ИНФОРМАТИКА

2008 апрель-июнь № 2

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

УДК 519.854.2

А.А. Карпук

АЛГОРИТМЫ РЕШЕНИЯ МНОГОМЕРНОЙ ЗАДАЧИ О НАЗНАЧЕНИЯХ

Рассматривается многомерная задача о назначениях, полученная из классической задачи о назначениях при дополнительных предположениях, что одного претендента можно назначить на несколько должностей и любое подмножество должностей может оказаться занятым претендентами, несовместимыми между собой на этих должностях. Показывается, что NP-трудная многомерная задача о назначениях сводится к задаче поиска в n-дольном взвешенном гиперграфе максимального полного подгиперграфа, имеющего минимальную сумму весов вершин и ребер. Предлагаются жадные алгоритмы решения задачи и алгоритмы локального поиска в окрестности начального решения.

Введение

Многомерная задача о назначениях была сформулирована в работе [1] как общая математическая модель задачи оптимизации использования радиочастотного ресурса при присвоении частот радиолиниям в виде

$$\sum_{i=1}^{n}\ \sum_{j=1}^{m} a_{ij}x_{ij} + \sum_{k=2}^{l}\frac{1}{k!}\left(\sum_{i_1=1}^{n}\ \sum_{j_1=1}^{m}\ \dots\sum_{i_k=1}^{n}\ \sum_{j_k=1}^{m} c^{(k)}_{i_1 j_1 \dots i_k j_k}\prod_{p=1}^{k} x_{i_p j_p}\right) \to \min; \tag{1}$$

$$\sum_{j=1}^{m} x_{ij} = 1 \text{ для всех } i = \overline{1,n}; \tag{2}$$

$$a_{ij} \ge 0,\ x_{ij} \in \{0,1\} \text{ для всех } i = \overline{1,n},\ j = \overline{1,m}; \tag{3}$$

$$c^{(k)}_{i_1 j_1 i_2 j_2 \dots i_k j_k} \ \ge\ 0 \text{ для всех } k = \overline{2,l}; \tag{4}$$

$$2 \le l \le n \text{ – размерность задачи.}$$

Задача (1)–(4) получается из классической задачи о назначениях m претендентов на n вакантных должностей при дополнительных предположениях, что одного претендента можно назначить на несколько должностей и любые $k \in \{2,\dots,l\}$ должностей могут оказаться занятыми претендентами, несовместимыми между собой на этих должностях. Элемент $a_{ij} \ge 0$ матрицы $A = (a_{ij})$ размерности $n \times m$ равен сумме издержек (штрафа) за назначение претендента с номером j на вакантную должность с номером i. Через $c^{(k)}_{i_1 j_1 i_2 j_2 \dots i_k j_k} \ \ge\ 0$ обозначена величина штрафа за несовместимость между собой k претендентов с необязательно различными номерами $j_1, j_2, \dots, j_k$ при условии, что они будут назначены на должности с номерами $i_1, i_2, \dots, i_k$ соответственно.

Если ограничиться случаем $l = 2$ и положить $a_{ij} = 0$ для всех $i = \overline{1,n},\ j = \overline{1,m}$, то целевая функция (1) имеет вид

$$\sum_{i_1=1}^{n}\ \sum_{j_1=1}^{m}\ \sum_{i_2=1}^{n}\ \sum_{j_2=1}^{m} c^{(2)}_{i_1 j_1 i_2 j_2} x_{i_1 j_1} x_{i_2 j_2} \to \min, \tag{5}$$

и задача (2)–(5) отличается от известной квадратичной задачи о назначениях [2] только тем, что не содержит условия, требующего назначить каждого претендента только на одну должность.

В реальных приложениях значение l значительно меньше n. Например, в задаче оптимизации использования радиочастотного ресурса при присвоении частот радиолиниям [1] для получения решения, приемлемого для практического применения, достаточно ограничиться случаем $l=3$, и целевая функция (1) имеет вид

$$\begin{gathered}\sum_{i=1}^{n}\sum_{j=1}^{m}a_{ij}x_{ij}+\tfrac{1}{2}\sum_{i_1=1}^{n}\sum_{j_1=1}^{m}\sum_{i_2=1}^{n}\sum_{j_2=1}^{m}c^{(2)}_{i_1j_1i_2j_2}x_{i_1j_1}x_{i_2j_2}+\\+\tfrac{1}{6}\sum_{i_1=1}^{n}\sum_{j_1=1}^{m}\sum_{i_2=1}^{n}\sum_{j_2=1}^{m}\sum_{i_3=1}^{n}\sum_{j_3=1}^{m}c^{(3)}_{i_1j_1i_2j_2i_3j_3}x_{i_1j_1}x_{i_2j_2}x_{i_3j_3}\to\min.\end{gathered}\tag{6}$$

Вычислительная сложность классической задачи о назначениях оценивается величиной $O((\max\{n,m\})^3)$, т. е. задача разрешима за полиномиальное время, в то время как квадратичная задача о назначениях является NP-трудной. В настоящей статье рассматриваются вопросы интерпретации многомерной задачи о назначениях в терминах теории графов, оценки вычислительной сложности и построения алгоритмов решения задачи.

1. Графовая интерпретация задачи

Напомним, что гиперграфом называется пара (V,E), где V – непустое конечное множество вершин; E – семейство непустых подмножеств множества V, называемых ребрами. Будем рассматривать только гиперграфы, у которых отсутствуют кратные ребра. Гиперграф называется l-полным ($2\le l\le |V|$), если любое непустое подмножество вершин, мощность которого не превосходит l, есть ребро. Гиперграф называется взвешенным, если в нем каждой вершине и каждому ребру приписано некоторое действительное число, называемое весом. Степенью ребра называется количество вершин, входящих в ребро. Вершины гиперграфа (две или более) называются смежными, если в гиперграфе существует ребро, которому принадлежат эти вершины. Гиперграф называется n-дольным, если множество вершин V можно разбить на n взаимно непересекающихся подмножеств (долей) $V=\{V_1,V_2,...,V_n\}$ таким образом, что любые две вершины одной доли не являются смежными. Непосредственно из этого определения следует, что в n-дольном гиперграфе все k вершин любого ребра степени $k\le n$ принадлежат k различным долям, по одной вершине каждой доле. Гиперграф (V',E') называется частью гиперграфа (V,E), если $V'\subseteq V$ и $E'\subseteq E$. Часть (V',E') гиперграфа (V,E) называется его подгиперграфом, если она получена из исходного гиперграфа путем удаления некоторого подмножества вершин и удаления каждого ребра, в которое входит хотя бы одна удаленная вершина. Подгиперграф (V',E') гиперграфа (V,E) называется максимальным l-полным, если он является l-полным гиперграфом и в гиперграфе (V,E) не существует другого l-полного подгиперграфа (V'',E''), такого, что $|V''|>|V'|$.

Рассмотрим l-полный гиперграф $(2\le l\le n)$, содержащий $n\times m$ вершин $V=\{V_{ij}\mid i=\overline{1,n}, j=\overline{1,m}\}$. Построим часть этого гиперграфа по следующему правилу: из исходного гиперграфа удаляется каждое ребро, содержащее две или более вершины с одинаковым значением первого индекса i. В результате будет получен n-дольный гиперграф (V,E), содержащий вершины $V=\{V_{ij}\mid i=\overline{1,n}, j=\overline{1,m}\}$, причем первый индекс указывает номер доли, а второй – номер вершины в доле. Поскольку исходный гиперграф был l-полным, то полученный n-дольный гиперграф для всех $k=\overline{2,l}$ содержит все возможные ребра степени k, у которых каждая из k вершин принадлежит различной доле. Каждой вершине полученного

n-дольного гиперграфа припишем вес, задаваемый элементом $a_{ij} \geq 0$ $n \times m$-матрицы $A = (a_{ij})$, а каждому ребру степени k, $k = \overline{2, l}$, содержащему вершины $V_{i_1 j_1}, V_{i_2 j_2}, ..., V_{i_k j_k}$, припишем вес $c^{(k)}_{i_1 j_1 i_2 j_2 ... i_k j_k} \geq 0$. Построим подгиперграф (V', E') полученного n-дольного взвешенного гиперграфа, удалив из каждой доли все вершины, кроме одной, выбранной произвольно. Нетрудно убедиться в справедливости следующих утверждений:

гиперграф (V', E') является максимальным l-полным подгиперграфом взвешенного гиперграфа (V, E);

гиперграф (V', E') определяет допустимое решение задачи (1)–(4), полученное по следующему правилу: если $V_{ij} \in V'$, то $x_{ij} = 1$, иначе $x_{ij} = 0$;

любому допустимому решению задачи (1)–(4) соответствует максимальный l-полный подгиперграф (V', E') n-дольного взвешенного гиперграфа (V, E), полученный по следующему правилу: если $x_{ij} = 1$, то $V_{ij} \in V'$, иначе $V_{ij} \notin V'$;

значение целевой функции задачи (1)–(4) для допустимого решения, определяемого гиперграфом (V', E'), равно сумме весов всех вершин и ребер этого гиперграфа.

Таким образом, задача (1)–(4) сводится к задаче поиска в n-дольном взвешенном гиперграфе (V, E) максимального l-полного подгиперграфа (V', E'), имеющего минимальную сумму весов всех вершин и ребер.

В списке NP-полных задач [3] содержится следующая задача. Для заданных $n \times m$-матрицы с неотрицательными элементами $C = (c_{i_1 i_2}), c_{i_1 i_2} = 0$ при $i_1 = i_2$, $m \times m$-матрицы $D = (d_{j_1 j_2})$ и неотрицательного числа B определить, существует ли такое однозначное отображение $f : \{1, 2, ..., n\} \to \{1, 2, ..., m\}$, что $\sum_{i_1=1}^{n} \sum_{i_2=1}^{n} c_{i_1 i_2} d_{f(i_1) f(i_2)} \leq B$. Если положить $c^{(2)}_{i_1 j_1 i_2 j_2} = c_{i_1 i_2} d_{j_1 j_2}$ для всех $i_1 = \overline{1, n}, i_2 = \overline{1, n}, j_1 = \overline{1, m}, j_2 = \overline{1, m}$, то рассматриваемая NP-полная задача сводится к распознавательному варианту задачи (2)–(5), следовательно, распознавательная задача, соответствующая (2)–(5), также является NP-полной. Поскольку распознавательный вариант задачи (2)–(5) есть сужение распознавательного варианта многомерной задачи о назначениях, то последняя также является NP-полной, а исходная многомерная задача о назначениях является NP-трудной.

2. Жадные алгоритмы решения задачи

Можно разработать точные алгоритмы решения многомерной задачи о назначениях с применением техники методов последовательного анализа вариантов, методов построения последовательности решений, методов ветвей и границ и методов динамического программирования. Однако в силу NP-трудности задачи все точные алгоритмы будут иметь экспоненциальную вычислительную сложность, что делает невозможным их практическое применение при достаточно больших значениях n и m. В этом случае для решения NP-трудных задач применяют приближенные алгоритмы поиска локального оптимума. Наиболее распространенными приближенными алгоритмами являются так называемые жадные алгоритмы и алгоритмы локального поиска [4, 5].

Жадный алгоритм производит поиск значений переменных задачи, доставляющих минимум целевой функции, в виде последовательного процесса, на каждом шаге которого выбираются допустимые значения одной или нескольких переменных, дающие минимальное приращение целевой функции. Жадный алгоритм решения многомерной задачи о назначениях последовательно строит максимальный l-полный подгиперграф (V', E') n-дольного взвешенного гиперграфа (V, E). На каждом шаге алгоритма рассматриваются все вершины тех долей гиперграфа (V, E), из которых еще не был произведен выбор, вычисляются стоимости каждой вершины и выбирается вершина с минимальной стоимостью. Выбранная вершина добавляется в множество V', а в множество E' добавляются все ребра гиперграфа (V, E), содержащие выбранную вершину и вершины из V'. Второй индекс вершины, выбранной из доли гиперграфа с номером i, будем обозначать через $f(i)$. Алгоритм состоит из следующих шагов.

Предварительный шаг. Положить $I = \{1, 2, \ldots, n\}$, $I' = \varnothing$, $V' = \varnothing$ и $E' = \varnothing$.

Шаг p, $p = \overline{1, n}$. Выбор вершины из долей гиперграфа (V, E), из которых выбор не был произведен ранее. Для каждой из вершин $V_{ij} \in V, i \in I, j = \overline{1, m}$, вычислить стоимость S_{ij} по формуле

$$S_{ij} = a_{ij} + \sum_{k=2}^{l} \tfrac{1}{(k-1)!} \left(\sum_{i_1 \in I'} \ldots \sum_{i_{k-1} \in I'} c^{(k)}_{i_1 f(i_1) \ldots i_{k-1} f(i_{k-1}) ij} \right). \tag{7}$$

Найти индексы q и μ, такие, что $S_{q\mu} = \min\{S_{ij} \mid i \in I, j = \overline{1, m}\}$. Положить $I = I \setminus q$, $I' = I' \bigcup q$, $f(q) = \mu$, $V' = V' \bigcup V_{qf(q)}$. В множество E' добавить все ребра гиперграфа (V, E), содержащие вершину $V_{qf(q)}$ и вершины из V'. Если $p = n$, то закончить работу, иначе перейти к шагу $(p+1)$.

Если ограничиться случаем $l = 3$, т. е. рассматривать задачу (2)–(4), (6), то формула (7) имеет вид

$$S_{ij} = a_{ij} + \sum_{i_1 \in I'} c^{(2)}_{i_1 f(i_1) ij} + \tfrac{1}{2} \sum_{i_1 \in I'} \sum_{i_2 \in I'} c^{(3)}_{i_1 f(i_1) i_2 f(i_2) ij}. \tag{8}$$

Вычислительная сложность описанного жадного алгоритма при вычислении стоимостей S_{ij} по формуле (8) равна

$$m \sum_{i=1}^{n} (n - i + 1)(i + C^2_{i-1}) = O\left(mn^4\right).$$

Столь высокая вычислительная сложность этого алгоритма объясняется тем, что при выборе очередной вершины для включения в максимальный l-полный подгиперграф (V', E') рассматриваются и оцениваются все вершины тех долей гиперграфа (V, E), из которых еще не был произведен выбор. Для уменьшения вычислительной сложности жадного алгоритма можно предварительно упорядочить доли n-дольного взвешенного гиперграфа (V, E) и на каждом шаге алгоритма производить выбор вершины из очередной доли. Жадный алгоритм с предварительным упорядочиванием долей гиперграфа состоит из следующих шагов.

Предварительный шаг. Перенумеровать доли n-дольного взвешенного гиперграфа (V, E) таким образом, чтобы выполнялось условие: из $i_1 < i_2$ следует $\min\{a_{i_1 j} \mid j = \overline{1, m}\} \le$ $\le \min\{a_{i_2 j} \mid j = \overline{1, m}\}$. Положить $V' = \varnothing$ и $E' = \varnothing$.

Шаг p, $p = \overline{1, n}$. Выбор вершины из доли с индексом p. Для каждой из вершин $V_{pj} \in V, j = \overline{1, m}$, вычислить стоимость S_{pj} по формуле

$$S_{pj} = a_{pj} + \sum_{k=2}^{l} \tfrac{1}{(k-1)!} \left(\sum_{i_1=1}^{p-1} \ldots \sum_{i_{k-1}=1}^{p-1} c^{(k)}_{i_1 f(i_1) \ldots i_{k-1} f(i_{k-1}) pj} \right). \tag{9}$$

Найти индекс μ, такой, что $S_{p\mu} = \min\{S_{pj} \mid j = \overline{1,m}\}$. Положить $f(p) = \mu$, $V' = V' \bigcup V_{pf(p)}$. В множество E' добавить все ребра гиперграфа (V, E), содержащие вершину $V_{pf(p)}$ и вершины из V'. Если $p = n,$ то закончить работу, иначе перейти к выбору вершины из доли с индексом $(p+1)$.

Если ограничиться случаем $l = 3$, т. е. рассматривать задачу (2)–(4), (6), то формула (9) примет вид

$$S_{pj} = a_{pj} + \sum_{i_1=1}^{p-1} c^{(2)}_{i_1 f(i_1) pj} + \tfrac{1}{2} \sum_{i_1=1}^{p-1} \sum_{i_2=1}^{p-1} c^{(3)}_{i_1 f(i_1) i_2 f(i_2) pj}. \qquad (10)$$

Вычислительная сложность жадного алгоритма с предварительным упорядочиванием долей гиперграфа при вычислении стоимостей S_{pj} по формуле (10) равна $O(mn^3)$.

3. Алгоритмы локального поиска

Пусть известно решение многомерной задачи о назначениях (1)–(4), определяемое упорядоченным набором $\langle f(1), f(2), ..., f(n)\rangle$ значений некоторой функции $f : \{1, 2, ..., n\} \to \{1, 2, ..., m\}$. Решение можно получить методом случайного выбора одной вершины из каждой доли *n*-дольного взвешенного гиперграфа (V, E) или построить с помощью жадного алгоритма. Будем говорить, что решение многомерной задачи о назначениях (1)–(4), определяемое упорядоченным набором $\langle g(1), g(2), ..., g(n)\rangle$ значений некоторой функции $g : \{1, 2, ..., n\} \to \{1, 2, ..., m\}$ и не совпадающее с решением $\langle f(1), f(2), ..., f(n)\rangle$, принадлежит τ-окрестности решения $\langle f(1), f(2), ..., f(n)\rangle$, $1 \le \tau \le n$, если наборы чисел $\langle f(1), f(2), ..., f(n)\rangle$ и $\langle g(1), g(2), ..., g(n)\rangle$ различны не более чем в τ позициях. Очевидно, что 1-окрестность любого решения образуют все решения, отличающиеся от него только в одной позиции, а *n*-окрестностью любого решения является множество всех остальных допустимых решений задачи. Значение целевой функции многомерной задачи о назначениях для решения $\langle f(1), f(2), ..., f(n)\rangle$ обозначим через $S(f)$. Решение $\langle f(1), f(2), ..., f(n)\rangle$ называется локальным минимумом по отношению к своей τ-окрестности, если для любого решения $\langle g(1), g(2), ..., g(n)\rangle$ из этой τ-окрестности выполняется условие $S(f) \le S(g)$.

Алгоритм локального поиска начинает свою работу с некоторого решения $\langle f(1), f(2), ..., f(n)\rangle$, выбранного случайно либо построенного жадным алгоритмом. На каждом шаге локального поиска происходит переход от текущего решения к другому решению из его τ-окрестности с меньшим значением целевой функции до тех пор, пока не будет достигнут локальный минимум [5]. Алгоритм локального поиска может быть детерминированным или вероятностным. Все действия детерминированного алгоритма однозначно определяются выбором начального решения $\langle f(1), f(2), ..., f(n)\rangle$ и входными данными задачи. При переходе от текущего решения к другому решению детерминированный алгоритм анализирует определенное множество альтернатив и делает выбор по определенным критериям. Вероятностный алгоритм при переходе от текущего решения к другому решению использует случайный выбор из некоторого множества альтернатив. В настоящей работе ограничимся рассмотрением детерминированных алгоритмов локального поиска.

Для заданного решения $\langle f(1), f(2), ..., f(n)\rangle$ вкладом каждой вершины $V_{if(i)} \in V'$ назовем величину $W_{if(i)}$, вычисляемую по формуле

$$W_{if(i)} = a_{if(i)} + \sum_{k=2}^{l} \frac{1}{(k-1)!}\left(\sum_{i_1=1}^{n} \ldots \sum_{i_{k-1}=1}^{n} c^{(k)}_{i_1 f(i_1)\ldots i_{k-1} f(i_{k-1}) if(i)}\right). \qquad (11)$$

К примеру, формула (11) для вычисления вклада вершины $V_{if(i)} \in V'$ для задачи (2)–(4), (6) имеет вид

$$W_{if(i)} = a_{if(i)} + \sum_{i_1=1}^{n} c^{(2)}_{i_1 f(i_1) if(i)} + \frac{1}{2}\sum_{i_1=1}^{n} \sum_{i_2=1}^{n} c^{(3)}_{i_1 f(i_1) i_2 f(i_2) if(i)}. \qquad (12)$$

Если для некоторого $p \in \overline{1,n}$ выполняется условие $W_{pf(p)} > 0$ и среди вершин доли с индексом p в n-дольном взвешенном гиперграфе (V, E) найдется вершина $V_{pq} \in V$, такая, что $W_{pq} < W_{pf(p)}$, то в результате замены в гиперграфе (V', E') вершины $V_{pf(p)}$ на вершину V_{pq} значение целевой функции многомерной задачи о назначениях уменьшится на величину $(W_{pf(p)} - W_{pq})$. На основе этого утверждения можно построить детерминированный алгоритм локального поиска в 1-окрестности, состоящий из следующих шагов.

1. Для каждой вершины $V_{if(i)} \in V'$, $i = \overline{1,n}$, входящей в решение задачи $\langle f(1), f(2), ..., f(n)\rangle$, вычислить вклад $W_{if(i)}$ по формуле (11). Множество индексов просмотренных вершин P установить в начальное состояние, положив $P = \varnothing$.

2. Найти индекс p, такой, что $W_{pf(p)} = \max\{W_{if(i)} \mid i \notin P, i = \overline{1,n}\}$. Если $W_{pf(p)} = 0$, то закончить работу (достигнут глобальный минимум с нулевым значением целевой функции).

3. Для всех $j = \overline{1,m}$, $j \neq f(p)$ вычислить величины W_{pj} по формуле (11).

4. Найти индекс q, такой, что $W_{pq} = \min\{W_{pj} \mid j \neq f(p), j = \overline{1,m}\}$. Если $W_{pq} < W_{pf(p)}$, то заменить в гиперграфе (V', E') вершину $V_{pf(p)}$ на вершину V_{pq} (положить $f(p) = q$) и перейти к шагу 1, иначе перейти к шагу 5.

5. Положить $P = P \bigcup p$. Если $|P| = n$, то закончить работу (достигнут локальный минимум по отношению к 1-окрестности текущего решения), иначе перейти к шагу 2.

При вычислении величин $W_{if(i)}$ по формуле (12) вычислительная сложность проверки того факта, что текущее решение есть локальный минимум по отношению к своей 1-окрестности, равна $O(mn^3)$. Между тем число переходов от текущего решения к другому решению может быть любым, поэтому теоретически в общем случае рассмотренный алгоритм локального поиска имеет экспоненциальную вычислительную сложность.

Для заданного решения $\langle f(1), f(2), ..., f(n)\rangle$ вкладом двух вершин $V_{if(i)} \in V'$ и $V_{jf(j)} \in V'$, $i = \overline{1,n}$, $j = \overline{1,n}$, $i \neq j$, назовем величину $D_{if(i)jf(j)}$, вычисляемую по формуле

$$D_{if(i)jf(j)} = W_{if(i)} + W_{jf(j)} - c^{(2)}_{if(i)jf(j)} - \\ - \sum_{k=3}^{l} \frac{1}{(k-2)!}\left(\sum_{i_1=1}^{n} \ldots \sum_{i_{k-2}=1}^{n} c^{(k)}_{i_1 f(i_1)\ldots i_{k-2} f(i_{k-2}) if(i) jf(j)}\right). \qquad (13)$$

К примеру, формула (13) для вычисления вклада вершин $V_{if(i)} \in V'$ и $V_{jf(j)} \in V'$ для задачи (2)–(4), (6) имеет вид

$$D_{if(i)jf(j)} = W_{if(i)} + W_{jf(j)} - c^{(2)}_{if(i)jf(j)} - \sum_{i_1=1}^{n} c^{(3)}_{i_1 f(i_1) if(i) jf(j)} . \quad (14)$$

Если для некоторых $\alpha \in \overline{1,n}$ и $\beta \in \overline{1,n}$ выполняется условие $D_{\alpha f(\alpha)\beta f(\beta)} > 0$ и среди вершин долей с индексами α и β в n-дольном взвешенном гиперграфе (V, E) найдутся вершины $V_{\alpha g} \in V$ и $V_{\beta q} \in V$, такие, что $D_{\alpha g \beta q} < D_{\alpha f(\alpha)\beta f(\beta)}$, то в результате замены в гиперграфе (V', E') вершин $V_{\alpha f(\alpha)}$ и $V_{\beta f(\beta)}$ на вершины $V_{\alpha g}$ и $V_{\beta q}$ значение целевой функции многомерной задачи о назначениях уменьшится на величину $(D_{\alpha f(\alpha)\beta f(\beta)} - D_{\alpha g\beta q})$. На основе этого утверждения можно построить детерминированный алгоритм локального поиска в 2-окрестности, состоящий из следующих шагов.

1. К текущему решению задачи $\langle f(1), f(2), ..., f(n) \rangle$ применить детерминированный алгоритм локального поиска в 1-окрестности. Если алгоритмом достигнут глобальный минимум с нулевым значением целевой функции, то закончить работу. Если алгоритмом достигнут локальный минимум по отношению к 1-окрестности текущего решения, то перейти к шагу 2.

2. Множество индексов просмотренных пар вершин A установить в начальное состояние, положив $A = \varnothing$.

3. Найти индексы (α, β), такие, что

$$D_{\alpha f(\alpha)\beta f(\beta)} = \max\{ D_{if(i)jf(j)} \mid (i, j) \notin A, i = \overline{1, n-1}, j = \overline{i+1, n} \} .$$

4. Найти индексы p и q, такие, что

$$D_{\alpha p\beta q} = \min\{ D_{\alpha i\beta j} \mid i \neq f(\alpha), j \neq f(\beta), i = \overline{1,m}, j = \overline{1,m} \} .$$

Если $D_{\alpha p\beta q} < D_{\alpha f(\alpha)\beta f(\beta)}$, то заменить в гиперграфе (V', E') вершины $V_{\alpha f(\alpha)}$ и $V_{\beta f(\beta)}$ на вершины $V_{\alpha p}$ и $V_{\beta q}$ (положить $f(\alpha) = p$ и $f(\beta) = q$) и перейти к шагу 1, иначе перейти к шагу 5.

5. Положить $A = A \bigcup (\alpha, \beta)$. Если $|A| = 0{,}5n(n-1)$, то закончить работу (достигнут локальный минимум по отношению к 2-окрестности текущего решения), иначе перейти к шагу 3.

При вычислении величин $D_{if(i)jf(j)}$ по формуле (14), с учетом вычисления величин $W_{if(i)}$ по формуле (12), вычислительная сложность проверки того факта, что текущее решение есть локальный минимум по отношению к своей 2-окрестности, равна $O(m^2 n^4)$. Число переходов от текущего решения к другому решению также может быть любым, поэтому теоретически в общем случае рассмотренный алгоритм локального поиска имеет экспоненциальную вычислительную сложность. Вычислительная сложность детерминированного алгоритма локального поиска в 2-окрестности даже для задачи (2)–(4), (6) такова, что решение можно получить за приемлемое время только в тех случаях, когда порядок n не превосходит 2 и порядок m не превосходит 3. Этот факт указывает на то, что область применения детерминированных алгоритмов локального поиска в τ-окрестности ограничена величиной $\tau \le 2$. При $\tau > 2$ можно применять только вероятностные алгоритмы локального поиска в τ-окрестности.

Заключение

В статье показано, что NP-трудная многомерная задача о назначениях сводится к задаче поиска в n-дольном взвешенном гиперграфе максимального l-полного подгиперграфа минимального веса. Для решения задачи разработаны жадные алгоритмы, однако эти алгоритмы крайне редко дают оптимальное решение и могут использоваться только для получения начального решения, которое затем можно улучшить, применяя предложенные алгоритмы локального поиска в 1- и 2-окрестности. Алгоритм локального поиска в 1-окрестности можно применять даже в том случае, когда величины n и m имеют порядок 4, более точный алгоритм локального поиска в 2-окрестности можно применять при меньших порядках этих величин.

Под руководством автора разработан комплекс программ, реализующий для решения многомерной задачи о назначениях (2)–(4), (6) жадный алгоритм с предварительным упорядочиванием долей гиперграфа и детерминированный алгоритм локального поиска в 1-окрестности. Разработанный комплекс программ входит в состав программного обеспечения для моделирования электромагнитной совместимости и присвоения частот радиолиниям в системах фиксированной и мобильной радиосвязи [6]. Вычислительные эксперименты показали, что в реальных задачах частотно-территориального планирования радиосвязи при достаточно высокой плотности радиосредств различных радиолиний и ограниченном частотном ресурсе жадный алгоритм с предварительным упорядочиванием долей гиперграфа крайне редко (менее чем в 2 % случаев) приводит к оптимальному решению, зато детерминированный алгоритм локального поиска в 1-окрестности, который начинает работу с решения, полученного жадным алгоритмом, почти в 80 % случаев находит оптимальное решение за приемлемое время.

На подобное расхождение теоретических оценок вычислительной сложности алгоритмов локального поиска с результатами их практического использования указывал Ю.А. Кочетов в работе [5]. Возможным объяснением этого факта является гипотеза о существовании «большой долины», согласно которой в NP-трудных задачах на минимум распределение локальных минимумов в области допустимых решений не является равномерным. Локальные минимумы концентрируются в непосредственной близости к глобальному минимуму, занимая область небольшого диаметра, которую называют «большой долиной». Алгоритм локального поиска в 1-окрестности ведет исходное решение, полученное жадным алгоритмом, к «большой долине» и находит локальный минимум из «большой долины», достаточно близкий к глобальному минимуму или совпадающий с ним.

Результаты статьи докладывались и обсуждались на 13-й Всероссийской конференции «Математическое программирование и приложения» [7]. Автор благодарит заведующего кафедрой математической логики и высшей алгебры Нижегородского государственного университета им. Н.И. Лобачевского В.Н. Шевченко и профессора кафедры дискретного анализа и исследования операций Новосибирского государственного университета Ю.А. Кочетова за активное участие в обсуждении результатов работы и высказанные ценные замечания и предложения.

Список литературы

1. Карпук, А.А. Задача оптимизации использования радиочастотного ресурса при присвоении частот радиолиниям / А.А. Карпук // Информатика. – 2006. – № 4 (12). – С. 5–13.

2. Сергеев, С.И. Квадратичная задача назначения / С.И. Сергеев // Автоматика и телемеханика. – 1999. – № 8. – С. 127–147.

3. Гэри, М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М.: Мир, 1982. – 416 с.

4. Ахо, А. Структуры данных и алгоритмы / А. Ахо, Д. Хопкрофт, Д. Ульман. – М.: Издательский дом «Вильямс», 2000. – 384 с.

5. Кочетов, Ю.А. Вероятностные методы локального поиска для задач дискретной оптимизации / Ю.А. Кочетов // Дискретная математика и ее приложения: сб. лекций молодежных и научных школ по дискретной математике и ее приложениям. – М.: МГУ, 2001. – С. 87–117.

6. Special software for electromagnetic compatibility simulation and frequency assignment to radio sets of fixed-mobile radio communication systems / V. Voloshin [et al.] // Proc. 4th European Symposium on electromagnetic compatibility. – Brugge, 2000. – Vol. 2. – P. 119–122.

7. Карпук, А.А. Задача о назначениях с учетом коллизий / А.А. Карпук // Информационный бюллетень Ассоциации математического программирования: тез. докл. конф. «Математическое программирование и приложения». – Екатеринбург: УрО РАН, 2007. – № 11. – С. 178.

Поступила 24.05.07

Научно-исследовательский институт
средств автоматизации,
Минск, пр. Независимости, 117
e-mail: S3@niisa.iptel.by

A.A. Karpuk

ALGORITHMS FOR MULTIVARIATE ASSIGNMENT PROBLEM

The multivariate assignment problem is considered. This problem is obtained from a classical assignment problem under the following additional assumptions: one applicant can be assigned to several posts; assignments could lead to incompatibility between applicants. The multivariate assignment problem is *NP*-hard problem and is reduced to a problem of maximal full subhypergraph search with a minimal sum of weights of verteces and edges in a weighed n-partite hypergraph. For solving a multivariate assignment problem greedy algorithms and algorithms of local search in a neighbourhood of an initial solution are proposed.