w_{kl}^*$ и приближенное дискретное решение. Получение точного решения в окрестностях точек оптимума требует полного перебора. Алгоритм применим, когда известно, что заданные дискретные стоимости дуг можно с достаточной степенью адекватности аппроксимировать линейными функциями.

Второй алгоритм основан на общей схеме метода последовательного анализа и отсеивания вариантов (ПАВ), который был предложен в [8] и получил дальнейшее развитие в работах [9–11]. Его можно применять как к исходной постановке задачи, так и постановке в виде (3)–(6). Алгоритм перебирает решения, сужая на каждой итерации область допустимых решений, и может использоваться для любых монотонно неубывающих стоимостей дуг при увеличении их пропускных способностей. Для получения точного решения также необходимо выполнить полный перебор допустимых вариантов на заключительном этапе работы алгоритма.

Отметим, что на практике для сетей передачи данных и транспортных сетей пропускная способность дуг, как правило, должна быть одинаковой для прямого $kl$ и обратного $lk$ направлений. В целях снижения размерности задачи две ориентированные дуги, $kl$ и $lk$, заменяются одной неориентированной дугой $kl$ ($k < l$) и выбирается $f_{kl} = \max\{f_{kl}, f_{lk}\}$.

**Алгоритм А.** Идея алгоритма заключается в следующем. Предположим, что для всех дуг $kl \in E$ известны коэффициенты $c_{kl}^0$, $c_{kl}^1$ линейной зависимости (7). Такие коэффициенты можно получить для каждой дуги $kl$ длиной $d_{kl}$ линейной аппроксимацией (например, методом наименьших квадратов) дискретных стоимостей для ряда стандартных дискретных пропускных способностей $w_{kl} \in \{w_1, w_2, \dots, w_\alpha\}$, одинаковых для всех дуг. Зная коэффициенты $c_{kl}^1 = tg\, \varphi$ (рис. 1) и значения $f_{kl}$, по формуле (8) можно найти пропускные способности $w_{kl}^*$. Далее в окрестности непрерывных оптимумов $w_{kl}^*$ по определенной процедуре выбираем подходящие значения пропускных способностей из дискретного ряда.

Алгоритм включает следующие шаги.

1. Для каждой дуги методом наименьших квадратов определяем коэффициенты $c_{kl}^1$. Вычисляем $w_{kl}^*$, $kl \in E$, согласно (8).

2. Из $w_{kl} \in \{w_1, w_2, \dots, w_\alpha\}$ для каждой дуги $kl \in E$ выбираем ближайшие к $w_{kl}^*$ допустимые значения $w_{kl}^j$, $j = \overline{1, \alpha}$, такие, что $f_{kl} < w_{kl}^j \le w_{kl}^*$. Если $w_{kl}^* > w_\alpha$, то набор $\{w_1, w_2, \dots, w_\alpha\}$ задан некорректно и его необходимо расширить до значения $w_\alpha = \lceil w_{kl}^* \rceil$, где $\lceil \cdot \rceil$ — знаки округления до большего целого. В случае если $w_{kl}^j \le f_{kl}$, то в качестве $w_{kl}^j$ выбираем ближайшее большее значение $w_{kl}^j > f_{kl}$ (может оказаться, что $w_{kl}^j \ge w_{kl}^*$). Если $f_{kl} = 0$, то принимается, что дуги $kl$ не существует.

3. В окрестности точки $w_{kl}^*$ для каждой дуги найдем значения $c_{kl}^* = \Delta c_{kl} / \Delta t_{kl}$, где $\Delta c_{kl} = c_{kl}(w_{kl}^{j+1}) - c_{kl}(w_{kl}^j)$, $\Delta t_{kl} = f_{kl} / (w_{kl}^j - f_{kl}) - f_{kl} / (w_{kl}^{j+1} - f_{kl})$, $j \in \{1, \dots, \alpha - 1\}$ (рис. 1).

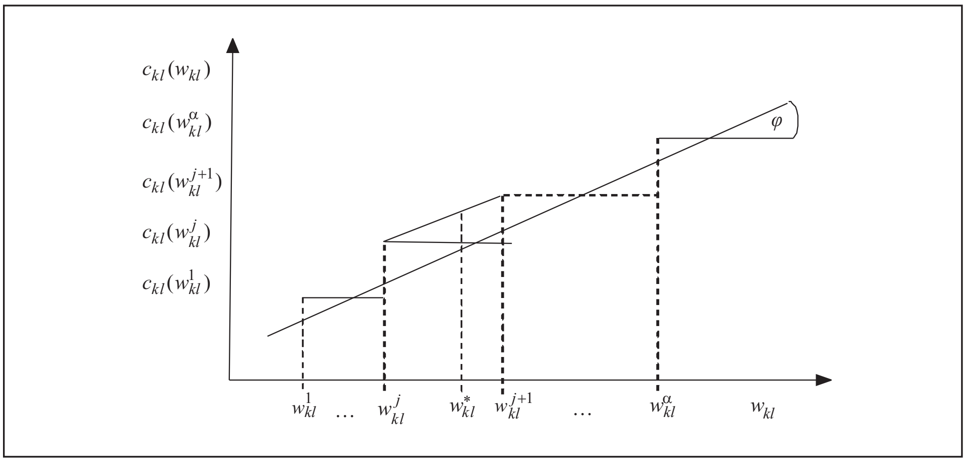


*Рис. 1.* Функция стоимости дуг

4. Упорядочим все дуги $kl \in E$ в порядке возрастания значений $c_{kl}^*$ и получим множество $E^* = \{(k,l)_1, (k,l)_2, \ldots, (k,l)_e\}$. Основанием для такого упорядочения является то, что для всех дуг $c_{kl}(w_{kl}^j) \le c_{kl}(w_{kl}^{j+1})$, а $t_{kl}(w_{kl}^j) = f_{kl} / (w_{kl}^j - f_{kl}) >$ $> t_{kl}(w_{kl}^{j+1}) = f_{kl} / (w_{kl}^{j+1} - f_{kl})$.

Установим начальное значение счетчика дуг $i = 0$.

5. Полагаем $i \leftarrow i+1$. Если $i \le e$, то выбираем дугу $(k,l)_i$ из множества $E^*$ и переходим к шагу 6. Иначе переходим к шагу 7.

6. Увеличиваем пропускную способность $w_{(k,l)_i}^j$ дуги $(k,l)_i$ до ближайшего большего значения из дискретного ряда $\{w_1, w_2, \ldots, w_\alpha\}$, т.е. выбираем $w_{(k,l)_i}^{j+1}$ такую, что $w_{(k,l)_i}^{j+1} > w_{kl}^* \ge w_{(k,l)_i}^j$, $j \in \{1, \ldots, \alpha-1\}$. Пересчитываем значение $t_{av}$ с учетом увеличения пропускной способности дуги $(k,l)_i$. Если $t_{av} \le T_{\max}$, то переходим к шагу 7. Иначе переходим к шагу 5.

7. Конец работы алгоритма. Найденные значения $w_{(k,l)_i}^j$, $j \in \{1, \ldots, \alpha\}$, $i \in \{1, \ldots, e\}$, являются приближенным решением задачи.

Временная сложность алгоритма составляет $O(e\alpha^2 + e\alpha + e \log e + K_1 e)$, $K_1$ — константа, и в основном определяется временной сложностью аппроксимации стоимостей дуг линейными функциями и выбором алгоритма сортировки.

**Алгоритм ПАВ.** Приведем общую схему метода ПАВ для решения ЗВПС, сформулированной в виде (1), (2). Метод состоит из процедур отсеивания $W_1$ и $W_2$.

**Процедура $W_1$ — отсеивание по ограничению.** 1. Пусть $\lambda$, $\lambda = 1, 2, \ldots$, — номер подмножества выбранных пропускных способностей для $kl \in E$, а $m$, $m = 1, 2, \ldots, \alpha$, — индекс упорядоченных по возрастанию дискретных пропускных способностей из заданного ряда $\{w_1, w_2, \ldots, w_\alpha\}$. Определим диапазон возможных значений пропускных способностей $W_{kl}(\lambda) = [\min w_{kl}^m, \max w_{kl}^m] \in$ $\in \{w_1, w_2, \ldots, w_\alpha\}$ по каждой дуге $kl \in E$, используя неравенство $t_{av} \le T_{\max}$. При этом должно выполняться $\min w_{kl}^m \ge w_1$, $\max w_{kl}^m \le w_\alpha$ и хотя бы одно из этих неравенств строгое. Из множества $\{w_1, w_2, \ldots, w_\alpha\}$ отсеиваются те значения, которые заведомо не удовлетворяют условию $t_{av} \le T_{\max}$. Первоначально для $\lambda = 1$ и $kl \in E$ выбираем $\min w_{kl}^1 = w_1$, $\max w_{kl}^\alpha = w_\alpha$, т.е. $W_{kl}(1) = [\min w_{kl}^1, \max w_{kl}^\alpha]$.

2. Для отсеивания используем условие

$$a = \frac{f_{kl}}{U_\Sigma (\min w_{kl}^m - f_{kl})} > b = T_{\max} - \frac{1}{U_\Sigma} \sum_{rs \in E,\, rs \ne kl} \frac{f_{rs}}{\max w_{rs}^{m-1} - f_{rs}}. \tag{9}$$

Последовательно просматриваем слева направо заданный набор пропускных способностей $\{w_1, w_2, \ldots, w_\alpha\}$ и выбираем $\min w_{kl}^m = w_1, w_2, \ldots > f_{kl}$ до тех пор, пока выполняется (9). Пусть условие (9) выполняется включительно до $\min w_{kl}^m = w_\beta$ и не выполняется при $\min w_{kl}^m = w_{\beta+1}$, тогда для дуги $kl$ отсеиваются все $w_\xi \le w_\beta$, $\xi = \overline{1, \beta}$, (рис. 2) и $\min w_{kl}^{\beta+1} = w_{\beta+1}$. Продолжая эту процедуру для всех дуг, получаем новые границы изменения пропускных способностей $W_{kl}(2) = [\min w_{kl}^{\beta+1}, \max w_{kl}^\alpha]$, $kl \in E$. Переходим к процедуре $W_2$.

**Процедура $W_2$ — отсеивание по значению целевой функции.** 1. Задаем начальный порог $C_1^* = \frac{1}{2}(C_{\min} + C_{\max})$, где $C_{\min} = \sum_{kl \in E} c_{kl}(\min w_{kl}^m)$, $C_{\max} =$ $= \sum_{kl \in E} c_{kl}(\max w_{kl}^m)$, $m = 1, 2, \ldots, \alpha$.

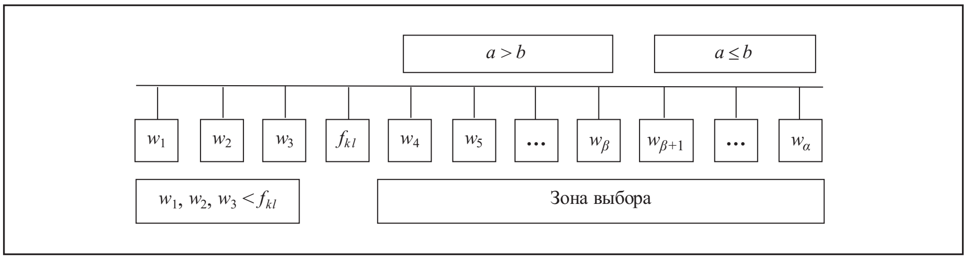


*Рис. 2.* Схема отсеивания вариантов по ограничению

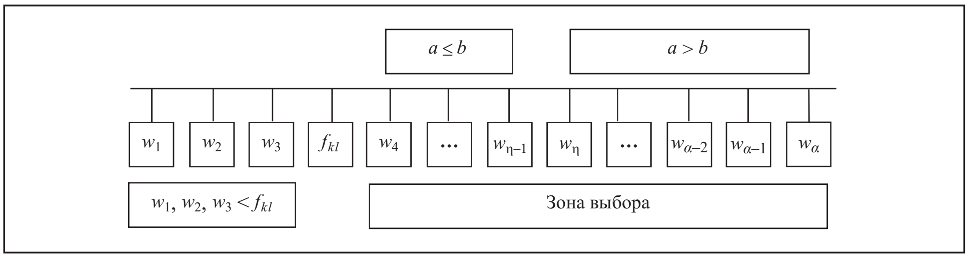


*Рис. 3.* Схема отсеивания вариантов по значению целевой функции

2. Для отсеивания допустимых вариантов используем условие

$$a = c_{kl}(\max w_{kl}^m) > b = C_1^* - \sum_{rs \in E,\, rs \ne kl} c_{rs}(\min w_{kl}^m). \tag{10}$$

Последовательно просматриваем справа налево заданный набор пропускных способностей $\{w_1, w_2, \dots, w_\alpha\}$ и выбираем $\max w_{kl}^m = w_\alpha, \dots, w_2, w_1 > f_{kl}$ до тех пор, пока выполняется (10). Пусть условие (10) выполняется включительно до $\max w_{kl}^m = w_\eta$ и не выполняется при $\max w_{kl}^m = w_{\eta-1}$, тогда для дуги $kl$ отсеиваются все $w_\xi \ge w_\eta$, $\xi = \overline{\eta, \alpha}$, (рис. 3) и $\max w_{kl}^{\eta-1} = w_{\eta-1}$.

После выполнения процедуры отсеивания вариантов для всех дуг по условию (10) возможны следующие случаи.

**Случай 1.** Если отсеивание произошло хотя бы для одной дуги, то получаем новые границы изменения пропускных способностей $W_{kl}(3) = [\min w_{kl}^{\beta+1}, \max w_{kl}^{\eta-1}]$, $kl \in E$. Переходим к процедуре $W_1$.

**Случай 2.** Отсеивания не произошло. Сужаем область поиска, устанавливая новый порог $C_1^* \leftarrow \frac{1}{2}(C_1^* + C_{\min})$, и переходим к процедуре отсеивания по условию (10).

**Случай 3.** Отсеялись все варианты пропускных способностей. Расширяем область поиска, устанавливая новый порог $C_1^* \leftarrow \frac{1}{2}(C_1^* + C_{\max})$, и переходим к процедуре отсеивания по условию (10).

Повторяем процедуры $W_1$ и $W_2$ до тех пор, пока для всех $kl \in E$ границы изменения пропускных способностей $W_{kl}(g) = [\min w_{kl}^\beta, \max w_{kl}^\eta]$ не сократятся до двух значений, т.е. когда $|W_{kl}(g)| = |[\min w_{kl}^\beta, \max w_{kl}^\eta]| = 2$ при $\eta - \beta = 1$ и для каждой дуги можно выбрать либо $\min w_{kl}^\beta$, либо $\max w_{kl}^\eta$. Далее выбор оптимального решения можно выполнить полным перебором, но его сложность экспоненциальная и оценивается как $O(2^e)$. Для получения приближенного решения в алгоритме ПАВ может использоваться такая же, как в алгоритме А, процедура упорядочения дуг во множестве $E^* = \{(k,l)_1, (k,l)_2, \ldots, (k,l)_e\}$ по возрастанию значений $c^*_{kl}$. При этом $c^*_{kl}$ определяются по следующей формуле:

$$c^*_{kl} = \frac{c_{kl}(\max w^{\eta}_{kl}) - c_{kl}(\min w^{\beta}_{kl})}{\dfrac{f_{kl}}{\min w^{\beta}_{kl} - f_{kl}} - \dfrac{f_{kl}}{\max w^{\eta}_{kl} - f_{kl}}}, \; kl \in E.$$

Сходимость алгоритма ПАВ основана на конечности метода [8], а его временная сложность без полного перебора на заключительном этапе зависит от количества итераций по $\lambda$, числа $K$ сужений и расширений области допустимых решений, выбора алгоритма сортировки дуг и составляет $O[\lambda(e(e+\alpha)+Ke(e+\alpha))+ + e\log e + K_1 e]$, где $K_1$ — некоторая константа.

Как было показано, оба алгоритма (А и ПАВ) на заключительной стадии работы сужают область допустимых решений до двух значений: $w_{ij} \in \{w_j, w_{j+1}\}$, $j = \overline{k, \alpha-1}$, $i = \overline{1, e}$, и при полном переборе $2^e$ вариантов позволяют получить точное решение. Запишем сокращенную задачу в виде

$$\min \sum_{i=1}^{e} \sum_{j=1}^{2} c_{ij} x_{ij} \tag{11}$$

при ограничениях

$$\frac{1}{U_\Sigma} \sum_{i=1}^{e} \sum_{j=1}^{2} t_{ij} x_{ij} \le T_{\max}, \tag{12}$$

$$\sum_{j=1}^{2} x_{ij} = 1, \; i = \overline{1, e}, \tag{13}$$

$$x_{ij} \in \{0, 1\}, \tag{14}$$

где $c_{ij}$, $j = \overline{1,2}$, $i = \overline{1,e}$, — дискретные стоимости дуг; $t_{ij} = f_i / (w_{ij} - f_i)$, $j = \overline{1,2}$, $i = \overline{1,e}$, — задержки потоков на дугах; $f_i$ — поток по дуге $i$, $i = \overline{1,e}$.

Задачу (11)–(14) можно преобразовать в 0–1 задачу о ранце (0–1 Knapsack Problem):

$$\min \sum_{i=1}^{e} (c_{i1} + \Delta c_i x_i) \tag{15}$$

при ограничениях

$$\frac{1}{U_\Sigma} \sum_{i=1}^{e} (t_{i1} - \Delta t_i x_i) \le T_{\max}, \tag{16}$$

$$x_i \in \{0, 1\}, \; i = \overline{1, e}, \tag{17}$$

где $\Delta c_i = c_{i2} - c_{i1}$, $t_{i1} = f_i / (w_{i1} - f_i)$, $\Delta t_i = f_i / (w_{i1} - f_i) - f_i / (w_{i2} - f_i)$, $i = \overline{1,e}$, $\dfrac{1}{U_\Sigma} \sum_{i=1}^{e} t_{i1} > T_{\max}$, $\dfrac{1}{U_\Sigma} \sum_{i=1}^{e} t_{i2} \le T_{\max}$.

Для задачи о ранце (15)–(17) и задачи о ранце с мультивыбором (11)–(14) существуют точные псевдополиномиальные алгоритмы и полностью полиномиальные приближенные схемы решения (FPTAS — Fully Polynomial Time Approximation Scheme) [12, 13]. Это означает, что для них имеются алгоритмы, которые за полиномиальное время от размера входа задач и $1/\varepsilon$ позволяют получить $(1+\varepsilon)$ гарантированное приближенное решение, где $\varepsilon$ — сколь угодно малое положительное число. Поэтому на заключительном этапе решения задачи в обоих алгоритмах для получения точного или гарантированного $\varepsilon$-приближенного решения возможно применение алгоритмов, описанных в работах [12–17]. Данные алгоритмы, в частности алгоритм ветвей и границ, реализованный в пакете lp_ solve 5.5.2.5, также можно использовать для решения задачи в постановке (3)–(6).

Авторам не известно о существовании на общедоступных серверах эталонных примеров с наборами данных различной размерности и индивидуальной особенностью для задачи выбора пропускной способности. В связи с этим проверка алгоритмов осуществлялась на примерах, сгенерированных датчиком псевдослучайных чисел. Сравнение приближенных и точных решений, полученных алгоритмами А и ПАВ, проводилось для линейных и нелинейных функций стоимости. Поскольку при проведении эксперимента не использовались пакеты программ целочисленной оптимизации, а выполнялся полный перебор допустимых решений, были сгенерированы сети малой размерности для: $n = 4$ и степени узлов $\mathrm{val}=3$; $n=5$, $\mathrm{val}=4$; $n=10$, $\mathrm{val}=3$; $n=10$, $\mathrm{val}=4$. Для полного перебора вариантов решения задачи (11)–(14) или (15)–(17) в обоих приближенных алгоритмах использовался алгоритм полного перебора (АПП), основанный на алгоритме генерации последовательности двоично-отраженных $e$-разрядных кодов Грея, предложенном в [18, 19]. Алгоритм полного перебора дает возможность в процессе решения эффективно вычислять значения целевой функции (11) или (15) и ограничения (12) или (16). Длины дуг $d_{kl}$, $kl \in E$, варьировались в пределах от 50 до 100 км, а потоки $u_{ij}$, $i, j = \overline{1, n}$, — от одного до пяти транспортных блоков. Значения $f_{kl}$, $kl \in E$, были получены при распределении всех потоков по кратчайшим путям. Пропускные способности дуг задавались набором $w_{kl} \in \{5, 10, 15, 20, \ldots\}$. Стоимости дуг для заданного набора пропускных способностей вычислялись по формуле $C_{kl}(w_{kl}, d_{kl}) = k_0^j + k_1^j d_{kl}$, $j = 1, 2, \ldots$, $kl \in E$. Для линейных функций коэффициенты $k_0^1 \le k_0^2 \le \ldots$ и $k_1^1 \le k_1^2 \le \ldots$ выбирались из наборов $\{0, 0, 0, 0, \ldots\}$ и $\{10, 20, 30, 40, \ldots\}$. Для нелинейных функций значения $k_0^1 = 5$, $k_1^1 = 10$, а значения последующих $k_0^j$ и $k_1^j$, $j = 2, 3, \ldots$, возрастали на случайную величину из интервала [5, 10]. В табл. 1 приведены результаты решения задачи при ограничении на задержку $T_{\max} = 0{,}05$. Для всех решений показаны полученные значения $t_{av}$, для приближенных решений в скобках приведены отклонения (в процентах) от оптимального решения, а для точных решений указано отклонение от лучшего решения для алгоритмов А и ПАВ; лучшие решения выделены жирным шрифтом. Время счета задачи (в секундах), округленное до двух знаков после запятой, показано только для точного решения, полученного АПП.

Как и предполагалось, для линейных функций приближенные и точные решения для алгоритмов А и ПАВ совпадают. Приближенные значения целевой функции для этих алгоритмов отличаются от оптимальных в пределах 0,44%.

Для нелинейных функций приближенные значения целевой функции для алгоритмов А и ПАВ отличаются от оптимальных в пределах 0,35 и 2% соответственно. Точные решения алгоритма ПАВ лучше результатов алгоритма А (на 2,74, 3,96, 8,41 и 12,7%), что объясняется погрешностью аппроксимации нелинейных функций стоимости линейными.

Результаты эксперимента показывают, что оба алгоритма без полного перебора вариантов на заключительном этапе не дают точного решения. Алгоритм полного перебора целесообразно применять, когда число дуг не превышает 30–35, а время работы алгоритма генерации последовательности кодов Грея изменяется от 0 до 146,06 с ($e \le 35$, рис. 4). Как видно из рисунка, начиная с числа битов 36 и более время работы данного алгоритма резко возрастает. Однако АПП может эффективно использоваться в различных схемах ветвления, например в методах ветвей и границ, когда часть переменных фиксируется, а число свободных переменных в узлах ветвления менее 35.

**Т а б л и ц а 1.** Результаты экспериментального исследования алгоритмов А и ПАВ

| **Функция** | **Алгоритм, время счета, с** | **Значение целевой функции (в усл. ед.) при** | | | |
|---|---|---|---|---|---|
| | | $n = 4$, val = 3, $e = 12$ | $n = 5$, val = 4, $e = 20$ | $n = 10$, val = 3, $e = 30$ | $n = 10$, val = 4, $e = 40$ |
| Линейная | А (приближенное решение) | 37070 (0%) 0,04997977 | 61550 (0,44%) 0,04963951 | 186980 (0,27%) 0,04979776 | 217450 (0,18%) 0,04986274 |
| | А (точное решение) | **37070** 0,04997927 | **61280** 0,04995353 | **186480** 0,04999647 | **217070** 0,04999137 |
| | Время счета, с | 0,00 | 0,02 | 11,90 | 12163,90 |
| | ПАВ (приближенное решение) | 37070 (0%) 0,04997977 | 61550 (0,44%) 0,04963951 | 186980 (0,27%) 0,04979776 | 217450 (0,18%) 0,04986274 |
| | ПАВ (точное решение) | **37070** 0,04997927 | **61280** 0,04995353 | **186480** 0,04999647 | **217070** 0,04999137 |
| | Время счета, с | 0,00 | 0,02 | 10,19 | 13091,23 |
| Нелинейная | А (приближенное решение) | 32243,02 (0,18%) 0,04972611 | 42867,63 (0,35%) 0,04963951 | 139395,03(0,31%) 0,04974249 | 162542,47(0,31%) 0,04976205 |
| | А (точное решение) | 32184,18 (2,74%) 0,04999059 | 42719,81(3,96%) 0,04995353 | 138966,34 (8,41%) 0,04999217 | 162035,20 (12,7%) 0,04997742 |
| | Время счета, с | 0,00 | 0,02 | 11,72 | 11786,80 |
| | ПАВ (приближенное решение) | 31819,73 (1,58%) 0,04978262 | 41091,21 (0%) 0,04987555 | 128182,71 (0%) 0,04998188 | 143902,47 (0,09%) 0,04994724 |
| | ПАВ (точное решение) | **31324,34** 0,04990151 | **41091,21** 0,04987555 | **128182,71** 0,04998188 | **143770,94** 0,04999994 |
| | Время счета, с | 0,00 | 0,02 | 10,22 | 13160,57 |

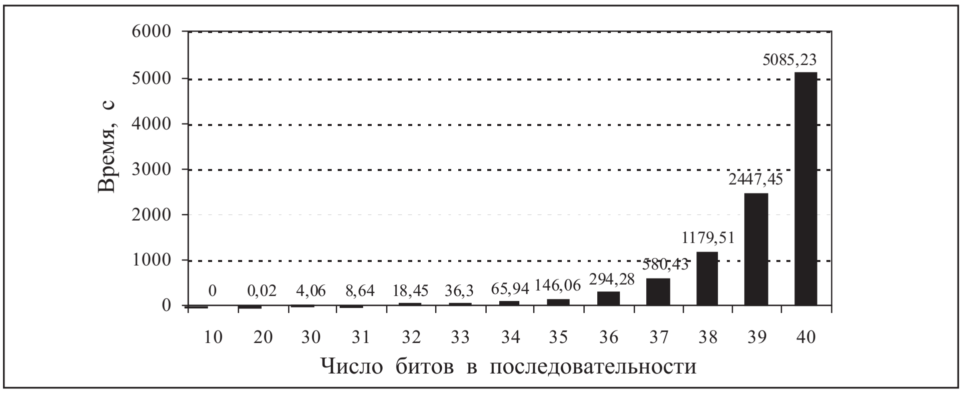


*Рис. 4.* График зависимости времени работы алгоритма генерации последовательности кодов Грея от числа битов в последовательности

Приближенные жадные алгоритмы А и ПАВ имеют полиномиальную оценку временной сложности и не дают больших погрешностей при незначительном разбросе значений $c_{kl}$ и $t_{kl}$, $kl \in E$, что в большинстве случаев позволяет их использовать для инженерных расчетов на сетях, содержащих более 1000 узлов и 3000 дуг. В этом отношении интересна работа [20], в которой рассматривался формальный подход к проблеме представления точной характеристики наборов весов $t_{kl}$ для 0–1 задачи о рюкзаке, когда множество возможных решений образуют матроид. Было показано, что в этом случае жадные алгоритмы позволяют получить точное решение для произвольных значений $c_{kl}$, но условия на разброс весов $t_{kl}$ (которые можно проверить за полиномиальное время) сильно ограничены.

Результаты эксперимента получены на двуядерном ПК с тактовой частотой 2.66 GHz и оперативной памятью 2 Гб под управлением ОС Windows XP. Все программы написаны на языке Фортран в среде Microsoft Developer Visual Studio.

**ЗАКЛЮЧЕНИЕ**

В работе доказано, что задача оптимизации выбора пропускных способностей дуг является NP-трудной. Показано, что данная задача в исходной постановке за полиномиальное время преобразуется в 0–1 задачу о ранце с мультивыбором. Для ее решения предложено два приближенных алгоритма, основанных на аппроксимации дискретных функций стоимости линейными и методе ПАВ. Проведены экспериментальные исследования решения задачи, которые показали, что погрешность приближенных алгоритмов существенно зависит от разброса значений вычисляемых параметров $c_{kl}$ и $t_{kl}$, $kl \in E$. Для решения задачи в ранцевой постановке могут применяться известные точные псевдополиномиальные алгоритмы и приближенные схемы FPTAS, реализованные в современных пакетах целочисленного программирования.

СПИСОК ЛИТЕРАТУРЫ


1. Kleinrock L. Queueing systems. Vol. II: Computer applications. New York: John Wiley & Sons, 1976. 549 p.
2. Bertsekas D., Gallager R. Data networks (2nd Ed.). Englewood Cliffs (N.J.): Prentice-Hall, Inc., 1992. 556 p.
3. Зайченко Ю.П. Задача проектирования структуры распределенных вычислительных сетей. *Автоматика*. 1981. № 4. С. 35–44.
4. Зайченко Е.Ю. Комплекс моделей и алгоритмов оптимизации характеристик сетей с технологией MPLS. *Системні дослідження та інформаційні технології*. 2007. № 4. С. 58–71.
5. Trofymchuk O.M., Vasyanin V.A. Simulation of packing, distribution and routing of small-size discrete flows in a multicommodity network. *Journal of Automation and Information Sciences*. 2015. Vol. 47, Iss. 7. P. 15–30.
6. Garey M.R., Johnson D.S. Computers and intractability: A guide to the theory of NP-completeness. New York: W.H. Freeman & Co., 1979. 338 p.
7. Karp R.M. Reducibility among combinatorial problems. In: *Complexity of computer computations*. Miller R.E., Thatcher J.W. (Eds.). *Proceedings of a symposium on the Complexity of Computer Computations*. IBM Thomas J. Watson Research Center, Yorktown Heights. New York: Plenum Press, 1972. P. 85–103.
8. Михалевич В.С., Шор Н.З. Численные решения многовариантных задач по методу последовательного анализа вариантов. *Научно-методические материалы экономико-математического семинара*. Москва, 1962. Вып. 1. С. 15–42. (Ротапринт АН СССР; ЛЭМИ.)
9. Михалевич В.С. Последовательные алгоритмы оптимизации и их применение. I. *Кибернетика*. 1965. № 1. С. 45–55; II. *Там же*. № 2. С. 85–89.
10. Волкович В.Л., Волошин А.Ф. Об одной общей схеме последовательного анализа и отсеивания вариантов. *Кибернетика*. 1978. № 5. С. 98–105.
11. Михалевич В.С., Волкович В.Л., Волошин А.Ф., Поздняков Ю.М. Алгоритмы последовательного анализа и отсеивания вариантов в задачах дискретной оптимизации. *Кибернетика*. 1980. № 3. С. 76–85.
12. Martelo S., Toth P. Knapsack problems: algorithms and computer implementations. Chichester: Wiley, 1990. 296 p.
13. Kellerer H., Pferschy U., Pisinger D. Knapsack problems. Berlin; Heidelberg: Springer-Verlag, 2004. 548 p.

14. Bansal M.S., Venkaiah V.Ch. Improved fully polynomial time approximation scheme for the 0–1 multiple-choice Knapsack pproblem. Technical Report Number: IIIT-H/TR/2004/003, IIIT, Hyderabad, India. 2004. 10 p.

15. Suri B., Bordoloi U.D., Eles P. A scalable GPU-based approach to accelerate the multiple-choice knapsack problem. *Design, Automation & Test in Europe Conference & Exhibition* (*DATE*), 12–16 March 2012. Dresden: IEEE, 2012. 4 p. DOI: https://doi.org/10.1109/DATE.2012.6176665.

16. Rhee D. Faster fully polynomial approximation schemes for Knapsack problems. Boston: Massachusetts Institute of Technology. Operations Research Center. 2015. 62 p. URL: http://hdl.handle.net/1721.1/98564.

17. Bednarczuk E.M., Miroforidis J., Pyzel P. A multi-criteria approach to approximate solution of multiple-choice knapsack problem. *Computational Optimization and Applications*. 2018. Vol. 70, Iss. 3. P. 889–910.

18. Bitner J.R., Ehrlich G., Reingold E.M. Efficient generation of the binary reflected Gray code and its applications. *Comm. ACM*. 1976. Vol. 19, Iss. 9. P. 517–521.

19. Knuth D.E. The art of computer programming. Vol. 4A. Combinatorial algorithms. Part 1. Boston: Addison Wesley Longman, 2011. 933 p.

20. Cerdeira J.O., Barcia P. When is a 0–1 knapsack a matroid? *Portugaliae Mathematica*. 1995. N 52. P. 475–480.

*Надійшла до редакції 30.10.2018*

**О.М. Трофимчук, В.О. Васянін**

**ЗАДАЧА ВИБОРУ ПРОПУСКНИХ СПРОМОЖНОСТЕЙ ДУГ З ОБМЕЖЕННЯМ НА ЧАС ЗАТРИМКИ ПОТОКІВ**

**Анотація.** Розглянуто задачу вибору пропускних спроможностей дуг із заданого набору, актуальну для розподілу потоків в багатопродуктових комунікаційних мережах з обмеженням на час затримки потоків. Доведено, що така задача є NP-складною. Наведено алгоритми наближеного розв'язання задачі та результати їхнього експериментального порівняння з точним переборним алгоритмом на основі генерації послідовності двійково-відображених кодів Грея. Відзначено, що отримання точного розв'язку можливо з використанням псевдополіноміальних алгоритмів для 0–1 задачі про ранець з мультивибором.

**Ключові слова:** потоки у мережах, час затримки потоків, задачі комбінаторної оптимізації.

**O.M. Trofymchuk, V.A. Vasyanin**

**CHOOSING THE CAPACITY OF ARCS WITH CONSTRAINT ON FLOW DELAY TIME**

**Abstract.** The authors consider the problem of choosing the capacity arcs from a given set, which is important in flow distribution in multicommodity communication networks with constraint on flow delay time. It is proved that such problem is NP-hard. The algorithms for the approximate solution of the problem and results of heir experimental comparison with exact algorithm based on generating a sequence of binary reflected Gray codes are given. It is noted that obtaining an exact solution is possible with the use of pseudopolynomial algorithms for the 0–1 Multiple-choice Knapsack Problem.

**Keywords:** flows in networks, flow delay time, combinatorial optimization problems.

**Трофимчук Александр Николаевич,**
чл.-кор. НАН Украины, профессор, доктор техн. наук, директор Института телекоммуникаций и глобального информационного пространства НАН Украины, Киев, e-mail: itelua@kv.ukrtel.net.

**Васянин Владимир Александрович,**
доктор техн. наук, ведущий научный сотрудник Института телекоммуникаций и глобального информационного пространства НАН Украины, Киев, e-mail: archukr@meta.ua.