

**ПРИМЕНЕНИЕ МЕТОДА ИЗМЕНЯЮЩИХСЯ ВЕРОЯТНОСТЕЙ  
ДЛЯ ЗАДАЧ ОПТИМАЛЬНОГО РАЗМЕЩЕНИЯ НА СЕТИ**

А. Н. Антамошкин, Л. А. Казаковцев

Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева  
Российская Федерация, 660014, г. Красноярск, просп. им. газ. «Красноярский рабочий», 31  
E-mail: levk@bk.ru

*Рассматривается  $p$ -медианная задача оптимального размещения на графе (сети), являющаяся популярной моделью как в логистических задачах (оптимальное размещение транспортной, телекоммуникационной и иной инфраструктуры), так и опосредованно в задачах кластерного анализа, распознавания образов, в теории оценивания. Цель  $p$ -медианной задачи на графе (сети) заключается в нахождении определенного числа вершин (узлов) таких, чтобы суммарное расстояние от всех узлов сети до ближайшей из выбранных вершин (узлов) было минимальным. Для приближенного решения рассматриваемой неразрешимой (в общем случае) за полиномиальное время задачи предложен эвристический алгоритм, основанный на идеях метода изменяющихся вероятностей, являющегося методом случайного поиска с адаптацией. Идея алгоритма заключается в случайном выборе вершин в соответствии с заданными вероятностями выбора каждой из вершин и изменении этих вероятностей для каждой из выбранных вершин и вершин в некоторой ее окрестности в зависимости от достигнутого значения целевой функции (т. е. в зависимости от суммарного расстояния от каждой из вершин графа до ближайшей из выбранных вершин). При создании алгоритма для  $p$ -медианных задач с большой размерностью и вычислительной сложностью ставится задача оптимального использования вычислительных мощностей. Эффективность алгоритма подтверждена экспериментальной проверкой на тестовых примерах из библиотеки ORLIB. Также приведены результаты работы комбинированного алгоритма – комбинации предложенного алгоритма с процедурой локального поиска для решения данной задачи. Экспериментально показано, что результаты такой комбинации превосходят результаты оригинального многократных запусков алгоритма, реализующего локальный поиск из случайным образом выбранных вершин. Кроме того, в работе экспериментально показано, что для небольших задач алгоритм способен получить точное решение задачи.*

*Ключевые слова:* дискретная оптимизация,  $p$ -медианная задача, методы случайного поиска, задачи размещения.

Vestnik SibGAU  
2014, No. 5(57), P. 10–19**APPLICATION OF THE PROBABILITY CHANGING METHOD  
FOR OPTIMAL LOCATION PROBLEMS ON A NETWORK**

A. N. Antamoshkin, L. A. Kazakovtsev

Siberian State Aerospace University named after academician M. F. Reshetnev  
31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660014, Russian Federation  
E-mail: levk@bk.ru

*In this paper, the authors consider the  $p$ -median optimal location problem on a graph (network) which is a popular model for both logistic problems (optimal transportation, telecommunication and other infrastructure elements placement) and problems of cluster analysis, pattern recognition and estimation theory. The aim of the  $p$ -median problem on a graph (network) is to find the given number  $p$  of vertices of a graph such that the total distance from each vertex to the nearest chosen vertex reaches its minimum. To solve this NP-hard (in general) problem approximately, the authors propose a heuristic algorithm based on ideas of the probability changing method which is a random search method with adaptation. The idea of the proposed algorithm is based on random choosing of vertices in accordance with the given probabilities of choosing each of them. The probabilities for every chosen vertex and its neighborhood alter depending on the value of the objective function (i.e. in accordance with the total distance from each of the vertices to the nearest chosen vertex). For the algorithm for the large-scale  $p$ -median problems, a problem of optimal use of computational capacity is important. The effectiveness of the proposed algorithm was proved experimentally on test datasets from the ORLIB (special dataset library for operations research). In addition, the authors illustrate the results for the proposed algorithm combined with a local search procedure for the considered problem. The*

experiments show that the results of such combination exceed the results of the multiple start of the local search procedure from randomly chosen vertices. In addition, the experiments demonstrate the ability of the proposed algorithm to reach an exact solution for small problems.

*Keywords: discrete optimization,  $p$ -median problem, methods of random search, the location problem.*

**Введение.** Задачи размещения – класс задач оптимизации, в которых основными параметрами выступают координаты (местоположения) точек в некотором пространстве (непрерывном либо дискретном) и расстояния между ними [1; 2]. Задачи находят непосредственное практическое применение как при размещении производств, сетей, объектов инфраструктуры, датчиков и т. п., так и при решении задач кластерного анализа, распознавания образов, теории оценивания [3].

Задача Вебера (Ферма–Вебера) [1] – задача о поиске точки, сумма взвешенных расстояний от которой до известных точек (точек требования) минимальна, является обобщением простейшей задачи Ферма о нахождении точки, сумма расстояний от которой до трех известных точек минимальна, и имеет, в свою очередь, множество обобщений, главные из которых – множественная задача Вебера [4] и  $p$ -медианная задача [5] на плоскости или на сети:

$$F(X_1, X_2, \dots, X_M) = \sum_{i=1}^N w_i \min_{j \in \{1, M\}} L(X_j, A_i) \rightarrow \min, \quad (1)$$

где  $\{A_i | i = 1, N\}$  – множество точек требования;  $\{X_j | j = 1, p\}$  – множество размещаемых точек;  $w_i$  – весовой коэффициент  $i$ -й точки требования;  $L(\cdot)$  – функция расстояния.

В общем случае задача не может быть решена за полиномиальное время, алгоритм с полиномиальным временем известен только для деревьев [6]. Использование релаксации Лагранжа [7] позволяет получать приближенные решения задач размерности до  $n = 3\,795$ . Под размерностью задачи будем понимать число вершин графа.

Непрерывные задачи размещения в евклидовой метрике ( $l_2$ ), а также в метриках  $l_1$  (прямоугольной или манхэттенской) и  $l_\infty$  (чебышевской) изучены и созданы универсальные алгоритмы на основе процедуры Вайсфельда, исследована их вычислительная сложность [8; 9]. Решение тех же задач, например с запретными зонами [10], уже не является тривиальным. В практических задачах при использовании евклидовой или прямоугольной метрики такие модели являются достаточно грубым приближением [11; 12], поскольку не учитывают свойства пространства и средств транспортировки, в частности, наличие и качество дорог, барьеры, рельеф и т. п. Нередко функция расстояния является решением другой (вложенной) оптимизационной задачи и задается в этом случае алгоритмически [12]. Таким образом, и в случае задачи Вебера, зачастую целесообразной является ее дискретизация [13; 14]. Такие задачи можно рассматривать как  $p$ -медианные задачи на полном графе. Компромиссные подходы, сочетающие в формулировке задачи

элементы дискретного и непрерывного подходов, рассмотрены в [15–17].

В  $p$ -медианной задаче на сети [5], которая рассматривается в настоящей работе, поиск решений (медиан) осуществляется в известном конечном множестве узлов графа (сети). В данной задаче требуется выбрать  $p$  вершин таким образом, чтобы сумма взвешенных расстояний от каждой из вершин графа до ближайшей из выбранных вершин была минимальной. Такое множество будем называть  $p$ -медианой, а элементы этого множества – медианами.

Эвристические методы случайного поиска не гарантируют нахождение точного решения, но они статистически оптимальны: процент задач, решенных «почти оптимально», растет с ростом размерности задачи [18] (т. е. с ростом числа булевых переменных). При решении дискретных задач размещения применяется генетический алгоритм и другие методы [19; 20].

Изначально разработанный для решения безусловных задач псевдоболевой оптимизации, метод изменяющихся вероятностей (МИВЕР) [18; 21; 22] – это метод случайного поиска, который некоторыми авторами рассматривается как специфический вариант генетического алгоритма [23], в котором происходит полная смена поколения особей от итерации к итерации, а процедуре скрещивания соответствует процедура адаптации вероятностей и генерация нового поколения в соответствии с новыми вероятностями.

*Алгоритм 1.* Метод изменяющихся вероятностей.

Дано:  $f_b(X)$  – псевдоболева целевая функция – вещественная функция  $N$  булевых переменных,  $R$  – предельное число итераций. Здесь  $X = (x_1, \dots, x_N)$  – вектор булевых переменных.

1. Присвоить  $k = 0$ . Установить начальные значения вероятностей  $P_k = \{p_{k1}, p_{k2}, \dots, p_{kN}\}$ , где  $p_{kj} = P\{x_j = 1\}$ . Здесь и далее индекс  $k$  – номер итерации. Корректный выбор начальных значений  $P_0$  – важная проблема, особенно при переходе к условной оптимизации.

2. В соответствии с вектором распределений  $P_k$  генерируется набор  $N_v$  векторов  $X_{ki}$ ,  $i \in \{1, \dots, N_v\}$ , каждый элемент  $p_{kj}$  вектора  $P_k$  задает математическое ожидание генерации единицы в  $j$ -й компоненте векторов  $X_{ki}$ . Число генерируемых на каждом шаге векторов  $N_v$  называется размерностью популяции по аналогии с генетическим алгоритмом.

3. Вычисляется значение  $f_b(X_{ki})$  для каждого из сгенерированных векторов.

4. Выбираются некоторые значения целевой функции  $f_b(X_{ki})$  и соответствующие векторы  $X_{ki}$ , например, экземпляры сгенерированных векторов с максимальным и минимальным значением  $f_b(X_{ki})$ .

5. На основе результатов шага 4 модифицируется вектор  $P_k$ .

6.  $k = k + 1$ , если  $k < R$ , то переход к шагу 2. Возможны другие условия останова.

7. Иначе останов.

В задачах условной оптимизации применяются специальные модификации метода. Например, параллельные модификации алгоритма, предложенные в [24], позволяют решать задачи «рюкзачного» типа большой размерности (десятки тысяч переменных), а для задачи о коммивояжере, сформулированной в булевых переменных, число переменных может достигать  $10^6$ .

При создании алгоритма для  $p$ -медианных задач с большой размерностью и, соответственно, вычислительной сложностью, попутно должны быть решены задачи оптимального использования вычислительных мощностей. Прогнозируемое появление и распространение в ближайшем будущем персональных кластеров делают оправданной разработку параллельных алгоритмов размещения для систем с распределенной памятью. Эффективность параллельных алгоритмов на базе метода изменяющихся вероятностей, описанных в [13; 25; 26], для задач большой размерности доказана экспериментально.

Целью данной работы является адаптация метода изменяющихся вероятностей для решения  $p$ -медианной задачи и исследование применимости полученного алгоритма для решения задач большой размерности ( $n > 500$ ).

**Постановка задачи, алгоритм расчета значения целевой функции.** Пусть  $G = (V, E)$  – ненаправленный связный граф (сеть),  $V = \{v_1, \dots, v_n\}$  – множество вершин,  $E = \{e_i | i = \overline{1, m}\}$  – множество ребер,  $e_i = (v_j, v_k)$ ,  $j \in \{1, \dots, n\}$ ,  $k \in \{1, \dots, n\} \forall i = \overline{1, m}$  без петель, т. е.  $e_i \neq (v_j, v_j)$ ,  $\forall i = \overline{1, m}$ ,  $j = \overline{1, n}$ . Каждому ребру  $e_i$  поставлена в соответствие его длина  $l_i > 0$  (для  $e_i = (v_j, v_k)$  введем обозначение  $l_{j,k} = l_i$ ), каждой  $j$ -й вершине поставлен в соответствие вес  $w_j \geq 0$ . Для любой пары вершин  $(i, j)$  определена функция расстояния  $L(i, j)$  – длина кратчайшего пути между данными вершинами:

$$L(i, j) = \sum_{k \in P_{i,j}^*} l_k = \min_{P_{path} \in P_{i,j}} \sum_{k \in P_{path}} l_k,$$

где  $P_{i,j}^*$  – множество ребер, составляющих кратчайший путь между  $i$ -й и  $j$ -й вершинами;  $P_{i,j}$  – множество всех возможных путей между данными вершинами;  $P_{path}$  – любой путь из множества возможных путей  $P_{i,j}$ .

Тогда  $p$ -медианную задачу можно сформулировать как

$$\begin{aligned} & \min_{m_1, \dots, m_p \in \{1, n\}} f(m_1, \dots, m_p) = \\ & = \min_{m_1, \dots, m_p \in \{1, n\}} \sum_{i=1}^n w_i \min_{j \in \{1, p\}} L(m_j, i), p < n. \end{aligned} \quad (2)$$

Введем дополнительные обозначения.

$C_i = \{k \in \{1, n\} | \exists e_j = (v_i, v_k), j \in \{1, m\}\}$  – множество индексов окружения  $i$ -й вершины (ее окрестности первого порядка);

$l_i^* = \min_{j=1, p} L(m_j, i)$  – длина кратчайшего пути из  $i$ -й вершины до ближайшей из  $p$  вершин  $m_1, \dots, m_p$ .

Значения функции  $f(m_1, \dots, m_p)$  вычислим с помощью модификации алгоритма Дейкстры [27]:

*Алгоритм 2.* Расчет значения целевой функции.

Дано: индексы  $p$  выбранных вершин  $m_1, \dots, m_p$ .

1. Присвоить  $l_i^* = +\infty \forall i = \overline{1, n}$ .

2. Присвоить  $l_{m_i}^* = 0 \forall i = \overline{1, p}$ .

3. Присвоить  $V^* = \{m_1, \dots, m_p\}$ ;  $V^{**} = V^*$ .

4. Пока  $V^{**} \neq \emptyset$ , цикл:

4.1. Пока  $V' = \emptyset$ .

4.2. Для  $i \in V^{**}$  цикл:

4.2.1. Для  $j \in C_i$  цикл:

4.2.1.1. Если  $j \notin V^*$ , тогда  $V' = V' \cup \{j\}$ ;  $l_j^* = l_i^* + l_{i,j}$ .

4.2.1.2. Иначе, если  $l_j^* > l_i^* + l_{i,j}$ , тогда  $l_j^* = l_i^* + l_{i,j}$ .

4.2.1.3. Конец цикла 4.2.1.

4.2.2. Конец цикла 4.2.

4.3. Присвоить  $V^{**} = V'$ ;  $V^* = V^* \cup V'$ .

4.4. Конец цикла 4.

5. Возврат  $f(m_1, \dots, m_p) = \sum_{i=1}^n w_i l_i^*$ .

Метод изменяющихся вероятностей является методом псевдобулевой оптимизации, для его применения целевая функция должна быть выражена как функция булевых переменных.

Введем новые переменные  $x_1, \dots, x_n$ :

$$x_i = \begin{cases} 1, & i \in \{m_1, \dots, m_p\}; \\ 0, & i \notin \{m_1, \dots, m_p\}. \end{cases} \quad (3)$$

В случае булевых переменных задача имеет ограничение

$$\sum_{i=1}^n x_i = p. \quad (4)$$

Обратный переход к исходным целочисленным переменным осуществляется следующим образом:

$$\{m_1, \dots, m_p\} = \{i | x_i = 1, i = \overline{1, n}\}.$$

Задачу с псевдобулевой целевой функцией можно сформулировать следующим образом:

$$\begin{aligned} f_b(x_1, \dots, x_n) &= f(\{j | x_j = 1, j = \overline{1, n}\}) = \\ &= \sum_{i=1}^n w_i \min_{\{j | x_j = 1, j = \overline{1, n}\}} L(j, i) \end{aligned}$$

с ограничением (4).

В данном виде задача может быть решена различными методами псевдобулевой оптимизации, в том числе и методом изменяющихся вероятностей. Предлагаемый нами алгоритм на базе метода изменяющихся вероятностей учитывает условие (4) на этапе

генерации очередной популяции векторов (шаг 2 алгоритма 1).

**Алгоритм на базе метода изменяющихся вероятностей.** Пусть  $L_{\max}$  – диаметр графа (максимальное из расстояний между парами вершин):

$$L_{\max} = \max_{i,j \in \{1, \dots, n\}} L(i, j),$$

$l_{\text{avg}}$  – средняя длина ребра:

$$l_{\text{avg}} = \sum_{j=1}^m l_j / m.$$

Наш алгоритм основан на следующих предположениях:

1. При

$$l_{\text{avg}} / L_{\max} \rightarrow 0 \quad (5)$$

свойства задачи размещения на сети приближаются к свойствам непрерывной задачи, а именно: при замене в решении  $\{m_1, \dots, m_p\}$  любой  $m_i$ -й вершины на другую  $j$ -ю вершину такую, что  $L(j, m_i)$  мало, изменение значения целевой функции будет малым.

2. Если в решении  $\{m_1, \dots, m_p\}$  для любых двух вершин  $v_{m_i}$  и  $v_{m_j}$  расстояние  $L(m_i, m_j)$  мало, велика вероятность того, что при замене одной из компонент решения  $m_i$  на  $k$ -ю точку такую, что при  $L(m_i, k) \gg \gg L(m_i, m_j)$ , значение целевой функции для нового решения будет ближе к оптимальному:

$$f(m_1, \dots, m_i, \dots, m_j, \dots, m_p) > f(m_1, \dots, m_i, \dots, k, \dots, m_p).$$

Под «малым» расстоянием будем подразумевать  $L(m_i, m_j) < L_0, L_0 = \text{const}$ .

Эксперимент для доказательства данного утверждения приведен ниже.

Наша модификация алгоритма 1 сводится к изменению на шагах 2 и 5.

На шаге генерации экземпляров вектора  $X$  должно быть учтено ограничение (4). Это достигается следующим алгоритмом.

**Алгоритм 3.** Генерация одного члена популяции на шаге 2 алгоритма 1.

Дано: вектор вероятностей  $P = (p_1, \dots, p_n)$ , число вершин в решении  $p$ .

1. Присвоить  $\chi = \emptyset$ .

2. Пока  $|\chi| < p$ , цикл:

$$2.1. r = \text{Random}(\cdot) \cdot \sum_{j=1}^n p_j; S = 0.$$

2.2. Для каждого  $j \in \{1, \dots, n\}$  цикл:

2.2.1. Присвоить  $S = S + p_j$ .

2.2.2. Если  $S \geq r$ , тогда  $\chi = \chi \cup \{j\}$ ; прервать цикл

2.2.

2.2.3. Конец цикла 2.2.

2.3. Конец цикла 2.

3. Возврат  $\chi$ .

Результатом алгоритма является множество  $\chi$ .

Соответствующий вектор  $X_{ki}$ ,  $i \in \{1, \dots, N_v\}$  (см. шаг 2 алгоритма 1), может быть получен согласно формуле (3).

Здесь  $\text{Random}()$  – функция, генерирующая случайные значения в интервале  $[0,1)$  с равномерным распределением.

Данный алгоритм имеет следующую геометрическую интерпретацию. Расположим на числовой прямой, начиная от точки 0, смежные отрезки  $S_1, \dots, S_n$ , причем длина  $i$ -го отрезка равна  $p_i$ ,  $i = \overline{1, n}$ . Длина

суммарного отрезка равна  $\sum_{j=1}^n p_j$ . Выберем случайную

точку  $r$  на суммарном отрезке. Определим, какому из отрезков  $S_1, \dots, S_n$  она принадлежит. Вероятность попадания  $r$  в отрезок  $S_i$  пропорциональна  $p_i$ . Дополним множество  $\chi$  соответствующим индексом. Повторим, пока в  $\chi$  не наберется  $p$  индексов.

В следующей версии алгоритма снижается вероятность генерации решений с близкими друг к другу вершинами:

**Алгоритм 4.** Модифицированная процедура генерации члена популяции.

Дано: вектор вероятностей  $P = (p_1, \dots, p_n)$ , число вершин в решении  $p$ .

1. Присвоить  $\chi = \emptyset$ .

2. Пока  $|\chi| < p$ , цикл:

2.1.  $P^* = P$ .

$$2.2. r = \text{Random}(\cdot) \cdot \sum_{j=1}^n p_j^*; S = 0.$$

2.3. Для каждого  $j \in \{1, \dots, n\}$  цикл:

2.3.1. Присвоить  $S = S + p_j^*$ .

2.3.2. Если  $S \geq r$ , тогда  $\chi = \chi \cup \{j\}$ ;  $j' = j$ ; прервать цикл 2.3.

2.3.3. Конец цикла 2.3.

2.4. Для каждого  $j \in \{k \in \{1, \dots, n\} \mid L(j', k) < L_0\}$  присвоить  $p_j^* = p_j^* \cdot L(j, j') / L_0$ ; конец цикла 2.4.

2.5. Конец цикла 2.

3. Возврат  $\chi$ .

Алгоритм 4 дополнительно оперирует вторым временным вектором вероятностей  $P^*$ , компоненты которого меняются по ходу генерации элементов множества  $\chi$ . Если на очередной итерации множество дополнено индексом  $j'$ , то вероятности  $p_j$ , соответствующие вершинам, лежащим от вершины  $v_{j'}$  на расстоянии, не превышающем  $L_0$ , уменьшаются, дабы избежать генерации близких друг к другу вершин в решении.

Шаг 5 алгоритма 1 (адаптация вектора вероятностей) на  $k$ -й итерации выполняется по формулам

$$p_{k,i} = p_{(k-1),i} \cdot d_{k,i};$$

$$d_{k,i} = d_{k,i}^b / d_{k,i}^w;$$

$$d_{k,i}^b = \begin{cases} 1 + L_0 / (1 + L(i^b, i)), & L(i, i^b) < L_0; \\ 1, & L(i, i^b) \geq L_0; \end{cases}$$

$$d_{k,i}^w = \begin{cases} 1 + L_0 / (1 + L(i^w, i)), & L(i, i^w) < L_0; \\ 1, & L(i, i^w) \geq L_0. \end{cases}$$

Здесь  $i^b$  и  $i^w$  – ближайшие к  $i$ -й вершине такие, что  $i^b \in \chi^b$ ,  $i^w \in \chi^w$ , где  $\chi^b$  и  $\chi^w$  – те экземпляры сгенерированных алгоритмом 2 или алгоритмом 3 множеств индексов, входящих в решение вершин  $\chi$ , в которых целевая функция принимает лучшее (минимальное) и худшее (максимальное) значения.

Следует отметить, что дискретизированную задачу Вебера, описанную в [13], можно считать частным случаем рассматриваемой в настоящей работе задачи на сети, вследствие чего предлагаемые алгоритмы имеют некоторое сходство.

**Результаты экспериментов.** Сначала докажем экспериментом состоятельность предположения 2.

*Алгоритм 5.* Дано: граф  $G$  с  $n$  вершинами, количество вершин в решении  $p$ , предельное расстояние  $L_0$ .

1. Присвоить  $N = 0$ ,  $N_s = 0$ .

2. Выбрать случайным образом  $p$  индексов вершин из  $n$ :  $m_1, \dots, m_p \in \{1, n\}$ .

3. Если не существуют  $i, j \in \{1, p\}: L(m_i, m_j) < L_0$ , то перейти к шагу 2.

4. Выбрать случайным образом  $k \in \{1, n\}$ . Если  $L(k, m_j) < L_0$ , то повторить шаг 4.

5. Если  $f(m_1, \dots, m_i, \dots, m_j, \dots, m_p) \geq f(m_1, \dots, m_i, \dots, k, \dots, m_p)$ , то  $N_s = N_s + 1$ .

6. Присвоить  $N = N + 1$ ; если  $N < 1\ 000$ , то перейти к шагу 2.

7. Останов, возврат  $P_s = N_s / 1000$ .

Результаты работы алгоритма 5 для автоматически сгенерированных задач (см. алгоритм 6) и для задач rmed22 и rmed39 из набора тестовых задач ORLIB [28] представлены на рис. 1. Оценка вероятности дана в процентах. Для задач rmed22 и rmed39, благодаря специфике их генерации [28], соотношение  $l_{avg} / L_{max}$  велико, и условие предположения 2 выполняется лишь при очень малых значениях  $L_0$  (много меньших, чем  $l_{avg}$ ).

Для целей тестирования по основной теме статьи мы использовали автоматически сгенерированные тестовые задачи (см. алгоритм 6), схема одной из которых показана на рис. 2, а. Здесь длины ребер изображены в масштабе, диаметр вершин на изображении соответствует их весовому коэффициенту. На рис. 2 также изображено полученное нашим алгоритмом решение для  $p = 32$ . Здесь вершины, входящие в решение, обведены. Расстояние от каждой вершины до ближайшей вершины, входящей в решение, отмечено соответствующим цветом: вершины с меньшим расстоянием изображены темным цветом, с большим – светлым.

*Алгоритм 6.* Генератор тестовых задач.

Дано:  $n$  – число вершин.

1. Для каждого  $i \in \{1, n\}$  цикл:  $a_i = Random() \cdot 500$ ;  $b_i = Random() \cdot 500$ ,  $w_i = 0,5 + Random() \cdot 10$ ; конец цикла.

2. Заполнить матрицу смежности  $A$  нулевыми значениями; множество ребер  $E = \emptyset$ .

3. Для  $i \in \{1, n\}$  цикл:

3.1. Если  $i > 0,7n$ , тогда  $D_i = 1$ .

3.2. Иначе, если  $i > 0,3n$ , тогда  $D_i = 2$ .

3.3. Иначе  $D_i = 3 + [Random() \cdot 4]$ .

3.4. Для  $d \in \{1, \sum_{j=1}^n A_{i,j}\}$  цикл:

$j = \arg \min_{j \in \{1, n\}, A_{i,j} > 0} (|a_i - a_j| + |b_i - b_j|)$ ;  $A_{i,j} = 1$ ;  $A_{j,i} = 1$ ;

добавить к множеству  $E$  ребро  $(i, j)$ ;

$l_{i,j} = |a_i - a_j| + |b_i - b_j|$ ; конец цикла.

4. Останов, возврат результатов: матрица смежности  $A$ , множество ребер  $E$ , длины ребер  $l_{i,j}$ , где  $(i, j) \in E$ , весовые коэффициенты  $w_i$ , где  $i \in \{1, n\}$ .

В алгоритме 6  $Random()$  – функция, генерирующая случайные значения в интервале  $[0, 1)$  с равномерным распределением. На шаге 3.4 ведется поиск вершин, близких к  $i$ -й, которые соединяются с ней ребрами, вследствие чего  $l_{avg} / L_{max}$  мало (см. условие (5)).

Ход решения (изменение значений вероятностей) проиллюстрирован на рис. 3, где вершины, для которых вероятность попадания в экземпляр решения максимальна, изображены светлым цветом, вершины с минимальной вероятностью – темным.

Диаметр окрестности  $L_0$  – важный параметр, во многом определяющий эффективность работы алгоритма. При  $p = 12$  сравнение результатов при различных значениях этого параметра показано на рис. 4.

Для построения графика зависимости достигнутого значения целевой функции от количества выполненных итераций  $k$  (см. шаг 6 алгоритма 1) в алгоритм добавлены специальные команды, записывающие значение целевой функции в массив. Для задачи, схема которой дана на рис. 2 ( $n = 500$ ,  $p = 12$ ), выполнялось по 10 запусков алгоритма с каждым из значений  $L_0 \in \{1, 3, 10, 25, 50, 90, 120, 170, 250, 350, 500, 750\}$ . На графике показаны усредненные значения для 10 запусков, достигнутые после каждой итерации. Линии  $L_0 = 0$  соответствует алгоритм, не учитывающий соседство вершин, в котором генерация поколений экземпляров векторов  $X$  осуществляется согласно алгоритму 3. Наилучшие результаты достигаются при  $L_0 = 90$ . Оптимальное значение параметра  $L_0$  зависит от параметра  $p$ . Тем не менее при изменении параметра  $L_0$  в широком диапазоне ( $L_0 = 10 \dots 350$ ) для данной задачи алгоритм стабильно показывает лучшие результаты, чем «классический» алгоритм схемы МИВЕР с генерацией экземпляров вектора  $X$  согласно алгоритму 3.

Применимость алгоритма для задач большой размерности проверена на тестовом примере с  $n = 5\ 000$  (рис. 5). Использовалась вычислительная система DEPO X8Sti (6-core CPU Xeon X5650 2.67 GHz, 12Gb RAM), HyperThreading отключен, компилятор ifort с опцией полной оптимизации (-O3).

Сравнение с алгоритмами, дающими точное решение, затруднено вследствие высокой вычислительной сложности задач. Так, например, метод ветвей и границ при соответствии задачи условию (5) практически вырождается в процедуру полного перебора. Методы,

предложенные Хаками [5], также имеют экспоненциальную асимптотическую сложность. Для  $p = 3$  сравнение результата на сети с 170 вершинами, полученного нашим алгоритмом и алгоритмом случайного поиска без адаптации (хаотический поиск, при котором на каждой итерации случайным образом выбираются  $p$  вершин), с точным результатом, полученным полным перебором (горизонтальная асимптота), показано на рис. 6. Для построения графика к шагу 6 алгоритма 1 добавлены команды, которые через каждые 10 итераций фиксируют в специальном массиве достигнутое значение целевой функции и время с момента запуска алгоритма. Запуск алгоритма 1 и алгоритма хаотического поиска осуществлен по 10 раз, на рис. 6 показан усредненный (для 10 запусков) результат хаотического поиска и худший (из 10 запусков) результат нашего алгоритма на базе метода изменяющихся вероятностей. Следует отметить, что модифицированный алгоритм как улучшает усредненные результаты 10 запусков, так и несколько сужает разброс результатов. Так, анализ наихудших значений, полу-

ченных алгоритмами после 10 запусков, дает картину, аналогичную на рис. 6. Вычислительная сложность полного перебора эквивалентна 402 220 итерациям алгоритма 1 (требуется 4 826 640 вычислений целевой функции, алгоритм генерирует на каждом шаге 12 экземпляров вектора  $X$ ).

Скомбинируем предложенный алгоритм с процедурой локального поиска. Дополним алгоритм 2 шагом 0:

Шаг 0 алгоритма 2: запустить процедуру локального поиска из начального множества вершин с индексами  $m_1, \dots, m_p$ .

Мы будем использовать простейший [29] из множества методов локального поиска для данной задачи [30] – поиск в окрестности, состоящей из вершин, смежных с выбранными вершинами  $m_1, \dots, m_p$ . Сравнение результатов полученного алгоритма с результатами многократного запуска процедуры локального поиска из случайно (хаотически) выбранных вершин для сгенерированной задачи большой размерности показано на рис. 7.

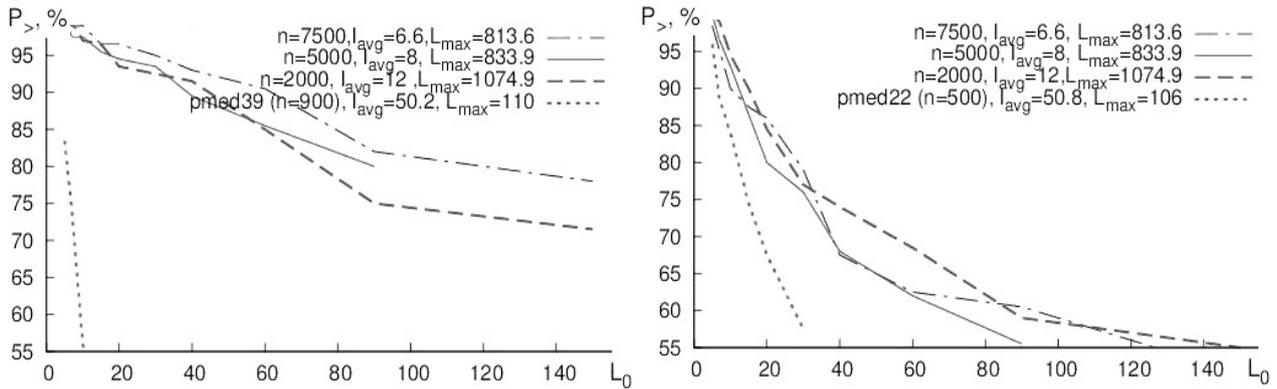


Рис. 1. Соотношение оценки вероятности выполнения условия предположения 2 и предельного расстояния  $L_0$  для сгенерированных задач и задач из библиотеки ORLIB (pmed39 и pmed22). Указаны параметры задачи:  $n$  – число вершин сети,  $l_{avg}$  – среднее,  $L_{max}$  – максимальное расстояние между вершинами в сети

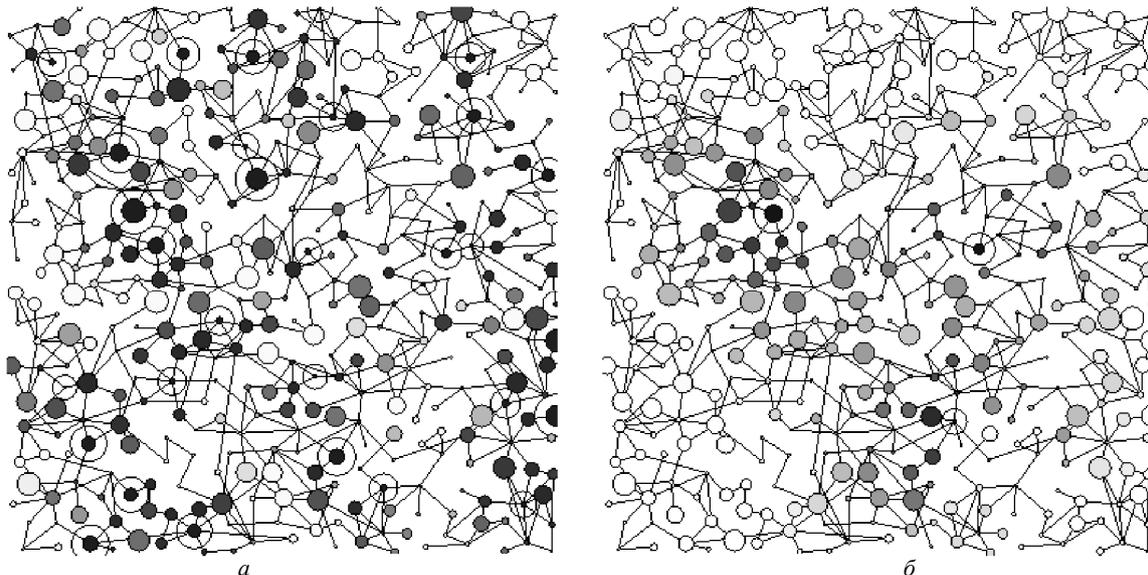


Рис. 2. Схема сгенерированной задачи ( $n = 500$ ) и ее решения (обведены окружностями):  
 а – решение для  $p = 32$ ; б – решение для  $p = 3$

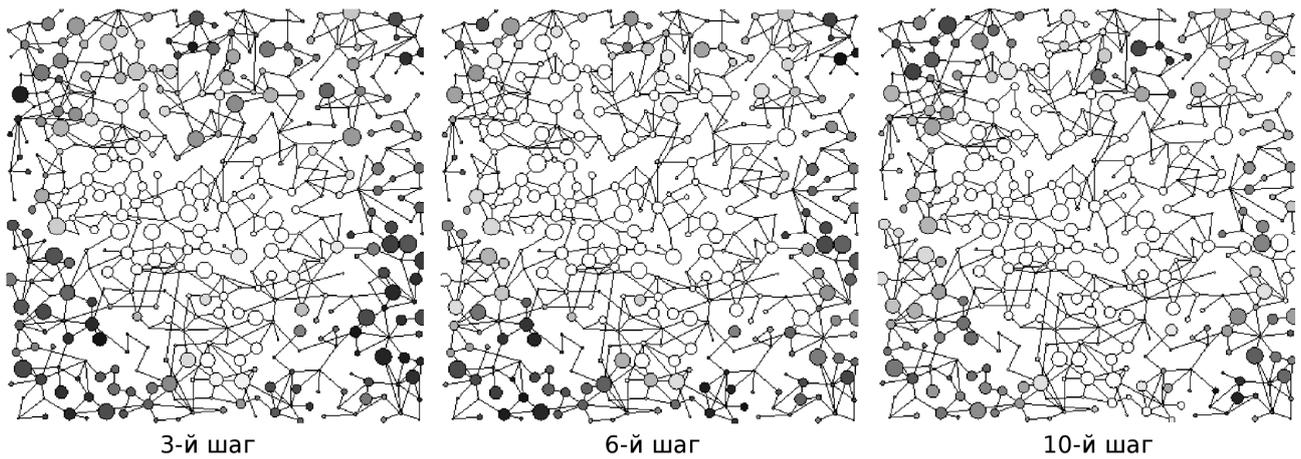


Рис. 3. Ход решения (изменение значений вероятностей) для  $p = 3, L_0 = 100$

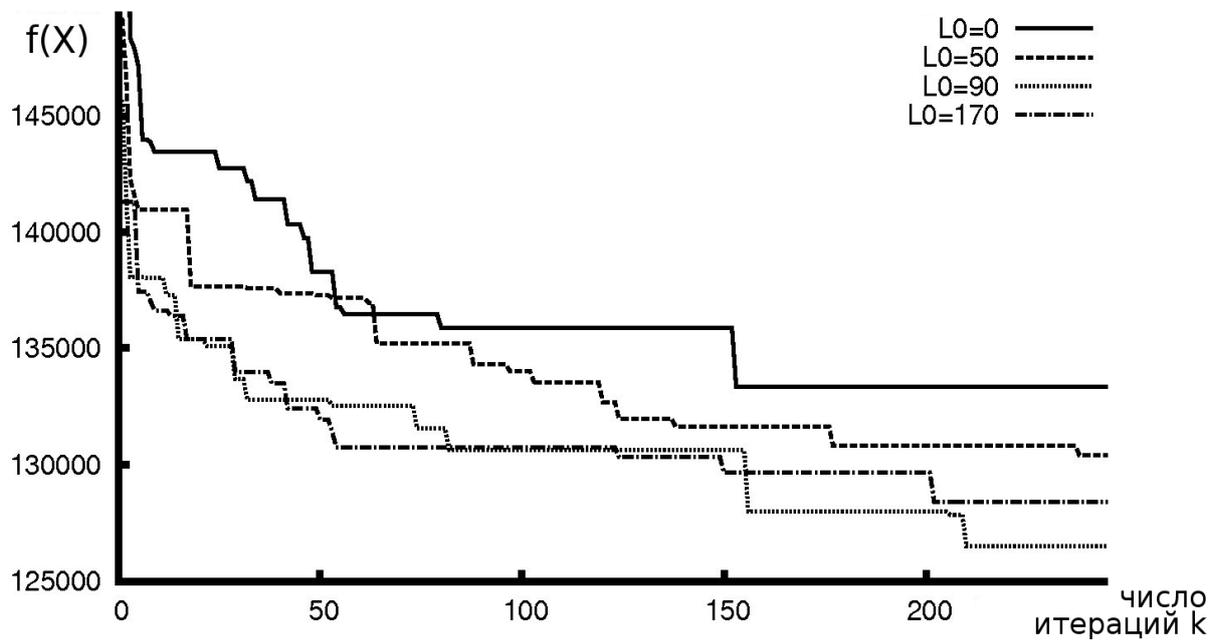


Рис. 4. Сравнение результатов с различными значениями параметра  $L_0, p = 12$

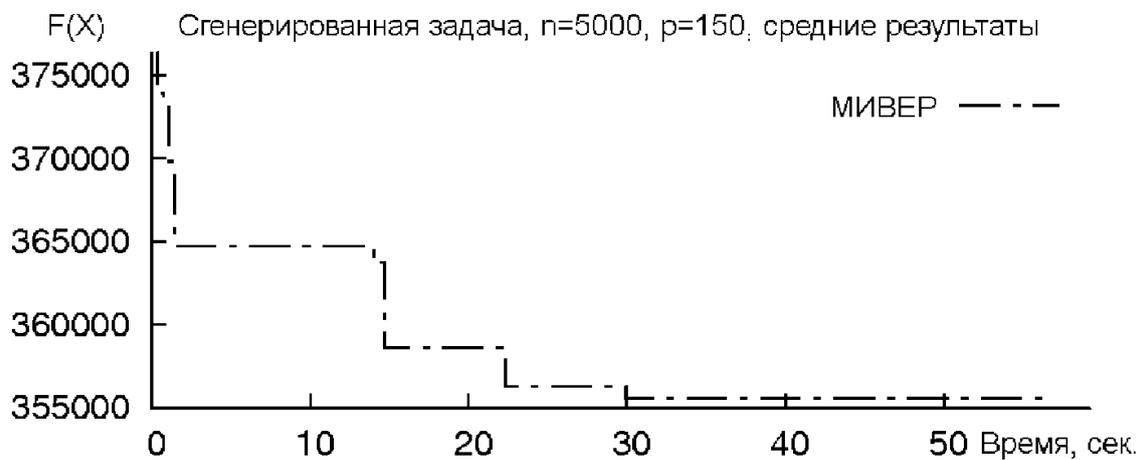


Рис. 5. Усредненные результаты для 10 попыток запуска тестового примера с  $n = 5\ 000$

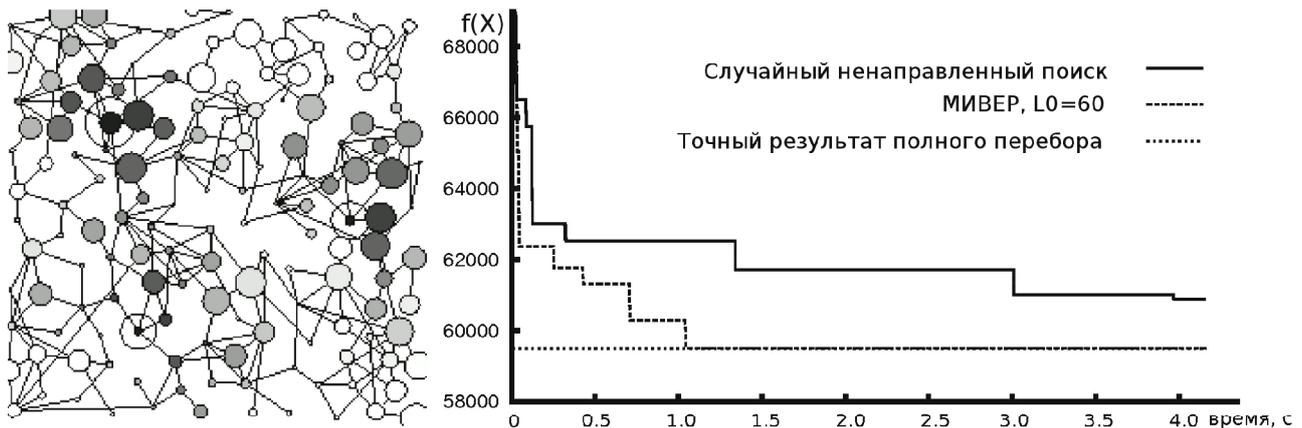


Рис. 6. Схема задачи с оптимальным решением и сравнение результата алгоритма для  $p = 3$  с результатом хаотического поиска и полного перебора

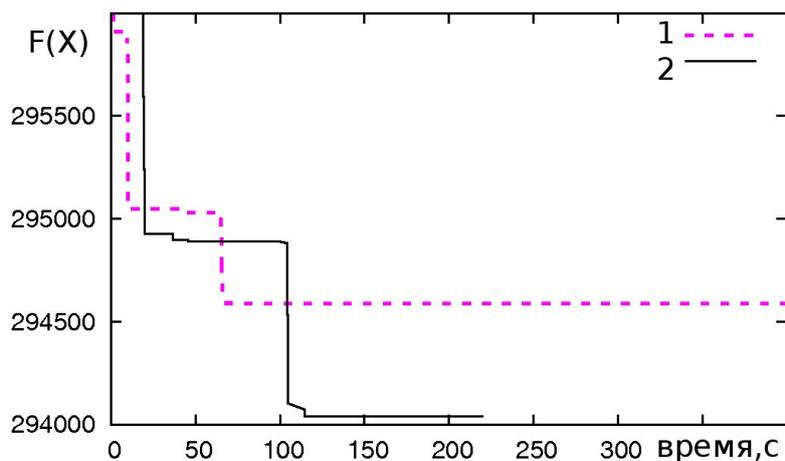


Рис. 7. Сравнение результатов комбинированного с локальным поиском алгоритма с результатами локального поиска из хаотически выбранных вершин. Сгенерированная задача,  $n = 7500$ ,  $p = 10$ : 1 – комбинированный алгоритм, 2 – локальный поиск

**Заключение.** Предложенный эвристический алгоритм на базе метода изменяющихся вероятностей может быть использован для решения  $p$ -медианной задачи на сети при условии, что длины ребер сети много меньше, чем ожидаемая длина пути между произвольно выбранными вершинами. Поиск оптимальных параметров алгоритма остается темой дальнейших исследований.

#### Библиографические ссылки

1. Wesolowsky G. The Weber problem: History and perspectives // Location Science. 1993. Vol. 1. P. 53.
2. Drezner Z., Wesolowsky G. O. A Trajectory Method for the Optimization of the Multifacility Location Problem with  $l_p$  Distances // Management Science. 1978. Vol. 24. P. 1507–1514.
3. Osinuga I. A., Bamigbola O. N. On the Minimum Norm Solution to Weber problem // SAMSA Conference Proceedings. Windhoek, 2007. P. 27–30.
4. Cooper L. An extension of the generalized Weber problem // Journal of Regional Science. 1968. Vol. 8, Iss. 2. P. 181–197.
5. Hakimi S. L. Optimum locations of switching centers and the absolute centers and medians of a graph // Operations Research. 1964. Iss. 12(3). P. 450–459.
6. Kariv O., Hakimi S. L. An algorithmic approach to network location problems. II. The  $p$ -medians // SIAM Journal on Applied Mathematics. 1979. Vol. 37(3). P. 539–560.
7. Avella P., Sassano A., Vasil'ev I. Computational Study of Large-Scale  $p$ -Median Problems // Mathematical Programming. 2007. Vol. 109(1). P. 89–114.
8. Masuyama, S., Ibaraki T., Hasegawa T. The Computational Complexity of the  $m$ -Center Problems on the Plane // The Transactions of the Institute of Electronics and Communication Engineers of Japan. 1981. Vol. 64E. P. 57–64.
9. Megiddo N., Supowit K. On the Complexity of Some Common Geometric Location Problems // SIAM Journal on Computing. 1984. No. 13. P. 182–196.
10. Забудский Г. Г., Нежинский И. В. Решение задачи размещения в евклидовом пространстве с запрещенной областью // Вестник Омского университета. 1999. № 2. С. 17–19.

11. Morris J. G. Convergence of the Weiszfeld Algorithm for Weber Problems Using a Generalized "Distance" Function // *Operations Research*. 1981. Vol. 29, No. 1. P. 37–48.

12. Wesolowsky G. O, Love R. F. A Nonlinear Approximation Method for Solving a Generalized Rectangular Distance Weber Problem // *Management Science*. 1972. Vol. 18, No. 11. P. 656–663.

13. Kazakovtsev L. A. Adaptation of the Probability Changing Method for Weber Problem with an Arbitrary Metric // *Facta Universitatis series Mathematics and Informatics*, University of Nis. 2012. Vol. 27, No. 2. P. 239–254.

14. Антамошкин А. Н., Казаковцев Л. А. Алгоритм случайного поиска для обобщенной задачи Вебера в дискретных координатах // *Информатика и системы управления*. 2013. Вып. 1. С. 87–98.

15. Single-facility Weber Location Problem Based on the Lift Metric / P. S. Stanimirovic [et al.] // *Facta Universitatis series Mathematics and Informatics*. 2012. Vol. 27, No. 2. P. 175–190.

16. Gugat M., Pfeiffer B. Weber problems with mixed distances and regional demand // *Math. Meth. Oper. Res.* 2007. Iss. 66. P. 419–449.

17. Bischoff M., Fleischmann T., Klamroth K. The Multi-Facility Location-Allocation Problem with Polyhedral Barriers // *Computers and Operations Research*. 2009. No. 36. P. 1376–1392.

18. Антамошкин А. Н., Сараев В. Н. Метод изменяющихся вероятностей // *Проблемы случайного поиска*. Рига : Зинатне. 1988. Вып. 11. С. 26–34.

19. A Comprehensive Study of Optimization Algorithm for Wireless Coverage in Indoor Area / A. W. Reza [et al.] // *Optimization Letters*. 2012. Doi 10.1007/s11590-012-0543-z. URL: <http://link.springer.com/article/10.1007%2Fs11590-012-0543-z?LI=true> (дата обращения: 10.05.2013).

20. Alp O., Erkut E., Drezner Z. An Efficient Genetic Algorithm for the p-Median Problem // *Annals of Operations Research*. 2003. Vol. 122, Iss. 1–4. P. 21–42.

21. Antamoshkin A., Saraev V. On Definition of Informative Subsystem of Signs in the Pattern Recognition Problem // *Computers and Artificial Intelligence*. 1985. Vol. 4, Iss. 3. P. 245–252.

22. Antamoshkin A. Brainware for Searchal Pseudoboolean Optimization // *Tr. of Tenth Prague Conf. on Inf. Theory, Stat.Dec. Functions, Random Processes*. Prague : Academia. 1987. Vol. 10A-B. P. 203–206.

23. Дегтерев А. С., Канашкин Ф. В., Сумароков А. Д. Обобщение генетических алгоритмов и алгоритмов схемы МИВЕР [Электронный ресурс] // *Электронный журнал «Исследовано в России»*. 2004. Т. 7. С. 1658–1665. URL: <http://zhurnal.ape.relarn.ru/articles/2004/153.pdf> (дата обращения: 13.05.2013).

24. Казаковцев Л. А., Ступина А. А. Параллельная реализация метода изменяющихся вероятностей // *Современные проблемы науки и образования*. 2012. № 4. URL: [www.science-education.ru/104-6810](http://www.science-education.ru/104-6810) (дата обращения: 31.10.2012).

25. Kazakovtsev L. A. Random Constrained Pseudo-Boolean Optimization Algorithm for Multiprocessor Systems and Clusters // *ICUMT 2012, International*

*Congress on Ultra-Modern Telecommunications*. IEEE Press. S-Petersburg, 2012. P. 650–656.

26. Казаковцев Л. А. Параллельный алгоритм случайного поиска с адаптацией для систем с распределенной памятью // *Системы управления и информационные технологии*. 2012. № 3 (49). С. 11–15.

27. Dijkstra E. W. A note on two problems in connexion with graphs // *Numerische Mathematik*. 1959. Vol. 1. P. 269–271.

28. Beasley J. E. OR-Library: distributing test problems by electronic mail // *Journal of the Operational Research Society*. 1990. Vol. 41, No. 11. P. 1069–1072.

29. Antamoshkin A. N., Kazakovtsev L. A. Random Search Algorithm for the p-Median Problem // *Informatica (Ljubljana)*. 2013. Vol. 37(1). Pp. 267–278.

30. Scatter Search and Path Relinking: Fundamentals, Advances and Applications / M. G. C. Resende [et al.] // *Handbook of Metaheuristics*. 2nd ed. / M. Gendreau and Y. Potvin (editors). Springer, 2010. Pp. 87–107.

## References

1. Wesolowsky G. The Weber problem: History and perspectives. *Location Science*. 1993, vol. 1, p. 53.

2. Drezner Z., Wesolowsky G. O. A Trajectory Method for the Optimization of the Multifacility Location Problem with lp Distances. *Management Science*. 1978, vol. 24, p. 1507–1514.

3. Osinuga I. A., Bamigbola O. N. On the Minimum Norm Solution to Weber problem. *SAMSA Conference Proceedings*. Windhoek, 2007, p. 27–30.

4. Cooper L. An extension of the generalized Weber problem. *Journal of Regional Science*. 1968, vol. 8, Iss. 2, p. 181–197.

5. Hakimi S. L. Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research*. 1964, Iss. 12(3), p. 450–459.

6. Kariv O., Hakimi S. L. An algorithmic approach to the network location problems. II. The p-medians. *SIAM Journal on Applied Mathematics*. 1979, vol. 37(3), p. 539–560.

7. Avella P., Sassano A., Vasil'ev I. Computational Study of Large-Scale p-Median Problems. *Mathematical Programming*. 2007, vol. 109(1), p. 89–114.

8. Masuyama S., Ibaraki T., Hasegawa T. The Computational Complexity of the m-Center Problems on the Plane. *The Transactions of the Institute of Electronics and Communication Engineers of Japan*. 1981, vol. 64E, p. 57–64.

9. Megiddo N., Supowit K. On the Complexity of Some Common Geometric Location Problems. *SIAM Journal on Computing*. 1984, no. 13, p. 182–196.

10. Zabudsky G. G., Nezhinsky I. V. [Solution to the problem of accommodation in Euclidean space with a prohibited area]. *Byulleten Omskogo universiteta*. 1999, no. 2, p. 17–19 (In Russ.).

11. Morris J. G. Convergence of the Weiszfeld Algorithm for Weber Problems Using a Generalized "Distance" Function. *Operations Research*. 1981, vol. 29, no. 1, p. 37–48.

12. Wesolowsky G. O, Love R. F. A Nonlinear Approximation Method for Solving a Generalized

- Rectangular Distance Weber Problem. *Management Science*. 1972, vol. 18, no. 11, p. 656–663.
13. Kazakovtsev L. A., Adaptation of the Probability Changing Method for Weber Problem with an Arbitrary Metric. *Facta Universitatis series Mathematics and Informatics, University of Nis*. 2012, vol. 27, no. 2, p. 239–254.
14. Antavoshkin A. N., Kazakovtsev L. A. [Random search algorithm for the generalized Weber problem in discrete coordinates]. *Informatika i sistemy upravleniya*. 2013, vol. 1, p. 87–98 (In Russ.).
15. Stanimirovic P. S., Ciric M., Kazakovtsev L. A., Osinuga I. A. Single-facility Weber Location Problem Based on the Lift Metric. *Facta Universitatis series Mathematics and Informatics*. 2012, vol. 27, no. 2, p. 175–190.
16. Gugat M., Pfeiffer B. Weber problems with mixed distances and regional demand. *Math. Meth. Oper. Res.* 2007, Iss. 66, p. 419–449.
17. Bischoff M., Fleischmann T., Klamroth K. The Multi-Facility Location-Allocation Problem with Polyhedral Barriers. *Computers and Operations Research*. 2009, no. 36, p. 1376–1392.
18. Antamoshkin A. N., Saraev V. N. [Probability changing method]. *Problemy sluchainogo poiska*. 1988, Iss. 11, p. 26–34 (In Russ.).
19. Reza A. W., Dimiyati K., Noordin K. A., Kausar A. S. M. Z., Sarker Md. S. A Comprehensive Study of Optimization Algorithm for Wireless Coverage in Indoor Area. *Optimization Letters*. 2012. doi 10.1007/s11590-012-0543-z, Available at: <http://link.springer.com/article/10.1007%2Fs11590-012-0543-z?LI=true> (accessed 10.05.2013).
20. Alp O., Erkut E. and Drezner Z. An Efficient Genetic Algorithm for the p-Median Problem. *Annals of Operations Research*. 2003, vol. 122, Iss. 1–4, p. 21–42.
21. Antamoshkin A., Saraev V. On Definition of Informative Subsystem of Signs in the Pattern Recognition Problem. *Computers and Artificial Intelligence*. 1985, vol. 4, Iss. 3, p. 245–252.
22. Antamoshkin A. Brainware for Searchal Pseudoboolean Optimization. *Tr. of Tenth Prague Conf. on Inf. Theory, Stat.Dec. Functions, Random Processes*. Prague: Academia. 1987, vol. 10A-B, p. 203–206.
23. Degterev A. S., Kanashkin F. V., Sumarokov A. D. [Generalization of genetic algorithms and algorithms of MIVER scheme]. *Issledovano v Rossii*. 2004, vol 7, p. 1658–1665. (In Russ.). Available at: <http://zhurnal.apelarn.ru/articles/2004/153.pdf> (accessed: 13.05.2013)
24. Kazakovtsev L. A., Stupina A. A. [Parallel implementation of the probability changing method]. *Sovremennye problem nauki i obrazovaniya*. 2012, no. 4 (In Russ.). Available at: [www.science-education.ru/104-6810](http://www.science-education.ru/104-6810) (accessed: 31.10.2012).
25. Kazakovtsev L. A. Random Constrained Pseudo-Boolean Optimization Algorithm for Multiprocessor Systems and Clusters. ICUMT 2012, International Congress on Ultra-Modern Telecommunications. IEEE Press. S-Petersburg, 2012. P. 650–656.
26. Kazakovtsev L. A. [Parallel algorithm of random search with adaptation for systems with distributed memory]. *Sistemy upravleniya i informatsionnye tekhnologii*. 2012, no. 3(49), p. 11–15 (In Russ.).
27. Dijkstra E. W. A note on two problems in connexion with graphs. *Numerische Mathematik*. 1959, vol. 1, p. 269–271.
28. Beasley J. E. OR-Library: distributing test problems by electronic mail. *Journal of the Operational Research Society*. 1990, vol. 41, no. 11, p. 1069–1072.
29. Antamoshkin A. N., Kazakovtsev L. A. Random Search Algorithm for the p-Median Problem. *Informatika (Ljubljana)*. 2013, vol. 37(1), p. 267–278.
30. Resende M. G. C., Ribero C. C., Glover F., Marti R. Scatter Search and Path Relinking: Fundamentals, Advances and Applications. *Handbook of Metaheuristics*, 2nd edition, Gendreau M. and Potvin Y. (editors), Springer, 2010, p. 87–107.