ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль—август 2011. Том 18, № 4. С. 3–16

УДК 519.8

ЛОКАЛЬНЫЙ ПОИСК ПО ОБОБЩЁННОЙ ОКРЕСТНОСТИ ДЛЯ ЗАДАЧИ ОПТИМИЗАЦИИ ПСЕВДОБУЛЕВЫХ ФУНКЦИЙ *)

В. Л. Береснев, Е. Н. Гончаров, А. А. Мельников

Аннотация. Для задачи оптимизации псевдобулевой функции рассматривается алгоритм локального поиска с обобщённой окрестностью. Такая окрестность строится для локально-оптимальных решений и включает в себя другие локально-оптимальные решения, «окружающие» данное. Приводятся результаты вычислительных экспериментов с использованием псевдобулевых функций, оптимизация которых эквивалентна задачам размещения предприятий, покрытия множества и конкурентного размещения предприятий. Целью экспериментов является сравнительная оценка локально-оптимальных решений, получаемых стандартным алгоритмом локального поиска и алгоритмом локального поиска с обобщённой окрестностью.

Ключевые слова: оптимизация, локальный спуск, полином от булевых переменных, задача размещения предприятий, задача о покрытии.

Введение

Рассматривается задача нахождения экстремума функции, заданной на множестве (0,1)-векторов и принимающей действительные (целые) значения. Такие функции называются *псевдобулевыми* [10]. Отыскание оптимального решения указанной задачи для многих видов псевдобулевых функций связано с принципиальными трудностями, поэтому в качестве основы при построении нетрудоёмких алгоритмов решения таких задач служат метод локального поиска и различные его модификации [11].

*)Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проекты 09–01–00059, 10–07–00195), ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № 14.740.11.0219), программы фундаментальных исследований СО РАН (междисциплинарный проект № 30, проект № 44), а также программы № 14 Президиума РАН (проект № 227).

Результатом работы этих алгоритмов является локально-оптимальное решение, т. е. наилучшее решение в пределах заданного множества, называемого окрестностью этого решения.

Использование обобщённой окрестности позволяет усилить требование к отыскиваемому решению. Оно должно быть не только локально-оптимальным, но еще и наилучшим в сравнении с другими локально-оптимальными решениями, «окружающими» данное решение.

В разд. 1 вводится понятие обобщённой окрестности для локально-оптимального решения и исследуется её строение на примере задачи минимизации псевдобулевой функции, эквивалентной задаче размещения предприятий [1, 8] и задаче о покрытии множества [1, 5]. Приводятся результаты вычислительного эксперимента с использованием известных примеров задачи размещения предприятий [4] и задачи о покрытии множества [5, 6], показывающие, какое количество элементов содержится в обобщённой окрестности и на какие расстояния они удалены от исходного локально-оптимального решения.

В разд. 2 содержится описание алгоритма локального поиска по обобщённой окрестности и приводятся результаты вычислительного эксперимента с использованием тех же классов задач, что и в разд. 1. Целью эксперимента является сравнительная оценка локально-оптимального решения, получаемого на предварительном шаге алгоритма, и итогового решения, получаемого в результате локального поиска по обобщённой окрестности.

Разд. 3 посвящён исследованию алгоритма локального поиска с обобщённой окрестностью применительно к задаче конкурентного размещения предприятий [2, 3, 9], которая формулируется как задача двухуровневого целочисленного программирования [7] и может быть представлена как задача максимизации некоторой псевдобулевой функции. Однако эта функция, в отличие от рассмотренной в разд. 2, не задана в явном виде, и для вычисления её значения необходимо произвести достаточно трудоёмкие вычисления. Для алгоритма локального поиска по обобщённой окрестности, построенного на основе стандартного алгоритма локального поиска из [3], приводятся результаты вычислительного эксперимента, цели которого аналогичны целям предыдущих вычислительных экспериментов. Для примеров задачи конкурентного размещения предприятий из [4] приводятся результаты вычислений, характеризующие строение обобщённой окрестности и позволяющие дать сравнительную оценку решениям, получаемым стандартным алгоритмом локального поиска и алгоритмом с использованием обобщённой окрестности.

1. Обобщённая окрестность

Рассмотрим задачу минимизации псевдобулевой функции $f(x)$, определённой на множестве B^m (0, 1)-векторов $x = (x_i)$, $i \in I = \{1, \ldots, m\}$. Пусть для всякого $x \in B^m$ задано множество $N(x) \subset B^m$, называемое *окрестностью решения* x. В случае множества B^m в качестве окрестности $N(x)$ точки $x \in B^m$ обычно используют следующие:

$$N_1(x) \ = \ \{y \in B^m \mid d(x, y) \ = \ 1\},$$

$$N_2(x) \ = \ \{y \in B^m \mid d(x, y) \ = \ 2, \ d(0, x) \ = d(0, y)\},$$

где $d(x, y)$ — расстояние Хэмминга, равное числу несовпадающих компонент (0, 1)-векторов x и y.

При заданной окрестности $N(x)$ решение $x_0 \in B^m$ называется *локально-оптимальным*, если $f(x_0) \leqslant f(x)$ для всякого $x \in N(x_0)$.

Стандартный алгоритм локального поиска по заданной окрестности $N(x)$, $x \in B^m$, включает конечное число однотипных шагов, на каждом из которых рассматривается некоторое текущее решение x_0. На первом шаге в качестве x_0 может быть взят любой (0, 1)-вектор. Шаг состоит в поиске элемента $x' \in N(x_0)$, улучшающего текущее решение x_0, т. е. такого $x' \in N(x_0)$, что $f(x') < f(x_0)$. Если решения x' найти не удаётся, то алгоритм останавливается и текущее решение x_0 есть результат его работы. В противном случае текущее решение x_0 заменяется решением x' и начинается следующий шаг.

Способ выбора решения x' в алгоритме локального поиска нуждается в уточнении. Если задан некоторый порядок просмотра элементов множества $N(x_0)$, то решение x' может быть выбрано в результате частичного просмотра окрестности $N(x_0)$. Таким решением будет первый в заданном порядке элемент $x \in N(x_0)$, для которого $f(x) < f(x_0)$. При полном просмотре окрестности $N(x_0)$ в качестве улучшающего решения выбирается такой элемент x', что $f(x') < f(x_0)$ и $f(x') \leqslant f(x)$ для каждого $x \in N(x_0)$.

Пусть для всякого $x \in B^m$ задана окрестность $N(x) \in B^m$, которую будем называть *базовой*, и пусть x_0 — локально-оптимальное решение относительно этой окрестности. Определим обобщённую окрестность $\widetilde{N}(x_0)$ локально-оптимального решения x_0. Это множество содержит не более чем m других локально-оптимальных решений, каждое из которых совпадает с одним из векторов $\widetilde{x}_0^k$, $k \in I$. Для всякого $k \in I$ локально-оптимальное решение $\widetilde{x}_0^k$ определяется следующим образом.

При фиксированном $k \in I$ рассмотрим наряду с базовой окрестностью $N(x)$ окрестность $N^k(x) \ = \ \{y \in N(x) \mid y_k = x_k\}$ и (0,1)-вектор

$y^k = (y_i^k)$, $i \in I$, отличающийся от решения $x_0 = (x_{0i})$, $i \in I$, только тем, что $y_k^k = 1 - x_{0k}$. Локально-оптимальное решение $\widetilde{x}_0^k$ строится в два этапа. Сначала с использованием стандартной процедуры локального поиска по окрестности $N^k(x)$ и вектору y^k в качестве начальной точки определяется решение y_0^k. Это решение является локально-оптимальным относительно вспомогательной окрестности $N^k(x)$. Затем по решению y_0^k с помощью стандартной процедуры локального поиска по окрестности $N(x)$ строится локально-оптимальное решение $\widetilde{x}_0^k$.

Обобщённой окрестностью локально-оптимального решения x_0 назовём множество

$$\widetilde{N}(x_0) = \{x \in B^m \mid x = \widetilde{x}_0^k \text{ для некоторого } k \in I\}.$$

Вектор x_0 будем называть *центром* обобщённой окрестности $\widetilde{N}(x_0)$.

Отметим, что для различных $k \in I$ построенные локально-оптимальные решения $\widetilde{x}_0^k$ могут совпадать между собой и с центром окрестности x_0, поэтому число элементов в окрестности $\widetilde{N}(x_0)$ может быть меньше чем m. В связи с этим возникает вопрос о строении множества $\widetilde{N}(x_0)$, которое можно охарактеризовать числом элементов в множестве и расстояниями от этих элементов до центра окрестности.

Рассмотрим псевдобулеву функцию $f(x)$ вида

$$f(x) = \sum_{i \in I} f_i x_i + \sum_{j \in J} \min_{i \mid x_i = 1} c_{ij},$$

где $I = \{1, \ldots, m\}$, $J = \{1, \ldots, n\}$, f_i, $i \in I$, — неотрицательные величины, (c_{ij}), $i \in I$, $j \in J$, — матрица с неотрицательными элементами.

Задача минимизации такой псевдобулевой функции эквивалентна задаче размещения предприятий [1, 8]. В случае, когда все величины f_i, $i \in I$, равны единице, а элементы матрицы (c_{ij}), $i \in I$, $j \in J$, принимают два значения: нуль и большое число, эта задача есть эквивалентная форма записи задачи о покрытии множества [1].

Приведём результаты вычислительного эксперимента, показывающего на примере этой задачи, как устроена обобщённая окрестность $\widetilde{N}(x_0)$.

Вычисления проводились для пяти классов задач минимизации указанной псевдобулевой функции. Первые два класса A и C взяты из библиотеки тестовых задач [4]. Примеры из этих классов являются задачами размещения предприятий, трудными для решения алгоритмами неявного перебора. В этих задачах $m = 100$ и $n = 100$. Три других класса взяты из [5, 6]. Эти задачи являются задачами о покрытии множества.

Примеры из класса E имеют размерность $m = 500$, $n = 50$, а заполняемость матрицы (c_{ij}), $i \in I$, $j \in J$, случайно выбранными нулевыми элементами равна 20%. Примеры из классов 4 и 5 имеют размерность $m = 1000$, $n = 50$ и $m = 2000$, $n = 200$, а заполняемость нулями матрицы $(c_{ij}),\ i \in I,\ j \in J$, равна 20%.

Т а б л и ц а 1

Расстояния	Классы		
	A	C	E
0	1,00	0,03	0,00
1–5	20,33	14,83	1,00
6–10	13,76	0,53	0,00
11–15	20,00	1,00	0,40
16–20	23,58	15,56	0,60
21–25	15,67	44,64	4,40
26–30	4,75	22,05	94,00
31–35	0,92	1,33	276,40
36–40	0,00	0,03	123,20

Т а б л и ц а 2

Расстояния	Классы	
	4	5
0	0	0
1100	0,9	1,7
101–200	2,2	2,1
201–300	2,1	2,4
301–400	299,5	1,7
401–500	692,3	1,5
501–600	1,9	3,2
601–700	1,1	2,2
701–800	0	3,8
801–900	0	813,3
901–1000	0	1107,8
1001–1100	0	57,2
1101–1200	0	3,1

В ходе эксперимента вычисляется локально-оптимальное решение x_0, для которого строятся локально-оптимальные решения $\widetilde{x}_0^k$, $k \in I$, определяющие обобщённую окрестность точки x_0. При этом используется стандартный алгоритм локального поиска по окрестности $N_1(x)$ с неполным просмотром окрестности, а в качестве исходной точки для построения локального оптимума x_0 берётся нулевой вектор.

В табл. 1 и 2 для серий из 30 задач классов A и C, серии из 5 задач класса E и серии из 10 задач классов 4 и 5 приводится среднее число точек из множества $\{\widetilde{x}_0^k,\ k \in I\}$, находящихся в заданном интервале расстояний Хэмминга от точки x_0.

Из табл. 1 и 2 видно, что в случае классов A и C лишь незначительная часть точек $\widetilde{x}_0^k,\ k \in I$, совпадает с точкой x_0, а для классов E, 4 и 5 такое совпадение не зафиксировано ни для одного примера. Из табл. 1 следует, что около 60% точек $\widetilde{x}_0^k$, $k \in I$, удалены от точки x_0 на расстояние от 11 до 25 для класса A и от 16 до 30 — для класса C. В случае класса E практически все точки $\widetilde{x}_0^k,\ k \in I$, удалены от x_0 на расстояние больше 26. Аналогичная картина наблюдается и для классов 4 и 5. Из табл. 2 видно, что подавляющая часть точек $\widetilde{x}_0^k$, $k \in I$, находится от точки x_0 на расстоянии от 300 до 500 для класса 4 и от 800 до 1000 — для класса 5.

Важно отметить также, что в каждом из рассмотренных примеров построенные точки $\widetilde{x}_0^k$, $k \in I$, в основном попарно различны: в среднем 97,67% точек $\widetilde{x}_0^k$, $k \in I$, попарно различны для примеров из класса A и 99,41% — для примеров из класса C. Для примеров из классов E, 4 и 5 среди построенных точек $\widetilde{x}_0^k$, $k \in I$, не было совпадающих. Таким образом, можно заключить, что в рассмотренных случаях обобщённая окрестность $\widetilde{N}(x_0)$ не является вырожденной, а число элементов этой окрестности близко к числу m. Кроме того, значительная часть локально-оптимальных решений, образующих окрестность $\widetilde{N}(x_0)$, существенно отличается по числу изменённых компонент от локально-оптимального решения x_0.

2. Алгоритм локального поиска с обобщённой окрестностью

Алгоритм локального поиска по обобщённой окрестности так же, как и стандартный алгоритм локального поиска, представляет собой процедуру последовательного улучшения текущего решения, но в качестве его берётся локально-оптимальное.

Пусть для всякого $x \in B^m$ задана базовая окрестность $N(x)$. Алгоритм локального поиска по обобщённой окрестности при заданной базовой окрестности $N(x)$, $x \in B^m$, состоит из предварительного шага и конечного числа однотипных основных шагов, на каждом из которых рассматривается некоторое текущее локально-оптимальное решение x_0.

На предварительном шаге по заданному (0,1)-вектору с использованием стандартной процедуры локального поиска строится начальное локально-оптимальное решение x_0. После этого начинается основной шаг.

На основном шаге имеется текущее локально-оптимальное решение x_0. Шаг состоит в построении обобщённой окрестности $\widetilde{N}(x_0)$ и поиске элемента $x' \in \widetilde{N}(x_0)$ такого, что $f(x') < f(x_0)$. Если решение x' найти не удаётся, то алгоритм заканчивает работу, результатом которой является текущее локально-оптимальное решение x_0. В противном случае текущее локально-оптимальное решение x_0 заменяется локально-оптимальным решением x' и начинается следующий шаг.

Так же, как и в случае стандартного алгоритма локального поиска, будем использовать два способа выбора решения x' из окрестности $\widetilde{N}(x_0)$, улучшающего текущее решение x_0. При полном просмотре $\widetilde{N}(x_0)$ решение x' должно удовлетворять условиям $f(x') < f(x_0)$ и

$f(x') \leqslant f(x)$ для каждого $x \in \widetilde{N}(x_0)$, а при частичном элементы из $\widetilde{N}(x_0)$ исследуются в некотором заданном порядке и в качестве решения x' выбирается первый элемент $x \in \widetilde{N}(x_0)$ такой, что $f(x) < f(x_0)$.

Приведём результаты вычислительного эксперимента, цель которого — установить, насколько локально-оптимальное решение $\widetilde{x}_0$, найденное алгоритмом локального поиска по обобщённой окрестности, отличается по целевой функции от локально-оптимального решения x_0, найденного на предварительном шаге. В качестве базовой окрестности в алгоритме используется окрестность $N_1(x)$, а выбор улучшающего элемента как в базовой окрестности $N_1(x)$, так и в обобщённой окрестности $\widetilde{N}(x_0)$, производится по правилу полного просмотра окрестности. Вычисления проводились для тех же пяти подзадач A, C, E, 4 и 5.

Таблица 3

Задача	f_1	f_2	$f*$	$f_1/f*$	$f_2/f*$
a332	45134	39120	36154	1,25	1,08
a432	42118	39118	36155	1,16	1,08
a532	42132	36150	36150	1,17	1,00
a632	39159	39152	36162	1,08	1,08
a732	36144	36144	33157	1,09	1,09
a832	42127	36145	36136	1,17	1,00
a932	42125	39126	36133	1,17	1,08
a1032	39126	36127	36127	1,08	1,00
a1132	39143	39143	36163	1,08	1,08
a1232	45123	39146	39123	1,15	1,00
c333	51114	48134	42147	1,21	1,14
c433	45144	45142	42145	1,07	1,07
c533	48130	42106	39177	1,23	1,07
c633	48145	45144	42144	1,14	1,07
c733	48119	45112	42137	1,14	1,07
c833	45153	45140	42144	1,07	1,07
c933	48120	45129	42130	1,14	1,07
c1033	51117	42141	42138	1,21	1,00
c1133	48125	48111	42147	1,14	1,14
c1233	51128	45146	42142	1,21	1,07

В табл. 3 и 4 для каждого примера серий из 10 задач классов A, C, 4 и 5 и серии из 5 задач класса E приводятся следующие величины:

f_1 — значение целевой функции на локально-оптимальном решении x_0, полученном на предварительном шаге алгоритма;

f_2 — значение целевой функции на локально-оптимальном решении $\widetilde{x}_0$, полученном в результате работы алгоритма локального поиска по обобщённой окрестности.

Кроме того, в табл. 3 и 4 для каждого примера приводятся оптимальное значение $f*$ целевой функции и величины относительной погрешности f_1/f^* и f_2/f^* локально-оптимальных решений x_0 и $\widetilde{x}_0$.

Т а б л и ц а 4

Задача	f_1	f_2	f^*	$f_1/\ f*$	$f_2/\ f^*$
scp4.1	1649	435	429	3,84	1,01
scp4.2	1177	514	512	2,30	1,00
scp4.3	1698	516	516	3,29	1,00
scp4.4	1476	496	494	2,99	1,00
scp4.5	1361	514	512	2,66	1,00
scp4.6	1682	561	560	3,00	1,00
scp4.7	1531	432	430	3,56	1,00
scp4.8	1215	496	492	2,47	1,01
scp4.9	1597	650	641	2,49	1,01
scp4.10	1512	514	514	2,94	1,00
scp5.1	1270	261	253	5,02	1,03
scp5.2	1355	307	302	4,49	1,02
scp5.3	1251	230	226	5,54	1,02
scp5.4	1385	242	242	5,72	1,00
scp5.5	1387	211	211	6,57	1,00
scp5.6	1300	213	213	6,10	1,00
scp5.7	1430	298	293	4,88	1,02
scp5.8	1362	293	288	4,73	1,02
scp5.9	1309	279	279	4,69	1,00
scp5.10	1120	269	265	4,23	1,02
scpe.1	6	5	5	1,20	1,00
scpe.2	5	5	5	1,00	1,00
scpe.3	6	5	5	1,20	1,00
scpe.4	6	5	5	1,20	1,00
scpe.5	6	5	5	1,20	1,00

Из табл. 3 и 4 видно, что для большинства примеров рассматриваемых классов использование обобщённой окрестности позволяет улучшить получаемое решение. Для примеров из классов A и C значение целевой функции уменьшается в среднем на 10%. Для задач из классов 4, 5 и E это улучшение более значительное. Для некоторых примеров значение целевой функции на решении x_0 в 5–6 раз больше, чем на решении $\widetilde{x}_0$.

Из табл. 3 и 4 видно также, что для большинства рассматриваемых примеров локально-оптимальное решение $\widetilde{x}_0$, полученное в результате локального поиска по обобщённой окрестности, либо является оптимальным решением, либо близким по значению целевой функции к оптимальному.

3. Алгоритм локального поиска по обобщённой окрестности для задачи конкурентного размещения предприятий

Задача конкурентного размещения предприятий формулируется [2, 3] как задача двухуровневого целочисленного программирования следующего вида:

$$\max_{(x_i),(x_{ij})} \Big\{ -\sum_{i\in I} f_i x_i + \sum_{j\in J} \Big(\sum_{i\in I} p_{ij} x_{ij}\Big)\Big(1 - \sum_{i\in I} \widetilde{z}_{ij}\Big)\Big\}, \tag{1}$$

$$x_i + \sum_{k|i\succ_j k} x_{kj} \leqslant 1, \quad i\in I,\ j\in J, \tag{2}$$

$$x_i \geqslant x_{ij}, \quad i\in I,\ j\in J, \tag{3}$$

$$x_i, x_{ij} \in \{0,1\}, \quad i\in I,\ j\in J, \tag{4}$$

$$((\widetilde{z}_i), (\widetilde{z}_{ij})) - \text{оптимальное решение задачи} \tag{5}$$

$$\max_{(z_i),(z_{ij})} \Big\{ -\sum_{i\in I} g_i z_i + \sum_{i\in I}\sum_{j\in J} p_{ij} z_{ij}\Big\}, \tag{6}$$

$$x_i + z_i + \sum_{k|i\succ_j k} z_{kj} \leqslant 1, \quad i\in I,\ j\in J, \tag{7}$$

$$z_i \geqslant z_{ij}, \quad i\in I,\ j\in J, \tag{8}$$

$$z_i, z_{ij} \in \{0,1\} \quad i\in I,\ j\in J, \tag{9}$$

где $I = \{1,\ldots,m\}$, $J = \{1,\ldots,\ n\}$, (f_i), (g_i), $i\in I$, — векторы с неотрицательными элементами, (p_{ij}), $i\in I$, $j\in J$, — матрица с неотрицательными элементами, $\succ_j$, $j\in J$, — отношение порядка на множестве I.

Данная задача, как и всякая задача двухуровневого программирования, включает задачу верхнего уровня (1)–(4), которую будем обозначать через L, и задачу нижнего уровня (6)–(9), которую будем обозначать через F. Для задачи (1)–(9) используется обозначение $(L,\ F)$.

Обозначим допустимое решение задачи L через $X = ((x_i), (x_{ij}))$, оптимальное решение задачи F при заданном X — через $\widetilde{Z} = ((\widetilde{z}_i), (\widetilde{z}_{ij}))$. Пару $(X, \widetilde{Z})$ будем называть *допустимым решением* задачи (L, F). Обозначим через $L(X, \widetilde{Z})$ значение целевой функции задачи (L, F) на допустимом решении $(X, \widetilde{Z})$.

В качестве оптимального значения целевой функции задачи (L, F) в [3] принято её значение на так называемом оптимальном некооперативном решении. Допустимое решение $(X, \overline{Z})$ называется *допустимым некооперативным решением* задачи (L, F), если $L(X, \overline{Z}) \leqslant L(X, \widetilde{Z})$ для любого допустимого решения $(X, \widetilde{Z})$, а допустимое некооперативное решение $(X^*, \overline{Z}^*)$ называется *оптимальным некооперативным решением* задачи (L, F), если $L(X^*, \overline{Z}^*) \geqslant L(X, \overline{Z})$ для любого допустимого некооперативного решения $(X, \overline{Z})$.

В [3] отмечено, что по произвольному (0,1)-вектору $x = (x_i)$ строится допустимое некооперативное решение $(X, \widetilde{Z})$, где $X = ((x_i), (x_{ij}))$, и однозначно определяется значение целевой функции задачи (L, F).

Таким образом, задача (L, F) сводится к задаче максимизации некоторой псевдобулевой функции $f(x)$, $x \in B^m$. Только функция $f(x)$ в отличие от рассмотренной выше не задана в явном виде, а для вычисления её значения необходимо, в частности, найти решения двух задач целочисленного линейного программирования.

В [3] приведён алгоритм вычисления верхней границы для максимального значения функции $f(x)$ и предложены алгоритмы локального поиска для задачи отыскания максимального значения данной функции. С учётом трудоёмкости вычисления значения этой функции в предложенных алгоритмах используется окрестность специального вида $N_0(x) \subset N_1(x) \cup N_2(x)$, содержащая не более m элементов. Кроме того, в этих алгоритмах в качестве начального решения используется (0,1)-вектор, полученный в результате вычисления верхней границы для максимального значения функции $f(x)$.

Приведём результаты вычислительного эксперимента с алгоритмом локального поиска по обобщённой окрестности для задачи максимизации рассмотренной псевдобулевой функции $f(x)$. В этом алгоритме в качестве базовой окрестности используется $N_0(x)$, а выбор улучшающего элемента при построении локально-оптимальных решений производится по правилу неполного просмотра окрестностей. Выбор улучшающего элемента в обобщённой окрестности $\widetilde{N}_0(x)$ также производится в результате неполного её просмотра за исключением первого шага алгоритма, где улучшающий элемент определяется в результате полного просмотра окрестности. Кроме того, на предварительном шаге в качестве начальной точки используется (0,1)-вектор, получаемый одновременно с вычислением верхней границы для оптимального значения $f(x)$.

Цель данного вычислительного эксперимента, как и двух предыдущих, состоит в том, чтобы выяснить строение обобщённой окрестности $\widetilde{N}_0(x)$, рассматриваемой на первом шаге алгоритма, и оценить, насколько локально-оптимальное решение, найденное на предварительном шаге алгоритма, отличается от локально-оптимального решения, полученного в результате локального поиска по обобщённой окрестности.

Вычисления проводились для примеров задачи конкурентного размещения предприятий, взятых из библиотеки тестовых задач [4]. Использовались примеры из подклассов $A20$, $A30$, $A40$ и $A50$ класса A, для которых число переменных псевдобулевой функции $f(x)$ равно 20, 30, 40 и 50 соответственно. Для примеров из подклассов $A20$ и $A30$ известны оптимальные значения целевых функций.

В табл. 5 и 6 для 20 примеров из каждого подкласса приводятся следующие величины:

f_1 — значение целевой функции на локально-оптимальном решении x_0, полученном на предварительном шаге алгоритма;

f_2 — значение целевой функции на наилучшем локально-оптимальном решении из обобщённой окрестности, рассматриваемой на первом шаге алгоритма;

f_3 — значение целевой функции на локально-оптимальном решении $\widetilde{x}_0$, полученном в результате работы алгоритма локального поиска по обобщённой окрестности;

K— число элементов в обобщённой окрестности на первом шаге алгоритма;

d — среднее расстояние от элементов обобщённой окрестности, построенной на первом шаге алгоритма, до её центра.

Кроме того, в табл. 5 для каждого примера приводятся оптимальное значение f^* целевой функции и величины относительной точности f^*/f_1 и f^*/f_3 локально-оптимальных решений x_0 и $\widetilde{x}_0$.

Из табл. 5 и 6 видно, что исследуемая обобщённая окрестность локально-оптимального решения x_0 не является вырожденной. Элементы обобщённой окрестности в большинстве примеров в среднем отличаются от решения x_0 двумя или тремя компонентами.

Из табл. 5 и 6 видно также, что для большинства примеров локально-оптимальное решение $\widetilde{x}_0$ лучше, чем x_0, а из табл. 5 следует, что использование обобщённой окрестности для подавляющего большинства примеров позволяет получать оптимальное решение.

Таблица 5

Задача	f_1	f_2	f_3	K	d	f^*	f^*/f_1	f^*/f_3
a20-01	62	62	62	3	1,67	62	1,00	1,00
a20-02	53	53	53	2	1,00	53	1,00	1,00
a20-03	39	41	61	5	2,40	61	1,56	1,00
a20-04	23	35	35	8	2,50	35	1,52	1,00
a20-05	58	58	58	2	0,50	58	1,00	1,00
a20-06	41	41	41	2	1,00	51	1,24	1,24
a20-07	5	10	10	6	2,00	10	2,00	1,00
a20-08	27	27	27	4	2,25	38	1,41	1,41
a20-09	8	16	16	6	2,00	16	2,00	1,00
a20-10	9	24	24	9	3,56	29	3,22	1,21
a20-11	30	42	42	7	2,43	42	1,40	1,00
a20-12	15	22	22	10	2,00	22	1,47	1,00
a20-13	25	25	25	3	1,67	25	1,00	1,00
a20-14	51	51	51	2	1,50	51	1,00	1,00
a20-15	40	40	40	4	2,25	40	1,00	1,00
a20-16	12	27	27	6	2,67	27	2,25	1,00
a20-17	30	42	42	7	2,86	42	1,40	1,00
a20-18	50	50	50	2	1,00	50	1,00	1,00
a20-19	40	46	46	8	3,00	46	1,15	1,00
a20-20	32	32	32	6	2,83	32	1,00	1,00
a30-01	62	62	62	10	2,70	67	1,08	1,08
a30-02	37	44	44	9	1,89	44	1,19	1,00
a30-03	53	74	74	9	1,89	74	1,40	1,00
a30-04	58	63	67	8	1,25	67	1,16	1,00
a30-05	100	115	115	6	2,33	115	1,15	1,00
a30-06	41	41	41	6	2,33	41	1,00	1,00
a30-07	59	78	84	12	3,17	84	1,42	1,00
a30-08	44	51	56	8	1,63	56	1,27	1,00
a30-09	83	83	83	4	1,75	83	1,00	1,00
a30-10	54	67	67	16	2,56	67	1,24	1,00
a30-11	72	82	82	7	1,86	82	1,14	1,00
a30-12	61	61	61	7	2,43	71	1,16	1,16
a30-13	31	46	49	8	1,38	49	1,58	1,00
a30-14	44	68	73	6	1,50	73	1,66	1,00
a30-15	21	43	43	10	2,70	45	2,14	1,05
a30-16	76	76	76	4	2,00	76	1,00	1,00
a30-17	95	114	114	6	3,33	114	1,20	1,00
a30-18	55	87	87	10	3,90	87	1,58	1,00
a30-19	39	57	57	8	2,75	57	1,46	1,00
a30-20	52	62	70	7	1,00	70	1,35	1,00

Т а б л и ц а 6

Задача	f_1	f_2	f_3	K	d
a40-01	62	62	62	9	2,33
a40-02	135	135	135	10	2,50
a40-03	81	81	81	2	2,50
a40-04	96	96	96	10	2,20
a40-05	65	94	97	13	3,69
a40-06	81	81	81	8	2,38
a40-07	81	90	96	11	2,27
a40-08	121	125	125	7	2,29
a40-09	81	83	83	7	2,29
a40-10	105	107	107	3	2,33
a40-11	101	101	101	6	1,50
a40-12	47	53	57	7	1,71
a40-13	163	163	163	5	1,80
a40-14	61	67	67	10	1,90
a40-15	93	100	100	9	3,00
a40-16	69	77	77	6	1,67
a40-17	107	132	136	6	2,67
a40-18	131	131	131	10	2,30
a40-19	95	112	112	11	2,45
a40-20	52	54	54	7	3,00
a50-01	128	158	158	8	2,75
a50-02	100	100	100	9	2,11
a50-03	85	106	106	12	2,50
a50-04	85	86	92	9	2,89
a50-05	103	103	103	3	1,33
a50-06	94	122	122	10	4,00
a50-07	68	91	91	12	3,83
a50-08	81	92	92	8	1,75
a50-09	65	93	93	12	2,50
a50-10	80	119	119	15	2,80
a50-11	115	119	120	9	2,22
a50-12	117	127	127	6	1,83
a50-13	54	84	94	21	3,76
a50-14	63	77	77	5	2,00
a50-15	112	122	122	9	2,78
a50-16	85	85	85	14	2,71
a50-17	156	156	156	11	2,00
a50-18	65	97	107	13	2,46
a50-19	99	107	111	13	2,69
a50-20	40	89	89	16	4,25

ЛИТЕРАТУРА

1. **Береснев В. Л.** Дискретные задачи размещения и полиномы от булевых переменных. — Новосибирск: Изд-во Ин-та математики, 2005. — 408 с.
2. **Береснев В. Л.** Верхние оценки для целевых функций дискретных задач конкурентного размещения предприятий // Дискрет. анализ и исслед. операций. — 2008. — Т. 15, № 4. — С. 3–24.
3. **Береснев В. Л., Мельников А. А.** Приближённые алгоритмы для задачи конкурентного размещения предприятий // Дискрет. анализ и исслед. операций. — 2010. — Т. 17, № 6. — С. 3–19.
4. http://www.math.nsc.ru/AP/benchmarks/
5. **Beasly J. E.** An algorithm for set covering problem // Eur. J. Oper. Res. — 1987. — Vol. 31. — P. 85–93.
6. **Beasly J. E.** OR-library: distributing test problems by electronic mail // Eur. J. Oper. Res. — 1990. — Vol. 41. — P. 1069–1072.
7. **Dempe S.** Foundations of bilevel programming. — Dordrecht: Kluwer Acad. Publ., 2002. — 332 p.
8. Discrete location theory (eds. Mirchandani P. B., Francis R. L.). — New York: John Wiley and Sons, 1990. — 555 p.
9. **Dobson G., Karmarkar U.** Competitive location on network // Oper. Res. — 1987. — Vol. 35. — P. 565–574.
10. **Hammer P. L., Rudeanu S.** Boolean method in operations research and related areas. — Berlin: Springer-Verl., 1968. — 330 p.
11. Local Search in Combinatorial Optimization. — Chichester: John Wiley and Sons, 1997. — 512 p.

Береснев Владимир Леонидович,
e-mail: beresnev@math.nsc.ru

Гончаров Евгений Николаевич,
e-mail: goncharov@math.nsc.ru

Мельников Андрей Андреевич,
a.a.melnikov@hotmail.com

Статья поступила
4 апреля 2011 г.