Управляемые системы, ИМ СО АН СССР, 1979 г., выпуск 19.
УДК 519.854.3

О ДВУХ ЗАДАЧАХ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ

С.Е.Гвоздев (Новосибирск)

Рассматриваются задача отыскания связывающего дерева максимального веса с дополнительным ограничением и задача целочисленного линейного программирования, известная под названием двумерной задачи о ранце.

Для их решения предлагаются алгоритмы, построенные на основе метода, изложенного в [1] . Обосновываются априорные оценки отклонения получаемых решений от оптимальных. Для предложенных алгоритмов доказаны оценки трудоемкости (числа операций и объема памяти) .

1. Формулировка задач

Задача 1. Дан связный неориентированный n -вершинный граф $G=(V,U)$ с множеством вершин $V=\{v_j,\ 1\leq j\leq n\}$ и множеством ребер $U=\{u_i,\ 1\leq i\leq m\}$. Каждому ребру u_i приписаны вес $c_i\geq 0$ и длина $a_i\geq 0$. Требуется выделить такой подграф $\mathcal{D}=(V_{\mathcal{D}},U_{\mathcal{D}})$ графа G , называемый остовным деревом, чтобы сумма длин всех ребер подграфа $\mathcal{D}$ не превышала величины A , а сумма весов его ребер была максимальна среди всех остовных деревьев, обладающих указанным выше свойством.

Математически задача записывается следующим образом:

$$\sum_{u_i\in U_{\mathcal{D}}} c_i \longrightarrow max, \qquad /1/$$

$$\mathcal{D}\in G, \qquad /2/$$

$$\sum_{u_i\in U_{\mathcal{D}}} a_i \leq A . \qquad /3/$$

Хотя задачу /1/-/2/ можно решить эффективно [2] , для точного решения задачи 1 эффективные алгоритмы неизвестны. В данной статье под эффективным алгоритмом понимается алгоритм, трудоемкость которого является полиномом от длины записи исходной информации (в смысле Карпа [3]).

Задача 2. Пусть $A,\ B,\ a_j,\ b_j,\ c_j,\ j=\overline{1,n}$,- неотрицательные целые числа. Требуется найти вектор $x=(x_1,x_2,\dots,x_n)$,удовлетворяющий условиям:

$$\sum_{j=1}^{n} a_j x_j \leq A , \qquad /4/$$

$$\sum_{j=1}^{n} b_j x_j \leqslant B, \qquad /5/$$

$$x_j \in \{0,1\}, \quad j=\overline{1,n}, \qquad /6/$$

на котором достигается максимум функции

$$\sum_{j=1}^{n} c_j x_j . \qquad /7/$$

Есть основания предполагать, что эффективный алгоритм получения точного решения задачи 2 принципиально не может быть создан [3] .

Для решения задачи 2 может быть применен метод динамического программирования с трудоёмкостью $\mathcal{K} \sim n^2 \cdot A \cdot B$ операций[x] и памятью $\Pi \sim A \cdot B$ ячеек [4] . Однако, в смысле Карпа, он не является эффективным.

Задача 2 находит практическое применение при решении многих экономических задач. Чаще всего она рассматривается как вспомогательная при решении более сложных задач, для получения соответствующих оценок в методах типа плавного перебора. Поэтому изучение малотрудоемких приближенных алгоритмов с гарантированной оценкой отклонения получаемых решений от оптимального является актуальным.

2. Описание алгоритмов приближенного решения задач I и 2

Алгоритм нахождения приближенного решения задачи I основан на идее, изложенной в[I], и состоит в вычислении функции

$$\varphi(\lambda) = \max_{\mathcal{D}\in G}\Big(\sum_{u_i\in U_{\mathcal{D}}} c_i - \lambda \sum_{u_i\in U_{\mathcal{D}}} a_i\Big) = \max_{\mathcal{D}\in G} \sum_{u_i\in U_{\mathcal{D}}} (c_i - \lambda a_i) =$$

$$= \max_{\mathcal{D}\in G} \sum_{u_i\in U_{\mathcal{D}}} \tilde{c}_i(\lambda) = \sum_{u_i\in U_{\mathcal{D}_\lambda}} \tilde{c}_i(\lambda) = c^{(1)}(\lambda) - \lambda a^{(1)}(\lambda)$$

при значениях $\lambda \geqslant 0$, где $c^{(1)}(\lambda) = \sum_{u_i\in U_{\mathcal{D}_\lambda}} c_i$ и $a^{(1)}(\lambda) = \sum_{u_i\in U_{\mathcal{D}_\lambda}} a_i$.

Это позволяет свести решение задачи I к решению последовательности задач $\max_{\mathcal{D}\in G} \sum_{u_i\in U_{\mathcal{D}}} \tilde{c}_i(\lambda)$ при различных λ . В дальнейшем для упрощения записи вместо $\sum_{u_i\in U_{\mathcal{D}}}$ будем писать $\sum_{i\in\mathcal{D}}$, подразумевая при этом, что суммирование происходит по $(n-1)$-му ребру, принадлежащему основному дереву $\mathcal{D}$.

При каких значениях λ нужно вычислять $\varphi(\lambda)$? Чтобы ответить на этот вопрос, рассмотрим графики функций $c^{(1)}(\lambda)$ и $a^{(1)}(\lambda)$. Нетрудно заметить, что функции $c^{(1)}(\lambda)$ и $a^{(1)}(\lambda)$ невозрастающие и кусочно-линейные, а точки разрыва этих функций содержатся в мно-

---

[x] Под операцией всюду понимается сложение либо сравнение двух чисел.

жестве

$$\mathcal{L}_1=\left\{ \lambda_{il} \middle| \lambda_{il}=\frac{c_i-c_l}{a_i-a_l},\ 1\leq i, l\leq m \right\}.$$

Введем величины $\lambda^{(1)}_{max}=\max\limits_{\lambda_{il}\in\mathcal{L}_1}\lambda_{il}$ и $\Delta\lambda^{(1)}=\min\limits_{\lambda_{il},\lambda_{i_1l_1}\in\mathcal{L}_1}\left\{|\lambda_{il}-\lambda_{i_1l_1}| \,\middle|\, \lambda_{il}\neq\lambda_{i_1l_1}\right\}$.

Очевидно, что для любого $\lambda\geqslant\lambda^{(1)}_{max}$ имеем $c^{(1)}(\lambda)=c^{(1)}(\lambda_{max})$ и $a^{(1)}(\lambda)=a^{(1)}(\lambda_{max})$ и, кроме того, интервалу $[\lambda, \lambda+\Delta\lambda^{(1)}]$ $(\lambda\geqslant 0)$ принадлежит не более одной точки из множества $\mathcal{L}_1$. Для приближенного решения задачи I достаточно найти значение параметра λ°, удовлетворяющего условию $a^{(1)}(\lambda^{\circ})\leqslant A< a^{(1)}(\lambda^{\circ}-\Delta\lambda^{(1)})$. Но прежде чем описать алгоритм нахождения λ°, покажем, что дерево $\mathcal{D}_{\lambda^{\circ}}$ является оптимальным решением задачи I, если $a^{(1)}(\lambda^{\circ})=A$.

Действительно, предположим противное, т.е. существует дерево $\hat{\mathcal{D}}$, являющееся решением задачи I, такое, что $\sum\limits_{i\in\hat{\mathcal{D}}}c_i>\sum\limits_{i\in\mathcal{D}_{\lambda^{\circ}}}c_i$. Но в силу определения функции $\varphi(\lambda)$ имеем

$$\sum_{i\in\mathcal{D}_{\lambda^{\circ}}}c_i-\lambda^{\circ}\sum_{i\in\mathcal{D}_{\lambda^{\circ}}}a_i=\sum_{i\in\mathcal{D}_{\lambda^{\circ}}}c_i-\lambda^{\circ}A=\varphi(\lambda^{\circ})\geqslant$$

$$\geqslant\sum_{i\in\hat{\mathcal{D}}}c_i-\lambda^{\circ}\sum_{i\in\hat{\mathcal{D}}}a_i\geqslant\sum_{i\in\hat{\mathcal{D}}}c_i-\lambda^{\circ}A.$$

Следовательно, $\sum\limits_{i\in\mathcal{D}_{\lambda^{\circ}}}c_i\geqslant\sum\limits_{i\in\hat{\mathcal{D}}}c_i$, что противоречит предположению. Таким образом, в силу невозрастания функций $c^{(1)}(\lambda)$ и $a^{(1)}(\lambda)$ вычисление $\varphi(\lambda)$ достаточно осуществлять лишь в точках

$$\lambda_1=\frac{\lambda_{max}}{2},\quad \lambda_K=\frac{\lambda'_{K-1}+\lambda''_{K-1}}{2},\quad K\geqslant 2, \qquad /8/$$

где $\lambda_{max}=\lambda^{(1)}_{max}$, а

$$\lambda''_{K-1}=\min_{1\leqslant l\leqslant K-1}\left\{\lambda_l \middle| A-a^{(1)}(\lambda_l)>0\right\},\quad \lambda'_{K-1}=\max_{1\leqslant l\leqslant K-1}\left\{a^{(1)}(\lambda_l)-A>0\right\}.$$

Причем естественно ввести следующие доопределения: $\max\{\emptyset\}=0$, $\min\{\emptyset\}=\lambda_{max}$. Критерием окончания вычисления функции $\varphi(\lambda)$ можеть служить выполнение равенства $a^{(1)}(\lambda_K)=A$ или неравенства $\lambda''_K-\lambda'_K\leqslant\Delta\lambda^{(1)}$. В первом случае λ° полагается равным λ_K, во втором - равным λ''_K.

Проведем аналогичные рассуждения для задачи 2. Запишем для нее функцию $\varphi(\lambda)$:

$$\varphi(\lambda)=\max_{(5),(6)}\left(\sum_{j=1}^{n}c_jx_j-\lambda\cdot\sum_{j=1}^{n}a_jx_j\right)=\max_{(5),(6)}\sum_{j=1}^{n}(c_j-\lambda a_j)x_j. \qquad /9/$$

Отыскание точного значения функции /9/ является столь же трудной задачей, что и решение исходной [3], поэтому ограничимся приближенным вычислением ее значений $\tilde{\varphi}(\lambda)$. При фиксирован-

ном λ будем считать, что отношения $\frac{c_j-\lambda a_j}{b_j}$, $j=\overline{1,n}$, упорядочены по невозрастанию. Найдя вектор $\tilde{x}(\lambda)=(\underset{m(\lambda)}{1,\dots,1},\underset{m(\lambda)+1}{0},\dots,0)$, где индекс $m(\lambda)$ определяется условием $\sum_{j=1}^{m(\lambda)} b_j \leqslant B < \sum_{j=1}^{m(\lambda)+1} b_j$, положим

$$\tilde{\varphi}(\lambda)=\sum_{j=1}^{m(\lambda)} c_j-\lambda\sum_{j=1}^{m(\lambda)} a_j=c^{(2)}(\lambda)-\lambda a^{(2)}(\lambda).$$

Рассмотрим графики функций $c^{(2)}(\lambda)$ и $a^{(2)}(\lambda)$: эти функции не возрастают, кусочно-линейны и точки их разрыва принадлежат множеству

$$\mathcal{L}_2=\left\{\lambda_{ij} \,/\, \lambda_{ij}=\frac{b_i c_j-b_j c_i}{b_i a_j-b_j a_i}, \quad 1\leqslant i,\ j\leqslant n\right\}.$$

Обозначим $\lambda^{(2)}_{max}=\max\limits_{\lambda_{ij}\in\mathcal{L}_2}\lambda_{ij}$, $\Delta\lambda^{(2)}=\min\limits_{1\leqslant i,i_1,j,j_1\leqslant n}\left\{|\lambda_{ij}-\lambda_{i_1j_1}| \,|\, \lambda_{ij}\neq\lambda_{i_1j_1}\right\}$.

Полагая $\lambda_{max}=\lambda^{(2)}_{max}$ и в определении λ''_{k-1} и λ'_{k-1} заменяя значения $a^{(1)}(\lambda)$ на $a^{(2)}(\lambda)$, по формуле /8/ найдем точки λ_k, в которых может понадобиться вычисление значений функции $\tilde{\varphi}(\lambda)$. Критерием окончания счета может служить выполнение неравенства $\lambda''_k-\lambda'_k<\Delta\lambda^{(2)}$.

Итак, задача I (задача 2) может быть решена приближенно алгоритмом, состоящим в вычислении $\varphi(\lambda)(\tilde{\varphi}(\lambda))$ при значениях параметра λ, определяемых соотношением /8/. Счет производится до выполнения равенства $a^{(1)}(\lambda_k)=A$ либо неравенства $\lambda''_k-\lambda'_k<$ $<\Delta\lambda^{(1)}(\lambda''_k-\lambda'_k\leqslant\Delta\lambda^{(2)})$. В качестве приближенного решения задачи I (задачи 2) достаточно взять дерево $\mathcal{D}_{\lambda^0}$ (вектор $\tilde{x}(\lambda''_k)$).

3. Об оценках точности полученных решений

Проведем рассуждения, предполагая, что все $\lambda_{i\ell}(\lambda_{ij})$, принадлежащие множеству $\mathcal{L}_1(\mathcal{L}_2)$, различны. Затем покажем, что это ограничение может быть снято.

Теорема I. Если $\lambda''_k-\lambda'_k\leqslant\Delta\lambda^{(1)}$, то

$$\sum_{i\in\mathcal{D}_{\lambda'_k}} c_i-\sum_{i\in\mathcal{D}_{\lambda''_k}} c_i\leqslant c^{(1)}_{max}-c^{(1)}_{min},$$

где $c^{(1)}_{max}=\max\limits_{1\leqslant i\leqslant m} c_i$ и $c^{(1)}_{min}=\min\limits_{1\leqslant i\leqslant m} c_i$.

Доказательство. В силу определения λ'_k и λ''_k имеем

$$\varphi(\lambda'_k)=\sum_{i\in\mathcal{D}_{\lambda'_k}} c_i-\lambda'_k\sum_{i\in\mathcal{D}_{\lambda'_k}} a_i, \quad \sum_{i\in\mathcal{D}_{\lambda'_k}} a_i\geqslant A, \qquad /10/$$

$$\varphi(\lambda''_k)=\sum_{i\in\mathcal{D}_{\lambda''_k}} c_i-\lambda''_k\sum_{i\in\mathcal{D}_{\lambda''_k}} a_i, \quad \sum_{i\in\mathcal{D}_{\lambda''_k}} a_i\leqslant A. \qquad /11/$$

Следовательно, $\mathcal{D}_{\lambda'_K} \neq \mathcal{D}_{\lambda''_K}$, Очевидно, что

$$U_{\mathcal{D}_{\lambda''_K}} = (U_{\mathcal{D}_{\lambda'_K}} \setminus (U_{\mathcal{D}_{\lambda'_K}} \setminus (U_{\mathcal{D}_{\lambda''_K}} \cap U_{\mathcal{D}_{\lambda'_K}}))) \cup (U_{\mathcal{D}_{\lambda''_K}} \setminus (U_{\mathcal{D}_{\lambda''_K}} \cap U_{\mathcal{D}_{\lambda'_K}})).$$

Основываясь на теореме I работы [5] , нетрудно показать,что

$$|U_{\mathcal{D}_{\lambda'_K}} \setminus (U_{\mathcal{D}_{\lambda''_K}} \cap U_{\mathcal{D}_{\lambda'_K}})| = 1 \quad \text{и} \quad |U_{\mathcal{D}_{\lambda''_K}} \setminus (U_{\mathcal{D}_{\lambda''_K}} \cap U_{\mathcal{D}_{\lambda'_K}})| = 1,$$

т.е. существуют ребра $\{j'_1, j'_2\} \in U_{\mathcal{D}_{\lambda'_K}}$ и $\{j''_1, j''_2\} \in U_{\mathcal{D}_{\lambda''_K}}$ с весами $c_{i'}$ и $c_{i''}$ соответственно, удовлетворяющие равенству

$$U_{\mathcal{D}_{\lambda''_K}} = (U_{\mathcal{D}_{\lambda'_K}} \setminus \{j'_1, j'_2\}) \cup \{j''_1, j''_2\}.$$

Следовательно, $\sum\limits_{i \in \mathcal{D}_{\lambda''_K}} c_i = \sum\limits_{i \in \mathcal{D}_{\lambda'_K}} c_i - c_{i'} + c_{i''}$ или

$$\sum_{i \in \mathcal{D}_{\lambda'_K}} c_i - \sum_{i \in \mathcal{D}_{\lambda''_K}} c_i = c_{i'} - c_{i''} \leqslant c^{(1)}_{max} - c^{(1)}_{min} .$$

Теорема I доказана, и тем самым, учитывая, что максимальное значение целевой функции задачи I заключено между величинами $\sum\limits_{i \in \mathcal{D}_{\lambda'_K}} c_i$ и $\sum\limits_{i \in \mathcal{D}_{\lambda''_K}} c_i$, доказана оценка отклонения приближенного решения задачи I от оптимального.

Для оценки отклонения приближенного решения задачи 2 от оптимального покажем, что имеет место

Теорема 2. Если $\lambda''_K - \lambda'_K \leqslant \Delta\lambda^{(2)}$, то

$$\sum_{j=1}^{n} c_j x^*_j - \sum_{j=1}^{m(\lambda''_K)} c_j \leqslant 2\, c^{(2)}_{max}, \tag{12}$$

где $c^{(2)}_{max} = \max\limits_{1 \leqslant j \leqslant n} c_j$ и $(x^*_1, \ldots, x^*_n)$ - оптимальное решение задачи 2.

Доказательство. Достаточно показать справедливость соотношений

$$\sum_{j=1}^{n} c_j x^*_j - \sum_{j=1}^{m(\lambda'_K)} c_j \leqslant c^{(2)}_{max}, \tag{13}$$

$$\sum_{j=1}^{m(\lambda'_K)} c_j - \sum_{j=1}^{m(\lambda''_K)} c_j \leqslant c^{(2)}_{max}. \tag{14}$$

По определению /9/ функции $\varphi(\lambda)$, для любого $\lambda \geqslant 0$

$$\varphi(\lambda) = \max_{(5),(6)} \Big(\sum_{j=1}^{n} c_j x_j - \lambda \sum_{j=1}^{n} a_j x_j \Big) \geqslant \sum_{j=1}^{n} c_j x^*_j - \lambda \sum_{j=1}^{n} a_j x^*_j .$$

Из алгоритма приближенного вычисления $\varphi(\lambda)$ следует существование индекса $\xi = m(\lambda'_K) + 1$ такого, что

$$\varphi(\lambda'_K) - \tilde{\varphi}(\lambda'_K) \leqslant c_\xi - \lambda'_K a_\xi .$$

Тогда

$$\sum_{j=1}^{n} c_j x^*_j - \lambda'_K \sum_{j=1}^{n} a_j x^*_j - \Big(\sum_{j=1}^{m(\lambda'_K)} c_j - \lambda'_K \sum_{j=1}^{m(\lambda'_K)} a_j \Big) \leqslant c_\xi - \lambda'_K a_\xi .$$

Далее, $a_\xi \geqslant 0$ по условию задачи 2; $\lambda'_\kappa \geqslant 0$ согласно соотношению /8/; $\sum_{j=1}^{m(\lambda'_\kappa)} a_j > A$ по определению λ'_κ. Следовательно,

$$\sum_{j=1}^{n} c_j x_j^* - \sum_{j=1}^{m(\lambda'_\kappa)} c_j \leqslant c_\xi - \lambda'_\kappa (a_\xi - \sum_{j=1}^{n} a_j x_j^* + \sum_{j=1}^{m(\lambda''_\kappa)} a_j) \leqslant C_{max}^{(2)},$$

т.е. имеет место неравенство /I3/.

Докажем соотношение /I4/. В силу неравенств $\lambda''_\kappa - \lambda'_\kappa \leqslant \Delta\lambda^{(2)}$ и $\lambda_{ij} \neq \lambda_{i_1 j_1}$, где $i \neq i_1$, $j \neq j_1$; λ_{ij}, $\lambda_{i_1 j_1} \in \mathcal{L}_2$, получаем

$$\sum_{j=1}^{m(\lambda''_\kappa)} c_j = \sum_{j=1}^{m(\lambda'_\kappa)} c_j + c_{m(\lambda'_\kappa)+1} - c_{m(\lambda'_\kappa)},$$

а значит,

$$\sum_{j=1}^{m(\lambda'_\kappa)} c_j - \sum_{j=1}^{m(\lambda''_\kappa)} c_j = c_{m(\lambda'_\kappa)} - c_{m(\lambda'_\kappa)+1} \leqslant C_{max}^{(2)}.$$

Таким образом, установлена справедливость неравенств /I3/, /I4/ и теоремы 2.

На примере задачи I покажем, как может быть снято ограничение, заключающееся в недопустимости равенства $\lambda_{i\ell} = \lambda_{i_1 \ell_1}$, $i = i_1$, $\ell = \ell_1$; $\lambda_{i\ell}$, $\lambda_{i_1\ell_1} \in \mathcal{L}_1$.

Для этого необходимо продолжать работу алгоритма по нахождению решения данной задачи, если найдется такой индекс, что

$$c_{i_0} - \lambda^* a_{i_0} > c_{i_1} - \lambda^* a_{i_1} = \ldots = c_{i_\ell} - \lambda^* a_{i_\ell} > c_{i_{\ell+1}} - \lambda^* a_{i_{\ell+1}},$$

причем для λ''_κ и λ'_κ соблюдены неравенства

$$c_{i_1} - \lambda''_\kappa a_{i_1} > c_{i_\ell} - \lambda''_\kappa a_{i_\ell} \quad \text{и} \quad c_{i_1} - \lambda'_\kappa a_{i_1} < c_{i_\ell} - \lambda'_\kappa a_{i_\ell}.$$

Параметр λ^* определяется соотношением

$$\lambda^* = \frac{\sum_{i \in \mathcal{D}_{\lambda'_\kappa}} c_i - \sum_{i \in \mathcal{D}_{\lambda''_\kappa}} c_i}{\sum_{i \in \mathcal{D}_{\lambda'_\kappa}} a_i - \sum_{i \in \mathcal{D}_{\lambda''_\kappa}} a_i},$$ где $\mathcal{D}_{\lambda''_\kappa}$ и $\mathcal{D}_{\lambda'_\kappa}$ удовлетворяют /I0/ и /II/ соответственно.

Будем вычислять $\max\limits_{\mathcal{D} \in G} \sum\limits_{i \in \mathcal{D}} \tilde{c}_i (\lambda^*, q_\kappa)$, где

$$\tilde{c}_{i_\xi}(\lambda^*, q_\kappa) = \begin{cases} \tilde{c}_{i_\xi}(\lambda^*), & \text{если } \xi \leqslant 0 \text{ либо } \xi \geqslant \ell+1, \\ \tilde{c}_{i_{\ell-\xi+1}}(\lambda^*), & \text{если } 1 \leqslant \xi \leqslant \ell - q_\kappa + 1, \\ \tilde{c}_{i_{\xi-(\ell-q_\kappa+1)}}(\lambda^*), & \text{если } \ell - q_\kappa + 2 \leqslant \xi \leqslant \ell, \end{cases}$$

по определению. Обозначим $\max\limits_{\mathcal{D} \in G} \sum\limits_{i \in \mathcal{D}} \tilde{c}_i(\lambda^*, q_\kappa) = \sum\limits_{i \in \mathcal{D}_{\lambda^* q_\kappa}} c_i - \lambda^* \sum\limits_{i \in \mathcal{D}_{\lambda^* q_\kappa}} a_i$.

Здесь

$$q_1 = \left]\frac{\ell}{2}\right[, \quad q_\kappa = \left]\frac{q'_{\kappa-1} + q''_{\kappa-1}}{2}\right[, \quad \kappa \geqslant 2,$$

где

$$\tau''_{k-1} = \min_{1\le p\le k-1}\left\{\tau_p \,\Big|\, \sum_{i\in \mathcal{D}_{\lambda^*\tau_p}} a_i - A<0\right\},$$

$$\tau'_{k-1} = \max_{1\le p< k-1}\left\{\tau_p \,\Big|\, \sum_{i\in \mathcal{D}_{\lambda^*\tau_p}} a_i - A>0\right\}$$

и $\min\{\phi\} = \ell+1, \quad \max\{\phi\}=0.$

Найдя $\tau''_{\hat k}$ и $\tau'_{\hat k}$, $\hat k =]\log \ell[$ *) введем обозначения:

$$\tilde c_{t_\xi}(\lambda^*,\mu_\rho) = \begin{cases} \tilde c_{t_\xi}(\lambda^*, \tau'_{\hat k}), & \text{если } \xi\le \ell-\tau'_{\hat k} \text{ либо } \xi\ge \mu_\rho+1,\\ \tilde c_{t_{\xi+1}}(\lambda^*, \tau'_{\hat k}), & \text{если } \ell-\tau'_{\hat k}+1\le\xi<\mu_\rho-1,\\ \tilde c_{t_{\tau'_{\hat k}}}(\lambda^*, \tau'_{\hat k}), & \text{если } \xi=\mu_\rho \end{cases}$$

и

$$\max_{\mathcal{D}\in G}\sum_{i\in\mathcal{D}}\tilde c_i(\lambda^*,\mu_\rho) = \sum_{i\in\mathcal{D}_{\lambda^*\mu_\rho}} c_i - \lambda^*\sum_{i\in\mathcal{D}_{\lambda^*\mu_\rho}} a_i .$$

Здесь

$$\mu_1 = \ell-\tau'_{\hat k}+1+\left]\frac{\tau'_{\hat k}-1}{2}\right[, \quad \mu_\rho=\mu_{\rho-1}+\left]\frac{\tau'_{\hat k}-1}{2}\right[\cdot \operatorname{sign}\Big(\sum_{i\in\mathcal{D}_{\lambda^*\mu_{\rho-1}}} a_i - A\Big), \rho\ge 2.$$

Далее, будем вычислять $\max\limits_{\mathcal{D}\in G}\sum\limits_{i\in\mathcal{D}}\tilde c_i(\lambda^*,\mu_\rho)$ до тех пор, пока $\rho\le\hat\rho$, где $\hat\rho =](\tau'_{\hat k}-1)[$. Деревья $\mathcal{D}_{\lambda^*\mu_{\hat\rho}}$ и $\mathcal{D}_{\lambda^*\mu_{\check\rho}}$ $(\check\rho=\hat\rho-1)$ будут отличаться лишь двумя ребрами: $\{j'_1, j''_2\}$ и $\{j''_1, j''_2\}$, т.е. путем вычисления S раз $(S\subseteq\,]\log\ell[)$ величин $\max\limits_{\mathcal{D}\in G}\sum\limits_{i\in\mathcal{D}}\tilde c_i(\lambda^*,\tau_k)$ и $\max\limits_{\mathcal{D}\in G}\sum\limits_{i\in\mathcal{D}}\tilde c_i(\lambda^*,\mu_\rho)$ снято ограничение, не допускающее равенства $\lambda_{i\ell}=\lambda_{i_1\ell_1}$ при $i=i_1$, $\ell=\ell_1$.

З а м е ч а н и е. В практических расчётах не обязательно пользоваться только что разработанной схемой продолжения алгоритма. Целесообразнее незначительно изменить значения величин c_{i_1}, a_{i_1}; c_{i_2}, a_{i_2}; ... ; c_{i_ℓ}, a_{i_ℓ} с тем, чтобы среди $\lambda_{i_\xi i_{\xi'}}$ $\left(\lambda_{i_\xi i_{\xi'}} = \dfrac{c_{i_\xi}-c_{i_{\xi'}}}{a_{i_\xi}-a_{i_{\xi'}}}\right)$, $1\le\xi$, $\xi'\le\ell$, $\xi\ne\xi'$, не встречалось равных значений. Это можно сделать, например, следующим образом. Рассмотрим две окружности, проходящие через точки (c_{i_0}, a_{i_0}), (c_{i_1}, a_{i_1}), (c_{i_ℓ}, a_{i_ℓ}) и (c_{i_1}, a_{i_1}), (c_{i_ℓ}, a_{i_ℓ}), $(c_{i_{\ell+1}}, a_{i_{\ell+1}})$. Пусть (α_1,β_1), (α_2,β_2) – центры этих окружностей, R_1 и R_2 – их радиусы. Обозначим $\dfrac{\alpha}{\beta}=\min\left\{\dfrac{\alpha_1}{\beta_1}, \dfrac{\alpha_2}{\beta_2}\right\}$, $R=\max\{R_1, R_2\}$.

Новые величины (c'_{i_ξ}, a'_{i_ξ}) $(\xi=\overline{2,\ell-1})$, используемые вместо (c_{i_ξ}, a_{i_ξ}), определим следующим образом: $c'_{i_\xi}=c_{i_\xi}\cdot\gamma_{i_\xi}$ и $a'_{i_\xi}=a_{i_\xi}\cdot\gamma_{i_\xi}$, где

---

*) Можно показать, что при $\hat k =]\log\ell[$ имеет место равенство $\tau''_{\hat k}-\tau'_{\hat k}=1$.

$\gamma_{i_\xi} = (\alpha \cdot a_{i_\xi} + \beta \cdot c_{i_\xi}) + ((\alpha \cdot a_{i_\xi} + \beta c_{i_\xi})^2 - (c_{i_\xi}^2 + a_{i_\xi}^2)\cdot(\alpha^2+\beta^2-R^2))^{\frac{1}{2}}(c_{i_\xi}^2 + a_{i_\xi}^2)^{-1}$.

Нетрудно проверить, что среди отношений $\dfrac{c'_{i_\xi} - c'_{i_{\xi'}}}{a'_{i_\xi} - a'_{i_{\xi'}}}$, $1\leq \xi, \xi' \leq l, \xi \neq \xi'$, нет равных.

4. О трудоемкости нахождения приближенных решений задач I и 2

Для оценки трудоемкости решения задачи I необходимо знать трудоемкость решения задачи $\max\limits_{\mathcal{D}\in G} \sum\limits_{i\in\mathcal{D}} \tilde{c}_i(\lambda)$ при фиксированном λ и число значений λ, при которых достаточно решать эту задачу. Найти $\max\limits_{\mathcal{D}\in G} \sum\limits_{i\in\mathcal{D}} \tilde{c}_i(\lambda)$ можно несколькими методами[2,6] одним из удачных является алгоритм Прима [2] с оценкой числа операций $\mathcal{K} \sim n^2$ и объемом памяти $\Pi \sim n$ ячеек. Каково количество обращений к работе алгоритма Прима? Ответим на этот вопрос,считая, что веса c_i ребер графа G и их длины a_i - целые. По определению величин $\lambda^{(1)}_{max}$ и $\Delta\lambda^{(1)}$, с учетом целочисленности c_i и a_i, имеем $\lambda^{(1)}_{max} \leq c^{(1)}_{max}$ и $\Delta\lambda^{(1)} \geq (a^{(1)}_{max})^2$, где $a^{(1)}_{max} = \max\limits_{1\leq i\leq m} a_i$. Так как при вычислении λ_K по соотношению /8/ применяется метод дихотомии, трудоемкость определения параметра λ''_K, удовлетворяющего соотношению $\lambda''_K - \lambda'_K \leq \Delta\lambda^{(1)}$, может быть оценена величиной $]\log(c^{(1)}_{max}\cdot(a^{(1)}_{max})^2)[$. Следовательно, число операций $\mathcal{K}_1$, требуемое для приближенного решения задачи I, оценивается величиной $\mathcal{K}_1 \sim n^2 \log t$, где $t = \max\{a^{(1)}_{max}, c^{(1)}_{max}\}$. Очевидно, что объем памяти Π_1 составляет $\sim n$ ячеек.

Трудоемкость решения задачи /I/-/2/ является нижней оценкой трудоемкости решения задачи I.Следовательно,трудоемкость приближенного решения задачи I описанным приемом отличается от нижней границы множителем $\log t$.

В работе [5] нахождение дерева $\mathcal{D}_{\lambda^0}$ для приближенного решения задачи I осуществляется перебором точек множества $\mathcal{L}_1$, в результате чего оценка трудоемкости возрастает до $\sim n^4 \log n$ операций при памяти $\sim n^2$ ячеек.

Оценим трудоемкость приближенного решения задачи 2. Для получения этого решения требуется вычислять функцию $\varphi(\lambda)$ при различных значениях параметра λ. При фиксированном λ вычисление $\tilde{\varphi}(\lambda)$, требующее упорядочения величин $\dfrac{c_j - \lambda a_j}{b_j}$ по невозрастанию, можно осуществить за $\sim n \log n$ операций [7]. Величины $\lambda^{(2)}_{max}$ и $\Delta\lambda^{(2)}$ оцениваются следующим образом: $\lambda^{(2)}_{max} \leq b^{(2)}_{max}\cdot c^{(2)}_{max}$ и $\Delta\lambda^{(2)} \geq (a^{(2)}_{max}\cdot b^{(2)}_{max})^2$, где $a^{(2)}_{max} = \max\limits_{1\leq j\leq n} a_j$

и $b^{(2)}_{max} = \max\limits_{1 \leq j \leq n} b_j$. Это означает, что функцию $\varphi(\lambda)$ достаточно вычислять лишь $\sim \log t'$ раз, где $t' = \max\{n, a^{(2)}_{max}, b^{(2)}_{max}, c^{(2)}_{max}\}$. Следовательно, трудоемкость приближенного решения задачи 2 оценивается числом операций $\mathcal{K}_2 \sim n\log^2 t'$. При этом достаточная для реализации алгоритма память $\mathcal{P}_2$ того же порядка, что и для хранения исходной информации.

Заметим, что трудоемкость получения приближенного решения задачи 2 отличается от трудоемкости упорядочения n чисел лишь множителем $\log t''$, где $t'' = \max\{a^{(2)}_{max}, b^{(2)}_{max}, c^{(2)}_{max}\}$.

В заключение автор выражает благодарность Э.Х.Гимади за постановку задачи и постоянное внимание к выполнению данной работы.

Поступила в ред.-изд.отдел
8 февраля 1978 г.

Литература

1. Гвоздев С.Е. Один подход к решению задач математического программирования. - Тез.докл. IУ Всесоюз.конф. по проблемам теор. кибернетики. Новосибирск, 1977, с.93-94.

2. Прим Р.К. Кратчайшие связывающие сети и некоторые обобщения.-Кибернетический сборник, 1961, № 2, с. 95-107.

3. Карп Р.М. Сводимость комбинаторных проблем. - Кибернетический сборник, 1975, вып.12, с. 16-38.

4. Беллман Р. Динамическое программирование.-М.: ИЛ, 1960, - 400 с.

5. Морозов С.А. О задаче нахождения кратчайшего связывающего дерева с ограничением. - В кн.: Управляемые системы. Новосибирск, 1976, вып. 15, с. 40-47.

6. Мухачева Э.А.,Рубинштейн Г.Ш. Математическое программирование. - Новосибирск: Наука, 1977. - 320 с.

7. Кронрод М.А. Оптимальный алгоритм упорядочения без рабочей памяти.-Докл.АН СССР, 1969, т.186, № 6, с.1256-1258.