УДК 512.25/26 + 519.3:330.115.

ОБ ОПТИМИЗАЦИИ КАЛЕНДАРНОЙ ДЛИТЕЛЬНОСТИ ОБРАБОТКИ ИЗДЕЛИЙ ПРИ ОДИНАКОВОЙ ПОСЛЕДОВАТЕЛЬНОСТИ ЗАПУСКА ИХ НА ВСЕХ СТАНКАХ

В.К.Тютюкин

Как известно, задача Беллмана-Джонсона состоит в нахождении плана запуска n изделий в обработку на m станках, минимизирующего суммарную календарную длительность их обработки. Эта задача была изящно (по всеобщему признанию) решена Джонсоном [1] лишь для двух и в некоторых частных случаях для трех станков. Трудность задачи быстро растет с увеличением числа станков. Уже для трех станков решение получается громоздким и не обобщается на случай четырех и более станков. Решение осложняется еще и тем, что в случае четырех и более станков нельзя считать, что последовательность запуска деталей на всех станках одинакова. Тем не менее рассмотрение задачи при этом дополнительном предположении представляет интерес, ибо соответствующее решение можно считать приближенным решением задачи Беллмана-Джонсона.

Ниже рассматривается задача Беллмана-Джонсона именно при допущении об одинаковой на всех станках последовательности запуска изделий в производство. Эту задачу назовем основной задачей в отличие от вспомогательных задач, которые будут рассмотрены в § 1 и использованы в § 2 для решения основной.

Введем обозначения:

$A = \| a_{ij} \|_{m,n}$ – матрица трудоемкостей, где a_{ij} – сумма подготовительно-заключительного и оперативного времени для j-го изделия ($j = 1,2,\ldots,n$) на i-м станке ($i=1,2,\ldots,m$);

$P = (j_1, j_2, \ldots, j_t, \ldots, j_n)$ – любой порядок запуска изделий в производство;

A_p - матрица, получающаяся из A расположением ее столбцов в порядке P ($A_p = \| a_{ijk} \|$).

T_p - календарная длительность обработки всех изделий, запускаемых в производство в порядке P .

Как известно [I], имеем

$$T(P) = \max_{1 = u_0 \leq u_1 \leq \ldots \leq u_m = n} \sum_{i=1}^{m} \sum_{k=u_{i-1}}^{u_i} a_{ijk} \qquad (*)$$

$L = (u_0, u_1, \ldots, u_m)$ назовем "допустимым путем" через матрицу с размерами $m \times n$, если выполняются условия:

I) $u_0 = 1,\ u_m = n$

(т.е. соединяется верхний левый угол (I,I) матрицы с ее нижним правым углом (m , n));

2) $u_0 \leq u_1 \leq \ldots \leq u_m$

(т.е. на каждом шаге сдвиг направлен либо на одну клетку вправо, либо на одну клетку вниз).

Длиной $\mathcal{D}(P,L)$ "допустимого пути" $L = (u_0, u_1, \ldots, u_m)$ через матрицу A_p назовем величину

$$\mathcal{D}(P,L) = \sum_{i=1}^{m} \sum_{k=u_{i-1}} a_{ijk} \,. \qquad (1)$$

Следовательно, формула ($*$) примет вид

$$T(P) = \max_{L} \mathcal{D}(P,L) \,. \qquad (I)$$

"Допустимый путь" L^* назовем "максимальным путем" через матрицу A_p , если

$$\mathcal{D}(P,L^*) = \max_{L} \mathcal{D}(P,L).$$

Тогда формулу (I) можно записать в следующем виде:

$$T(P) = \mathcal{D}(P,L^*).$$

Эта формула показывает, что календарная длительность определяется "максимальным путем" через матрицу трудоемкостей.

Задача заключается в нахождении перестановки P^* , для которой

$$T(P^*) = \min_{P} \max_{L} \mathcal{D}(P,L) \,.$$

Перестановку P^* будем называть оптимальной. Положим, далее, $T(A) = T(P^*)$.

Решение P^* нашей задачи, как правило, не будет решением задачи Беллмана-Джонсона. Однако, как предлагает Джонсон [1], в качестве практического правила можно принять для всех станков одинаковую последовательность запуска изделий (как это сделано в нашей задаче), воспользовавшись затем схемой "максимального пути" для того, чтобы сократить количество планов, из числа которых должен быть выбран оптимальный план задачи Беллмана-Джонсона.

Несмотря на упрощающее допущение в нашей задаче, она остается все же трудной: применяемый для ее решения метод "ветвей и границ" является трудоемким, а аналитическое решение еще не найдено. Для нахождения более эффективного алгоритма полезно изучить свойства оптимальной перестановки и ее зависимость от исходных данных (матрицы A). Кроме того, следует учесть, что в некоторых случаях можно избежать громоздких вычислений и применять весьма эффективные алгоритмы для нахождения оптимальной перестановки. Некоторые шаги в этом направлении излагаются ниже.

§ 1. Решение двух вспомогательных задач

З а д а ч а 1. Введем обозначения:

$A = \| a_{ij} \|$ — матрица с размерами $m \times n$,

$P = (j_1, j_2, \ldots, j_k, \ldots, j_n)$ — любая перестановка столбцов матрицы A.

A_P — матрица, получающаяся из матрицы A расположением ее столбцов в порядке P.

$L = (u_0, u_1, \ldots, u_{m-1}, u_n)$ — некоторый фиксированный "допустимый путь" через матрицу с размерами $m \times n$.

Рассматривается функция-критерий

$$F(P) = \mathcal{D}(P, L), \tag{П}$$

т.е. длина "допустимого пути" L через матрицу A_P (см. формулу (1)).

Требуется найти такую перестановку P^* столбцов матрицы A, при которой

$$F(P^*) = \min_P F(P).$$

Перестановку P^* будем называть оптимальной.

Число всевозможных перестановок столбцов матрицы A равно $n!$. Рассмотрение всех их для нахождения P^* весьма затруднительно, а при больших n и невозможно. Однако заметим, что вид функции-критерия в нашей задаче позволяет воспользоваться принципом оптимальности динамического программирования. Согласно этому принципу любая начальная последовательность оптимальной перестановки должна быть оптимальной. Отсюда вытекает возможность сравнения вариантов по значению функции-критерия не в конце построения множества всех возможных вариантов (как в схеме перебора), а на каждом шаге конструирования возможных вариантов. Такое пошаговое построение решения позволяет резко уменьшить число просматриваемых вариантов.

Пошаговое конструирование решения в нашей задаче состоит в следующем: на первом шаге выбирается тот столбец матрицы A, который должен стоять первым, на втором шаге – вторым и т.д.

Для каждой последовательности $P=(j_1, j_2, \ldots, j_k)$ определим последовательность $I(P)=(z_1, z_2, \ldots, z_k)$, получаемую из P перестановкой индексов j, и такую, что $z_\ell < z_{\ell+1}$ $(\ell=1,2,\ldots,k-1)$. Тогда две частичные последовательности P' и P'' приводят в одно и то же "состояние", если $I(P')=I(P'')$, т.е. если P' и P'' состоят из одинакового подмножества номеров столбцов матрицы A.

Исходя из принципа оптимальности динамического программирования, правило доминирования будет выглядеть следующим образом: для любых двух перестановок $P'=(j_1', j_2', \ldots, j_k')$ и $P''=(j_1'', j_2'', \ldots, j_k'')$, приводящих в одно и то же "состояние", перестановка P' "лучше" перестановки P'', если $F_k(P') \leq F_k(P'')$, где через F_k обозначена длина части "допустимого пути" L, лежащей в первых "k" столбцах матрицы с размерами $m \times n$. (Заметим, что $F_n = F^*$).

Схема решения задачи сводится к следующему.

Пусть на k-ом шаге определено множество "перспективных" начал вариантов $\bar{M}_k=\{P=(j_1, j_2, \ldots, j_k)\}$. "Развивая" каждое начало вариантов из $\bar{M}_k$ на один шаг, получим множество вариантов $\bar{M}_{k+1}=\{P=(j_1, j_2, \ldots, j_k, j_{k+1})\}$. Это множество сгруппируем в подмножества начал вариантов, приводящих в одинаковое "состояние", т.е. сгруппируем по $I(P)$. По правилу доминирования выделим в каждой группировке по одному варианту. Полученное множество и составит множество $\bar{M}_{k+1}$ "перспективных"

начал вариантов на ($k+1$)-ом шаге.
Из множества $\overline{M}_n$ остается выбрать варианты, на которых достигается минимальное значение функции-критерия $F(P)$.

На k-ом шаге конструирования множества вариантов из всех возможных перестановок вида ($j_1, j_2, \ldots, j_k$) некоторого фиксированного подмножества номеров столбцов остается только один вариант; следовательно, оценивать и "развивать" нужно всего

$$C_n^k = \frac{n(n-1)\ldots(n-k+1)}{1 \cdot 2 \cdot \ldots \cdot k}$$

вариантов. Таким образом, в описанной выше схеме последовательного анализа вариантов для нашей задачи анализируется

$$\sum_{k=1}^{n-1} C_n^k = 2^n - 1$$

вариантов, что значительно меньше числа вариантов ($n!$) в схеме перебора.

Задача 2. Дана матрица $A = \|a_{ij}\|_{m,n}$.
Рассматривается функция-критерий следующего вида:

$$H(P) = \max_{1 \leq u \leq n} \mathcal{D}(P, L_u), \qquad \text{(III)}$$

где

$P = (j_1, j_2, \ldots, j_k, \ldots, j_n)$ – любой порядок столбцов матрицы A ;
L_u – тот "допустимый путь" $L = (u_0, u_1, \ldots, u_m)$
через матрицу с размерами $m \times n$, для которого
$u_1 = u_2 = \ldots = u_{m-1} = u \quad (1 \leq u \leq n)$;
$\mathcal{D}(P, L)$ находится по формуле (1).

Таким образом, $H(P)$ является длиной того "допустимого пути" через матрицу A_P , который является наибольшим в подмножестве "допустимых путей", проходящих через весь какой-либо столбец матрицы A_P .

Требуется найти такую перестановку P^* столбцов матрицы A , при которой

$$H(P^*) = \min_P H(P) .$$

Перестановку P^* будем называть оптимальной.
Ниже будет найден алгоритм решения этой задачи, являющийся обобщением изящного алгоритма Джонсона для матрицы A размерами

$2 \times n$ (т.е. задачи обработки n изделий на двух станках) [1].

Пусть P_0 -порядок (I,2,..., n) запуска изделий в производство. Тогда будем иметь

$$H(P_0) = \max_{1 \le u \le n} \Big(\sum_{j=1}^{u} a_{1j} + \sum_{i=2}^{m-1} a_{iu} + \sum_{j=u}^{n} a_{mj} \Big)$$

или в другом виде

$$H(P_0) = \max \begin{cases} \max\limits_{1 \le u \le k-1} \Big(\sum_{j=1}^{u} a_{1j} + \sum_{i=2}^{m-1} a_{iu} + \sum_{j=u}^{n} a_{mj} \Big), \\ \sum_{j=1}^{k} a_{1j} + \sum_{i=2}^{m-1} a_{ik} + \sum_{j=k}^{n} a_{mj}, \\ \sum_{j=1}^{k+1} a_{1j} + \sum_{i=2}^{m-1} a_{i,k+1} + \sum_{j=k+1}^{n} a_{mj}, \\ \max\limits_{k+2 \le u \le n} \Big(\sum_{j=1}^{u} a_{1j} + \sum_{i=2}^{m-1} a_{iu} + \sum_{j=u}^{n} a_{mj} \Big). \end{cases} \qquad (2)$$

В перестановке P_0 поменяем местами столбцы с номерами k и $k+1$. Полученную перестановку обозначим через P_0'. Для нее будем иметь

$$H(P_0') = \begin{cases} \max\limits_{1 \le u \le k-1} \Big(\sum_{j=1}^{u} a_{1j} + \sum_{i=2}^{m-1} a_{iu} + \sum_{j=u}^{n} a_{mj} \Big), \\ \sum_{j=1}^{k-1} a_{1j} + \sum_{i=2}^{m-1} a_{i,k+1} + \sum_{j=k}^{n} a_{mj}, \\ \sum_{j=1}^{k+1} a_{1j} + \sum_{i=2}^{m} a_{ik} + \sum_{j=k+2}^{n} a_{mj}, \\ \max\limits_{k+2 \le u \le n} \Big(\sum_{j=1}^{u} a_{1j} + \sum_{i=2}^{m-1} a_{iu} + \sum_{j=u}^{n} a_{mj} \Big). \end{cases} \qquad (3)$$

Из формул (2) и (3) вытекает следующее: для того чтобы $H(P_0) \le H(P_0')$, достаточно

$$\max \Big(\sum_{j=1}^{k} a_{1j} + \sum_{i=2}^{m-1} a_{ik} + \sum_{j=k}^{n} a_{mj},\ \sum_{j=1}^{k+1} a_{1j} + \sum_{i=2}^{m-1} a_{i\,k+1} + \sum_{j=k+1}^{n} a_{mj} \Big) \le$$

$$\le \max\left(\sum_{j=1}^{k-1} a_{1j} + \sum_{i=1}^{m-1} a_{i,k+1} + \sum_{j=k}^{n} a_{mj},\quad \sum_{j=1}^{k+1} a_{1j} + \sum_{i=2}^{m} a_{ik} + \sum_{j=k+2}^{n} a_{mj}\right).$$

Вычитая из обеих частей этого неравенства величину

$$\sum_{j=1}^{k-1} a_{1j} + \sum_{i=2}^{m-1} a_{ik} + \sum_{i=2}^{m-1} a_{i,k+1} + \sum_{j=k}^{n} a_{mj},$$

получим

$$\max\left(-\sum_{i=1}^{m-1} a_{i,k+1},\ -\sum_{i=2}^{m} a_{ik}\right) \le \max\left(-\sum_{i=1}^{m-1} a_{ik},\ -\sum_{i=2}^{m} a_{i,k+1}\right).$$

Откуда

$$\min\left(\sum_{i=1}^{m-1} a_{ik},\ \sum_{i=2}^{m} a_{i,k+1}\right) \le \min\left(\sum_{i=1}^{m-1} a_{i,k+1},\ \sum_{i=2}^{m} a_{ik}\right).$$

Таким образом, для того чтобы выполнялось $H(P_0) \le H(P_0')$ достаточно неравенства

$$\min\left(A_k',\ A_{k+1}''\right) \le \min\left(A_{k+1}',\ A_k''\right), \qquad (4)$$

где положено

$$A_j' = \sum_{i=1}^{m-1} a_{ij}, \quad A_j'' = \sum_{i=2}^{m} a_{ij} \quad (j = 1, 2, \dots, n). \qquad (4')$$

Пользуясь неравенством (4) и учитывая его транзитивность, придем к перестановке P^*, для которой будет $H(P^*) \le H(P)$ при $\forall P$, ибо P^* можно получить из P последовательными перестановками и для перестановки на каждом шаге получается значение H, не превосходящее его значения на предыдущем шаге.

Следовательно, для нахождения оптимальной перестановки вычисляем числа A_j' и A_j'' по формуле (4') и для полученной матрицы

$$\hat{A} = \begin{pmatrix} A_1', A_2', \dots, A_n' \\ A_1'', A_2'', \dots, A_n'' \end{pmatrix}$$

применяем "рабочее правило" С.Джонсона для задачи обработки изделий на двух станках:

1. Находим минимальный элемент матрицы $\hat{A}$.

2. Если этот минимум выпал на первую строчку матрицы $\hat{A}$ (т.е. на A_j'), то ставим соответствующий столбец первым.

3. Если он выпал на вторую строчку (т.е. на A_j''), то ставим соответствующий столбец последним.

4. Вычеркиваем данные A_j' и A_j''.

5. Повторяем эту процедуру для уменьшенного множества $2n-2$ чисел A_j' и A_j'' и т.д.

§ 2. Достаточные условия оптимальности перестановки

Выше были рассмотрены три задачи, функции-критерии которых (формулы (I), (П), (Ш)) имеют соответственно следующие виды:

$$T(P) = \max_{L} \mathcal{D}(P, L) = \mathcal{D}(P, L^*)$$

– в основной задаче;

$$F(P) = \mathcal{D}(P, L)$$

– в задаче I;

$$H(P) = \max_{1 \le u \le n} \mathcal{D}(P, L_u)$$

– в задаче 2.

Оптимизация функции-критерия основной задачи затруднена тем, что при каждой перестановке P в соответствующей матрице A_P нужно рассматривать свой "максимальный путь" L^*.

Функции-критерии задач I и 2 являются частными случаями функции-критерия основной задачи. Действительно, при любой перестановке P в соответствующей матрице A_P рассматриваются: в основной задаче – все "допустимые пути" (кол-во их равно $C_{m+n-2}^{m-1} = C_{m+n-2}^{n-1}$); в задаче I – один и тот же, но зато произвольной формы, "допустимый путь"; в задаче 2 – несколько (а именно, n), но зато простой формы "допустимых путей".

Тем не менее решением основной задачи может быть, при некотором условии, решение более простой задачи I или 2. Соответствующие теоремы приводятся ниже.

Т е о р е м а I. Для того чтобы перестановка P^* была решением основной задачи, достаточно, чтобы выполнялись условия:

I. P^* является решением задачи I;

2. "Допустимый путь" L является "максимальным путем" через матрицу A_{P^*} .

Доказательство. В силу условия 2 теоремы имеем

$$T(P^*) = F(P^*), \tag{5}$$

а в силу условия I теоремы имеем

$$F(P^*) \leq F(P) \quad \text{при} \quad \forall P \ . \tag{6}$$

Так как длина "допустимого пути" не превосходит длины "максимального пути", то получим

$$F(P) \leq T(P) \quad \text{при} \quad \forall P \ . \tag{7}$$

Объединяя формулы (5), (6) и (7), получим

$$T(P^*) = F(P^*) \leq F(P) \leq T(P) \quad \text{при} \quad \forall P \ ,$$

откуда следует, что

$$T(P^*) \leq T(P) \quad \text{при} \quad \forall P \ .$$

Теорема доказана.

Теорема 2. Для того чтобы перестановка P^* была решением основной задачи, достаточно, чтобы выполнялись условия:

I. P^* является решением задачи 2.

2. Для "максимального пути" $L^* = (1, u_1^*, u_2^*, \ldots, u_{m-1}^*, n)$ через матрицу A_P выполняется условие: $u_1^* = u_2^* = \ldots = u_{m-1}^*$.

Доказательство этой теоремы аналогично доказательству теоремы I.

Литература

I. С.М.Джонсон. Оптимальные двух-и трехоперационные календарные планы производства с учетом подготовительно-заключительного времени. В сб. "Календарное планирование". Прогресс, М., 1966.

2. В.К.Тютюкин. Календарная длительность обработки одинаковой на всех станках последовательности изделий. Труды объединения "Ленэлектронмаш", вып.Ш, 1969 г.

Поступила в редакцию
3.X. 1969 г.