Вестник Омского университета, 2006, №3, с. 1–??.

УДК 519.8

ПРИБЛИЖЕННОЕ РЕШЕНИЕ ЗАДАЧИ УПРАВЛЕНИЯ ПОСТАВКАМИ СО МНОГИМИ ИНТЕРВАЛАМИ

А.В. Еремеев, П.М. Кузнецов

Омский государственный университет им. Ф.М. Достоевского
кафедра прикладной и вычислительной математики
644077, Омск пр.Мира 55А[1]

Получена 30 июня 2006 г.

We consider the supply management problem with a single consumer: to find a minimum cost delivery plan, given a set of admissible intervals for the shipment sizes of each supplier, lower-bounded demand, linear variable costs of delivery and fixed costs for opening each shipment. A fully polynomial time approximation scheme for this problem is proposed.

В работе предлагается вполне полиномиальная аппроксимационная схема для задачи управления поставками со многими интервалами. Требуется доставить единственному потребителю необходимое количество продукции от нескольких поставщиков так, чтобы суммарная стоимость доставки была минимальной. При этом существуют ограничения на допустимые объёмы поставок от каждого поставщика, заданные набором допустимых интервалов. Математическая модель задачи является обобщением модели [1], где рассматривался случай одиночных интервалов, и выглядит следующим образом:

$$f(x) = \sum_{i=1}^{n} c_i(x_i) \to \min, \qquad (1)$$

$$\sum_{i=1}^{n} x_i \geq A, \qquad (2)$$

$$x_i \in \{0\} \cup \bigcup_{j=1}^{I^{(i)}} [m_i^j; M_i^j], i = 1, \ldots, n. \qquad (3)$$

Переменными являются величины x_i, определяющие количество продукции, доставляемой от i-го поставщика, $i = 1, \ldots, n$. Здесь $A > 0$ – необходимое потребителю количество продукции; $I^{(i)}$ – число допустимых интервалов для объёма поставки i-го поставщика; $m_i^j \leq M_i^j$ – соответственно нижняя и верхняя границы j-го допустимого интервала i-го поставщика, $j = 1, \ldots, I^{(i)}$ ($M_i^j < m_i^{j+1}$, $j = 1, \ldots, I^{(i)} - 1$); функции $c_i(\cdot)$ определяют стоимости доставки:

$$c_i(x_i) = \begin{cases} 0, & \text{если } x_i = 0; \\ \alpha_i^j + \beta_i^j x_i, & \text{если } x_i \in [m_i^j; M_i^j], \\ & \quad j = 1, \ldots, I^{(i)}, \end{cases}$$

где α_i^j – фиксированная плата за открытие поставки объёма $x_i \in [m_i^j; M_i^j]$, β_i^j –стоимость перевозки единицы продукции при соответствующем объёме. Все исходные данные принадлежат множеству неотрицательных целых чисел Z^+. Переменные задачи являются целочисленными.

Под *ρ-приближенным алгоритмом* будем понимать алгоритм получения допустимого решения, стоимость которого не более, чем в ρ раз превышает стоимость оптимального решения (если задача разрешима).

Под *вполне полиномиальной аппроксимационной схемой* будем понимать семейство $(1+\varepsilon)$-приближенных алгоритмов при всевозможных $\varepsilon > 0$ со временной сложностью, полиномиально зависящей от длины входа задачи и от $1/\varepsilon$.

Основным результатом статьи является

Теорема 1. *Для задачи (1)–(3) существует вполне полиномиальная аппроксимационная схема трудоёмкости* $O(\log_2(c_{max}) I_{max} n^3/\varepsilon)$, *где* $c_{max} = \sum_i c_i M_i^{I^{(i)}}$.

Из леммы 1 работы [1] вытекает следующая

Лемма 1. *Разрешимая задача управления поставками со многими интервалами имеет оптимальное решение* x^* *такое, что все* x_i^**, за исключением, быть может, одного, принимают*

[1] e-mail: eremeev@iitam.omsk.net.ru, mailpk@rambler.ru
При поддержке INTAS, проект 03-51-5501.

значения из множества $\{0, m_i^j, M_i^j\}$, $j = 1, \ldots, I^{(i)}$, $i = 1, \ldots, n$.

Построение аппроксимационной схемы упрощается путем сведения задачи (1)–(3) к задаче, где все переменные принимают значения из [0,1]:

$$g(u,v) = \sum_{i=1}^{n} \sum_{j=1}^{I^{(i)}} (a_i^j u_i^j + b_i^j v_i^j) \to \min, \quad (4)$$

$$\omega(u,v) = \sum_{i=1}^{n} \sum_{j=1}^{I^{(i)}} (m_i^j u_i^j + (M_i^j - m_i^j) v_i^j) \geq A, \quad (5)$$

$$\sum_{j=1}^{I^{(i)}} u_i^j \leq 1, \ i = 1, \ldots, n, \quad (6)$$

$$v_i^j \leq u_i^j, \ j = 1, \ldots, I^{(i)}, \ i = 1, \ldots, n, \quad (7)$$

$$u_i^j \in \{0,1\}, \ (M_i^j - m_i^j) v_i^j \in Z^+, \ v_i^j \in [0,1],$$
$$j = 1, \ldots, I^{(i)}, \ i = 1, \ldots, n. \quad (8)$$

Здесь переменными величинами являются u_i^j, v_i^j. Задача (1)–(3) может быть сведена к виду (4)–(8), если воспользоваться следующими преобразованиями: $a_i^j = \alpha_i^j + \beta_i^j m_i^j$, $b_i^j = \beta_i^j (M_i^j - m_i^j)$, $x_i = \sum_{j=1}^{I^{(i)}} m_i^j u_i^j + (M_i^j - m_i^j) v_i^j$, $i = 1, \ldots, n$, $j = 1, \ldots, I^{(i)}$. Далее мы также будем использовать задачу, в которой условие (8) заменено на

$$u_i^j \in \{0,1\}, v_i^j \in \{0,1\},$$
$$j = 1, \ldots, I^{(i)}, i = 1, \ldots, n, \quad (9)$$

то есть размеры поставок принимают значения только на концах интервалов. Предположим, имеется априорная верхняя оценка U стоимости оптимального решения задачи (4)–(7), (9). Для каждого $k = 1, \ldots, n$ и $c = 0, \ldots, U$ определим множество $S(k,c)$, которое для каждой пары (k,c) содержит значения переменных u_i^j, v_i^j, обеспечивающие при фиксированной стоимости c наибольший объём поставок от первых k поставщиков. Теперь для пары (k,c), если $S(k,c) \neq \emptyset$, будем полагать $\varphi(k,c) = \omega(u,v)$, где $(u,v) \in S(k,c)$. Если же $S(k,c) = \emptyset$, то $\varphi(k,c) = -\infty$. Под $\varphi(k,c)$ понимается максимальное количество продукции, которое могут обеспечить k поставщиков при заданной стоимости поставок c. Тогда начальное множество состояний алгоритма динамического программирования (ДП) определяется следующим образом:

$$\varphi(1,c) = \begin{cases} 0, & \text{если } c = 0; \\ m_1^j, & \text{если } c = a_1^j \text{ при некотором } j; \\ M_1^j, & \text{если } c = a_1^j + b_1^j \\ & \qquad \text{при некотором } j; \\ -\infty, & \text{иначе.} \end{cases}$$

Переход в следующее состояние с $k = 2, \ldots, n$ осуществляется по рекурсивному соотношению:

$$\varphi(k,c) = \max\{V + \varphi(k-1, c - c_k(V))\},$$

где V – объём поставки, принимающий значения из множества $\{0, m_k^j, M_k^j\}$, при этом соответствующая стоимость поставки $c_k(V) \leq c$.

Таким образом, если g' является оптимальным значением целевой функции для задачи (4)–(7),(9) и $g' \leq U$, то $g' = \min\{c | \varphi(n,c) \geq A\}$ и множество $S(n, g')$ содержит одно или несколько оптимальных решений задачи. Как следует из леммы 1, решение задачи (4)–(8) мы можем найти, осуществляя перебор по $i = 1, \ldots, n$ и по каждой координате $v_i^j, j = 1, \ldots, I^{(i)}$ по всем её значениям $v_i^j \in [0,1]$, $(M_i^j - m_i^j) v_i^j \in Z^+$, полагая при этом $u_i^j = 1$. Значения прочих переменных вычисляются с помощью алгоритма ДП.

Предположим, имеется верхняя оценка UB на оптимальное значение целевой функции g^* задачи (4)–(8). Оптимум данной задачи может быть найден следующим точным алгоритмом.

Алгоритм A

Пусть $\tilde{g} = +\infty$.

Для i **от** 1 **до** n выполнять шаги 1–3:

1. Применить алгоритм ДП со значением $U = UB$ к задаче (4)–(7), (9), в которой m_i^j и $M_i^j, j = 1, \ldots, I^{(i)}$ заменены на 0.

2. Выбрать представителей $(u^{(0)}, v^{(0)}), \ldots, (u^{(UB)}, v^{(UB)})$ из каждого множества $S(n, 0), \ldots, S(n, UB)$.

3. **Для** c **от** 0 **до** UB выполнять шаги 3.1, 3.2:

3.1. **Если** $A - \varphi_i(n,c) \leq 0$ **и** $c \leq \tilde{g}$, **то** обновить $\tilde{g} := c$, $\tilde{u} := u^{(c)}$, $\tilde{v} := v^{(c)}$.

3.2. **Для** t **от** 1 **до** $I^{(i)}$ выполнять шаги 3.2.1–3.2.3:

3.2.1. **Если** $A - \varphi_i(n,c) = m_i^t$ **и** $c + a_i^t \leq \tilde{g}$, **то** обновить значение $\tilde{g} := c + a_i^t$ и положить $\tilde{u}_k := u_k{}^{(c)}$, $\tilde{v}_k := v_k{}^{(c)}, k \neq i$; $u_i^s := 0$, $v_i^s := 0, s \neq t, \tilde{u}_i^t := 1$, $\tilde{v}_i^t := 0$.

3.2.2. **Если** $m_i^t < A - \varphi_i(n,c) \leq M_i^t$ **и** $c + a_i^t + b_i^t \frac{A - m_i^t - \varphi_i(n,c)}{M_i^t - m_i^t} \leq \tilde{g}$, **то** обновить значение $\tilde{g} := c + a_i^t + b_i^t \frac{A - m_i^t - \varphi_i(n,c)}{M_i^t - m_i^t}$ и положить $\tilde{u}_k := u_k{}^{(c)}$, $\tilde{v}_k := v_k{}^{(c)}, k \neq i$; $u_i^s := 0, v_i^s := 0, s \neq t, \tilde{u}_i^t := 1, \tilde{v}_i^t := \frac{A - m_i^t - \varphi_i(n,c)}{M_i^t - m_i^t}$.

3.2.3. **Если** $M_i^t < A - \varphi_i(n,c) \le m_i^{t+1}$ **и** $c + a_i^{t+1} \le \tilde{g}$, **то** обновить значение $\tilde{g} := c + a_i^{t+1}$ и положить
$\tilde{u}_k := {u_k}^{(c)},\ \tilde{v}_k := {v_k}^{(c)},\ k \ne i;$
$u_i^s := 0, v_i^s := 0, s \ne t+1,$
$\tilde{u}_i^{t+1} := 1, \tilde{v}_i^{t+1} := 0.$

Если $\tilde{g} < +\infty$, то решение $(\tilde{u}, \tilde{v})$ оптимально, иначе задача неразрешима.

Утверждение 1. *Для решения задачи (4)–(8) алгоритмом* **A** *требуется* $O(n^2 I_{max} UB)$ *операций, где* $I_{max} = \max_{i=1}^n I^{(i)}$.

Данное утверждение следует из леммы 1 и того свойства алгоритма ДП, что от всех поставщиков, за исключением поставщика i, обеспечивается наибольшее количество доставляемой продукции при каждой заданной суммарной стоимости c.

Доказательство теоремы 1. Применим метод округления исходных данных, впервые использованный в [2].

Поскольку априори верхняя оценка оптимальной стоимости решения f^* не известна, то будем перебирать последовательность пробных оценок. Обозначим через UB' и LB' верхнюю и нижнюю пробные оценки. Последовательно будем рассматривать $UB' = 2^\ell$, $LB' = UB'/2$, $\ell = 0, \dots, \lceil \log_2 c_{max} \rceil$.

В том случае, когда неравенства $UB' > f^*$ и $LB' < f'$ выполнены для некоторого значения ℓ, будут применимы следующие рассуждения.

Пусть $\varepsilon > 0$ задано. Положим

$$\delta = \max\left\{\frac{\varepsilon LB'}{2n}, 1\right\}$$

и применим алгоритм **A** к задаче (4)–(8) с параметрами:

$$a_i^j = \left\lceil \frac{\alpha_i^j + \beta_i^j m_i^j}{\delta} \right\rceil,\ b_i^j = \left\lceil \frac{\beta_i^j (M_i^j - m_i^j)}{\delta} \right\rceil,$$

$$UB = \frac{UB'}{\delta} + 2n. \qquad (10)$$

Верхняя оценка UB является корректной, потому что стоимость оптимального решения для задачи (4)–(8) с коэффициентами a_i^j и b_i^j, определёнными в (10), не превосходит $\frac{f^*}{\delta} + 2n$. Заметим, что если $\frac{\varepsilon LB'}{2n} > 1$, то справедлива следующая верхняя оценка для задачи с округлёнными коэффициентами:

$$UB = \frac{UB'}{\delta} + 2n = \frac{2nUB'}{\varepsilon LB'} + 2n = \frac{4n}{\varepsilon} + 2n.$$

Если $\frac{\varepsilon LB'}{2n} \le 1$, то верхняя оценка для задачи с округлёнными коэффициентами по-прежнему корректна. Таким образом, по утверждению 1 трудоёмкость данного алгоритма составляет $O(\log_2(c_{max}) I_{max} n^3/\varepsilon)$ операций. Проверим, что получаемое решение

$$\tilde{x_i} = \sum_j \left(m_i^j \tilde{u}_i^j + (M_i^j - m_i^j)\tilde{v}_i^j \right),$$

$j = 1, \dots, I^{(i)}$, $i = 1, \dots, n$, является $(1+\varepsilon)$-приближенным решением задачи (1)–(3). Если $\delta = 1$, то $\tilde{x}$ является оптимумом. Если же $\delta > 1$, обозначим стоимость решения, соответствующего x в задаче (4)–(8) через $g(x) = g(u(x), v(x))$, где

$$u(x)_i^j = \begin{cases} 1, & \text{если } x_i \in [m_i^j, M_i^j]; \\ 0, & \text{иначе,} \end{cases}$$

$$v(x)_i^j = \begin{cases} \frac{x_i - m_i^j}{M_i^j - m_i^j} u(x)_i^j, & \text{если } M_i^j \ne m_i^j; \\ 0, & \text{иначе.} \end{cases}$$

Для оптимума x^* задачи (1)–(3) имеем:

$$\frac{f(x^*)}{\delta} = \sum_{i=1}^n \sum_{j=1}^{I^{(i)}} \frac{c_i m_i^j}{\delta} u(x^*)_i^j +$$

$$+ \sum_{i=1}^n \sum_{j=1}^{I^{(i)}} \frac{c_i (M_i^j - m_i^j)}{\delta} v(x^*)_i^j \ge$$

$$\sum_{i=1}^n \sum_{j=1}^{I^{(i)}} \left\lceil \frac{c_i m_i^j}{\delta} \right\rceil u(x^*)_i^j +$$

$$+ \sum_{i=1}^n \sum_{j=1}^{I^{(i)}} \left\lceil \frac{c_i (M_i^j - m_i^j)}{\delta} \right\rceil v(x^*)_i^j - 2n = g(x^*) - 2n.$$

Так как $(\tilde{u}, \tilde{v})$ является оптимальным решением задачи (4)–(8), то $\delta g(\tilde{x}) \le \delta g(x^*) \le f^* + 2n\delta$, и

$$\frac{|f(\tilde{x}) - f^*|}{f^*} \le \frac{\delta g(\tilde{x}) - f^*}{f^*} \le \frac{2n\delta}{LB^*} = \varepsilon.$$

Таким образом, при $\ell = 0, \dots, \lceil \log_2 c_{max} \rceil$ будет получено не более чем $1 + \lceil \log_2 c_{max} \rceil$ допустимых решений. Наименьшее по целевой функции среди этих решений будет $(1+\varepsilon)$-приближенным. Теорема доказана.

1. *Еремеев А.В., Романова А.А., Servax В.В., Чаухан С.С.* Приближенное решение одной задачи управления поставками // Дискретн. анализ и исслед. операций. 2006. Сер. 2, Т. 13, N. 1. С. 27–39.
2. *Ibarra O. H., Kim C. E.* Fast approximation algorithms for the knapsack and sum of subset problems // J. of the ACM. 1975. V. 22, N. 4. P. 463–468.