ISSN 2074-1863 Уфимский математический журнал. Том 3. № 4 (2011). С. 57-63.

УДК 519.1+519.8

КОМБИНАТОРНАЯ СЛОЖНОСТЬ ОДНОГО КЛАССА ЗАДАЧИ ЛИНЕЙНОГО РАСКРОЯ

В.М. КАРТАК, В.В. КАРТАК

Аннотация. В статье рассмотрена классическая задача одномерной упаковки (1dCSP), которая является NP-трудной. Приведен один из возможных комбинаторных алгоритмов ее решения, основанный на методе ветвей и границ. Оценена сложность представленного алгоритма для одного класса задач, который назван плотным. Выявлены примеры, наиболее трудные для решения комбинаторными алгоритмами. Этот результат согласуется с экспериментальными данными и может быть использован для генерации трудных тестовых задач, а также для прогнозирования времени работы алгоритма.

Ключевые слова: линейный раскрой, метод ветвей и границ, целочисленное программирование, сложность алгоритма.

1. Введение

Классическая задача одномерной упаковки *One-Dimensional Cutting Stock Problem (1dCSP)* состоит в следующем: дано множество неотрицательных чисел $l_i \in \mathbb{R}_+$, $i \in I = \{1, \ldots, m\}$ и некоторое положительное число $L > 0$. Требуется найти минимальное натуральное число n такое, что I разбивается на n не пересекающихся подмножеств $I = \cup_{k=1}^{n} I_k$ и $\sum_{i \in I_k} l_i \leqslant L$. Обозначим эту задачу как $E = (L, m, (l_1, l_2, \ \ldots \ l_m))$. Без потери общности предположим, что l_i упорядочены по мере их убывания $l_1 \geq l_2 \geq \ldots \geq l_m$.

В литературе эта задача известна как задача размещения в контейнеры, задача линейного раскроя. К этой постановке также сводится ряд задач календарного планирования и составления расписаний. Данная задача относится к классу NP-трудных задач и, следовательно, не может быть решена псевдополиномиальным алгоритмом (см. [1]).

И.В. Романовский в своих работах интерпретировал задачу одномерной упаковки как проблему комбинаторной оптимизации. Им была предложена общая идея переборного метода для ее решения и предложена его конкретизация в виде метода "ветвей и границ"(Method Branch and Bound, MBB) [2], который был реализован С.В. Кацевым, см. [3]. Позднее S.Martello и P.Toth в работе [4] и Э.А. Мухачева и В.М. Картак в работе [5] предложили улучшенные версии MBB, добавив в них дополнительные ограничения. В 1997 году D. Schwerin и G. Wascher провели классификацию входных данных для задачи линейного раскроя, см. [6]. В своей последующей работе [7] они, а также Э.А. Мухачева и В.М. Картак в работе [5], выделили наиболее трудные классы для получения оптимума (трудность класса определялось числом примеров, для которых удалось получить оптимум за отведенное время). При этом наиболее трудными примерами оказались „триплеты“, в которых $L/4 < l_i \leqslant L$.

V.M.Kartak, V.V.Kartak, Combinatorial complexity of a certain 1-dimensional Cutting Stock Problem.

Работа поддержана РФФИ 10-07-91330-ННИО-а, 09-01-00046-а и НШ-65497.2010.9.

Поступила 25 июня 2011 г.

Данная статья посвящена рассмотрению вычислительной сложности комбинаторных алгоритмов в случае плотной задачи.

2. Схема комбинаторного алгоритма

Каждому подмножеству разбиения I_k сопоставим m-мерный бинарный вектор $\mathbf{a}^k = (a_1^k, \ldots, a_m^k)^T$, где $a_i^k = 1$ если $i \in I_k$, иначе $a_i = 0$. Тогда любое решение задачи, состоящее из n подмножеств, может быть представлено в виде бинарной матрицы разбиения $\mathcal{A} = (a^1, a^2, \ldots, a^n)$ размеров $m \times n$. Разбиение, при котором n достигает минимума, называется *оптимальным.*

Для получения оптимального разбиения комбинаторные алгоритмы типа ветвей и границ вынуждены в худшем случае последовательно просматривать все возможные варианты разбиения [2], [4], [5]. Для избежания повторов вводится *лексикографическое упорядочивание матриц разбиения.*

Вектор a^j имеет более высокий *приоритет* по сравнению с a^k, если верно $\sum_{i=m}^{1} a_i^j \cdot 2^i > \sum_{i=m}^{1} a_i^k \cdot 2^i$. Потребуем, чтобы столбцы в матрице $\mathcal{A}$ были упорядочены по невозрастанию приоритетов. Из двух матриц приоритетной считается та, которая содержит более приоритетный вектор. Таким образом, все матрицы можно лексикографически упорядочить по мере невозрастания их приоритетов $\mathcal{A}_1 \geq \mathcal{A}_2 \geq \ldots \geq \mathcal{A}_K$. Теперь общая схема перебора будет выглядеть следующим образом:

Пусть $\Lambda(\mathcal{A}_i)$ — число столбцов в матрице $\mathcal{A}_i$.

Шаг 1. Подготовка к решению:
- вычисляем L_d — нижнюю границу;
- строим $\mathcal{A}_1$ с помощью алгоритма *первый подходящий* [1]. $L_u = \Lambda(\mathcal{A}_1)$ – верхняя граница;
- $\mathcal{A}_{best} = \mathcal{A}_1$ — лучший план раскроя.

Шаг 2. Если $L_d = L_u$, то переход к **Шагу 4.**

Шаг 3. Последовательно перебираем $\mathcal{A}_i$ с использованием лексикографического упорядочивания и использования отсечений ([4], [5]). Если $\Lambda(\mathcal{A}_i) < L_u$, то $\mathcal{A}_{best} = \mathcal{A}_i$, $L_u = \Lambda(\mathcal{A}_i)$, переход к **Шагу 2**.

Шаг 4. Найдено оптимальное решение $\mathcal{A}_{best}$.

Оценим вычислительную сложность представленного алгоритма. Очевидно, что максимальное число итераций алгоритм произведет в том случае, когда не удается достигнуть нижней границы, так как в этом случае алгоритм вынужден генерировать все возможные матрицы разбиения.

3. Оценка сложности алгоритма в плотном случае

Вектор a назовем *плотным*, если $\sum_{i=1}^{m} a_j l_j < \min_{k \in I/\{t:a_t=1\}} l_k$. Матрицу назовем *плотной*, если она состоит только из плотных векторов.

Пусть n – оптимальное значение задачи E. Будем говорить, что задача E *плотная*, если любая матрица разбиения $\mathcal{A}$, соответствующая оптимальному решению, является плотной.

Из представленной схемы видно, что для доказательства оптимальности значения n алгоритм вынужден построить все допустимые матрицы $\mathcal{A}$ размеров $m \times n$ и показать, что допустимой матрицы с числом столбцов $n - 1$ не существует. Оценим максимальное возможное число таких матриц.

Для этой цели возьмем некоторую последовательность неповторяющихся номеров $Q = (q_1, q_2, \ldots q_m)$, где $q_i \in I$ (всего возможно $m!$ различных таких последовательностей). Каждой такой последовательности однозначно сопоставим матрицу $\mathcal{A}$ по следующему правилу.

- Пусть k_1 такой номер, что $\sum_{i=1}^{k_1} l_{l_{q_i}} \leqslant L$ и $\sum_{i=1}^{k_1+1} l_{l_{q_i}} > L$. Тогда вектор a^1 строим по правилу $a^1_{q_i} = 1, i = \overline{1, k_1}$, остальные элементы нули.
- Номер k_2 такой, что $\sum_{i=k_1+1}^{k_1+k_2} l_{l_{q_i}} \leqslant L$ и $\sum_{i=k_1+k_2+1}^{k_1+k_2+1} l_{l_{q_i}} > L$. Вектор a^2 строится по правилу $a^2_{q_i} = 1, i = \overline{k_1+1, k_1+k_2}$, остальные элементы нули, и т.д.

После построения матрицы $\mathcal{A}$ ее столбцы лексикографически упорядочим.

Заметим, что одна и та же матрица $\mathcal{A}$ может быть получена из нескольких различных последовательностей. Пусть $K(\mathcal{A})$ – число последовательностей, из которых получается матрица $\mathcal{A}$. Очевидно, что если $\mathcal{A}$ плотная матрица, состоящая из n столбцов, то $K(\mathcal{A}) = k_1!k_2!..k_n!n!$.

Лемма 1. *Справедлива оценка* $k_1!k_2!..k_n! \geq (k!)^{n-r} \cdot ((k+1)!)^r$, *здесь* $k = [m/n]$ *— целая часть от деления* m *на* n, r *— остаток от деления* m *на* n.

$\triangleleft$ Найдем минимум выражения $k_1!k_2!..k_n!$, используя граф Феррера ([8], стр. 21). Составим таблицу из m строк и n столбцов. В первом столбце заполним k_1 строку, считая снизу, во втором k_2 строки, и т.д., в последнем n-ом столбце заполним k_n строк, и присвоим заполненным ячейкам веса, как показано на рисунке.

		k_3		
k_1		k_3-1		
k_1-1		k_3-2	$\dots$	k_n
k_1-2	k_2	k_3-3	$\dots$	k_n-1
k_1-3	k_2-1	k_3-4	$\dots$	k_n-2
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
1	1	1	$\dots$	1

Общее число всех заполненных клеток равно $k_1 + k_2 + \dots + k_n = m$. Мы ищем минимум произведения $k_1!k_2!k_3!\dots k_n!$, которое содержит ровно n множителей.

На графе разрешается выполнять следующую операцию: „перекладывать“ верхнюю занятую ячейку из одного столбца в другой столбец на верхнюю позицию. Например, переложим верхнюю ячейку из третьего столбца во второй:

k_1		k_3-1		
k_1-1	k_2+1	k_3-2	$\dots$	k_n
k_1-2	k_2	k_3-3	$\dots$	k_n-1
k_1-3	k_2-1	k_3-4	$\dots$	k_n-2
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
1	1	1	$\dots$	1

Возможны только два случая

$$k_1!(k_2+1)!(k_3-1)!\dots k_n! < k_1!k_2!k_3!\dots k_n! \quad \text{если} \quad k_2+1 < k_3,$$
$$k_1!(k_2+1)!(k_3-1)!\dots k_n! = k_1!k_2!k_3!\dots k_n! \quad \text{если} \quad k_2+1 = k_3.$$

Как видим, при „перекладывании“ ячейки вниз искомое произведение не увеличивается. Оно достигает своего минимума в той ситуации, когда при любом перекладывании ячейки на ряд ниже произведение остается тем же самым. Тогда операция „перекладывания“ сводится по сути к перестановке столбцов.

Итак, при фиксированных m и n произведение $k_1!k_2!k_3!\dots k_n!$ достигает своего минимума по переменным k_1, k_2, $\dots$, k_n в случае:

$$k_1!k_2!k_3!\dots k_n! = (k!)^{n-r}\cdot((k+1)!)^r\cdot n! = (k!)^n\cdot(k+1)^r\cdot n!. \quad \triangleright$$

Пусть E – плотная задача и $S(m,n)$ – число возможных различных плотных матриц $\mathcal{A}$ размеров $m\times n$. Тогда имеет место следующая оценка

$$\sum_{i=1}^{S(m,n)} K(\mathcal{A}) \leqslant m!.$$

Из Леммы 1 следует, что

$$\sum_{i=1}^{S(m,n)} K(\mathcal{A}) = \sum_{i=1}^{S(m,n)} k_1^i!k_2^i!..k_n^i!n! \leqslant m! \quad \Rightarrow \quad S(m,n)\cdot(k!)^{n-r}\cdot((k+1)!)^r n! \leqslant m!$$

или

$$S(m,n) \leqslant \frac{m!}{(k!)^{n-r}\cdot((k+1)!)^r\cdot n!}.$$

Заметим, что существуют такие задачи E, при которых достигается равенство. Например: $l_1 = l_2 = \dots = l_m$ и $m = n\lfloor L/l_1\rfloor$.

Ответим теперь на вопрос: **при каком соотношении между m и n функция $S(m,n)$ достигает максимума?**(m фиксированное число.) Для этого откажемся от целочисленности и заменим все факториалы на Гамма-функцию по правилу: $\Gamma(n+1) = n!$

Лемма 2. *Функция $S(m,n)$ мажорируется функцией $F(m,n)$*

$$S(m,n) \le F(m,n) = \frac{\Gamma(m+1)}{\Gamma^n(\frac{m}{n}+1)\Gamma(n+1)}.$$

$\triangleleft$ Докажем, что верно неравенство

$$\frac{m!}{(k!)^n\cdot(k+1)^r\cdot n!} \le \frac{\Gamma(m+1)}{\Gamma^n(\frac{m}{n}+1)\Gamma(n+1)}.$$

Задача равносильна доказательству

$$(k!)^n\cdot(k+1)^r \ge \Gamma^n\left(\frac{m}{n}+1\right) = \Gamma^n\left(\frac{kn+r}{n}+1\right) = \Gamma^n\left(k+\frac{r}{n}+1\right).$$

Обозначим $x = k+1$, $s = r/n$, причем $0 \le s < 1$, получим

$$\Gamma(x)\cdot(x)^s \ge \Gamma(x+s). \tag{1}$$

Неравенство (1) верно в силу известного неравенства (см. [9], [10], [11], [12])

$$x^{1-s} < \frac{\Gamma(x+1)}{\Gamma(x+s)}, \qquad 0<s<1,$$

так как после преобразования оно имеет следующий вид:

$$\frac{x}{x^s} < \frac{x\Gamma(x)}{\Gamma(x+s)}, \qquad 0<s<1.$$

В случае $s=1$ неравенство (1) превращается в тождество. $\triangleright$

Сформулируем следующую вспомогательную лемму.

Лемма 3. *При $z \ge 3$ дигамма-функция $\Psi(z) = \Gamma'(z)/\Gamma(z)$ оценивается:*

$$\ln z - C(z) \le \Psi(z) \le \ln z + C(z), \quad \text{где } C(z) \text{ — конечное выражение.}$$

$\triangleleft$ Распишем подробнее выражение

$$\Psi(z+1) = \Psi(z) + \frac{1}{z} = \Psi(z-1) + \frac{1}{z-1} + \frac{1}{z} = \cdots =$$

Представим переменную z в виде $z = [z] + \{z\} = n + \alpha$, $0 \le \alpha < 1$, тогда предыдущее равенство продолжается:

$$= \Psi(\alpha+2) + \frac{1}{\alpha+2} + \frac{1}{\alpha+3} + \cdots + \frac{1}{z}.$$

Справедлива оценка:

$$\int_{\alpha+k}^{\alpha+k+1} \frac{dt}{t} \le \frac{1}{\alpha+k} \le \int_{\alpha+k-1}^{\alpha+k} \frac{dt}{t}.$$

Просуммируем все неравенства, получим:

$$\int_{\alpha+2}^{\alpha+n+1} \frac{dt}{t} \le \sum_{k=2}^{n} \frac{1}{\alpha+k} \le \int_{\alpha+1}^{\alpha+n} \frac{dt}{t}.$$

Левая и правая части неравенства интегрируются:

$$\ln(z+1) - \ln(\alpha+2) \le \sum_{k=2}^{n} \frac{1}{\alpha+k} \le \ln z - \ln(\alpha+1).$$

Прибавим ко всем частям неравенства $\Psi(\alpha+2)$, получим:

$$\ln(z+1) + \Psi(\alpha+2) - \ln(\alpha+2) \le \Psi(z+1) \le \ln z + \Psi(\alpha+2) - \ln(\alpha+1).$$

Таким образом, при $z \ge 3$ справедлива оценка

$$\ln z - C(z) \le \Psi(z) \le \ln z + C(z),$$

здесь $C(z)$ — конечное выражение. $\triangleright$

Введем функцию, имеющую смысл числа заготовок в карте раскроя

$$\Phi(m) = \left\{ \frac{m}{n} \ : \ F(m,n) \to max \right\}.$$

Теорема 1. *При $m \to \infty$ значение $\Phi(m) \simeq \beta \cdot \ln m$, где β некоторая константа.*

$\triangleleft$ При фиксированном m найдем минимум знаменателя:

$$f(t) = \left(\Gamma\left(t+1\right)\right)^{\frac{m}{t}} \Gamma\left(\frac{m}{t}+1\right)$$

Для этого продифференцируем его по t:

$$\left(\Gamma\left(t+1\right)\right)^{\frac{m}{t}} m\Gamma\left(\frac{m+t}{t}\right)\left(-\ln\left(\Gamma\left(t+1\right)\right) + \Psi\left(t+1\right)t - \Psi\left(\frac{m}{t}+1\right)\right)t^{-2}$$

Рассмотрим уравнение

$$\ln\Gamma(t+1) - t\Psi(t+1) + \Psi\left(\frac{m}{t}+1\right) = 0. \tag{2}$$

Обозначим $A(t) = \ln\Gamma(t+1) - t\Psi(t+1)$ и воспользуемся известными равенствами $\Gamma(z+1) = z\Gamma(z)$ и $\Psi(z+1) = \Psi(z) + 1/z$ для подсчета $A(t+1)$:

$$A(t+1) = \ln\Gamma(t+2) - (t+1)\Psi(t+2) = \ln((t+1)\Gamma(t+1)) - \left(\Psi(t+1) + \frac{1}{t+1}\right)(t+1) =$$

$$= \ln(t+1) + \ln\Gamma(t+1) - t\Psi(t+1) - \Psi(t+1) - 1 = A(t) + \ln(t+1) - \Psi(t+1) - 1.$$

Таким образом,

$$A(t+1) - A(t) = \ln(t+1) - \Psi(t+1) - 1. \tag{3}$$

Правая часть (3) конечна по Лемме 3, поэтому можно оценить скорость роста $A(t)$: $A(t) \simeq \alpha t$, $\alpha = const$. Пусть некоторое t_m – решение уравнения (2), тогда, подставляя в него оценку для $A(t)$, получим $\Psi\left(\frac{m}{t_m}+1\right) \simeq \alpha t_m$.

Воспользуемся еще раз Леммой 3: $\ln(m/t_m) \simeq \alpha t_m$, отсюда $t_m \simeq \beta \ln m$, $\beta = const$. $\triangleright$

4. Выводы

Для целых значений $m \in \{1,\ldots,1000\}$ график функции $\Phi(m)$ был построен поточечно, см. Рис.1. По графику определено значение константы $\beta \approx 0.98$.

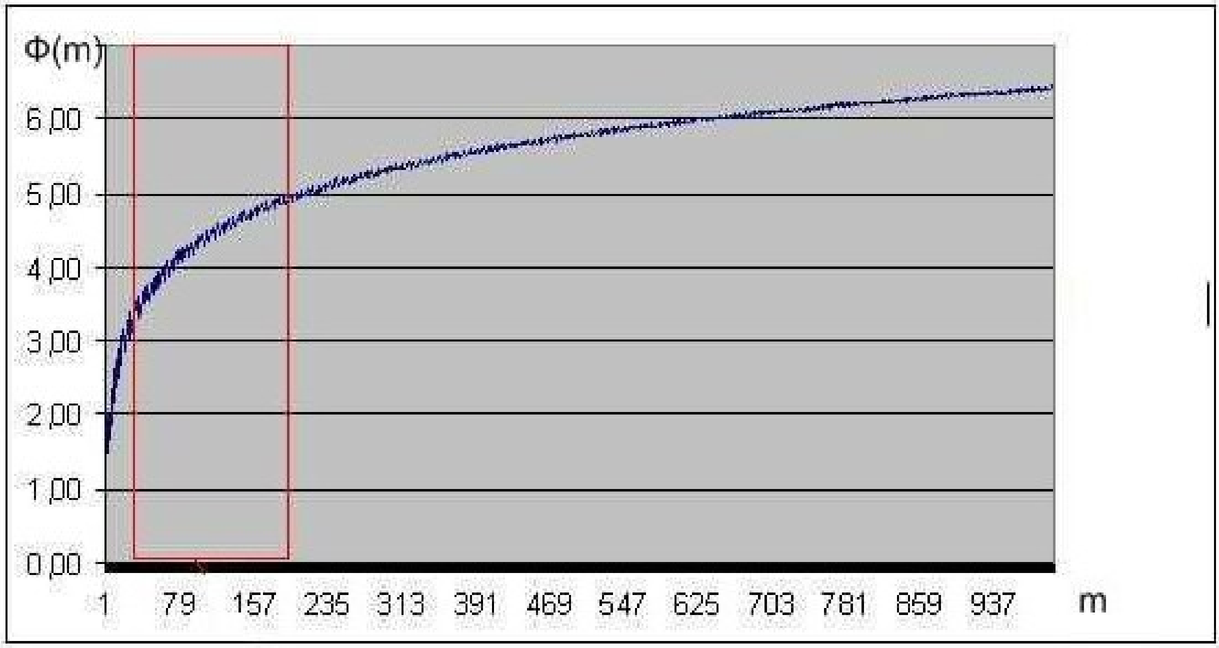

Рис. 1. График $\Phi(m)$, полученный численно

Полученный результат согласуется с данными Schwerin P., Wascher G., которые экспериментально выделили трудные классы для решения последовательными алгоритмами при $m \in [40..200]$, см. [7]. На Рис.1 их результат выделен прямоугольником.

Результаты Теоремы 1 можно также использовать для формирования наиболее трудоемких тестовых задач, в которых число допустимых решений будет максимально.

Авторы выражают искреннюю благодарность профессору Юлмухаметову Р.С. за ценные замечания и доказательство Теоремы 1.

СПИСОК ЛИТЕРАТУРЫ

1. Гэри М.П., Джонсон Д.С. *Вычислительные машины и трудноразрешимые задачи.* М.: Мир. 1982.
2. Романовский И.В. *Алгоритмы решения экстремальных задач.* М.: Наука. 1977. 88 с.
3. Кацев С.В. *Об одном классе дискретных минимаксных задач* // Кибернетика. 1979. № 5. С. 139–141.
4. S. Martello and P.Toth. *Lower Bounds and Reduction procedures for the Bin Packing Problem.* Discrete Applied Mathematics, 1990.
5. Мухачева Э.А.,Картак В.М. *Модифицированный метод ветвей и границ: алгоритм и численный эксперимент для задачи одномерного раскроя* // Информационные технологии. 2000. № 9. С. 15–21.

6. P. Schwerin, G. Wašcher *The Bin-Packing Problem: A problem Generator and Some Numerical Experiments with FFD Packing and MTP* // International Transactions in Operational Research. 1997. V. 4. №5/6. Pp. 377–389.
7. P. Schwerin, G. Wašcher *A new lower bound for the Bin-Packing Problem and its integration into MTP* // Betreils-wirtschaftliche Diskussionsbeitrage. Beitrag 1998. № 26 Martin-Luher Universitat Hall-Wittenberg. 23 p.
8. Г.Эндрюс *Теория разбиений.* М.: Наука. 1982.
9. Andrea Laforgia *Further Inequalities for the Gamma Function* // Mathematics of Computation. 1984. Vol. 42. № 166. Pp. 597–600.
10. W. Gautschi *A harmonic mean inequality for the gamma function* // SIAM J. Math. Anal. 1974. V. 5. Pp. 278–281.
11. W. Gautschi *Some mean value inequalities for the gamma function* // SIAM J. Math. Anal. 1974. V. 5. Pp. 282–292.
12. Digital Library of Mathematical Functions. http://dlmf.nist.gov/5.6.E4.

Картак Вадим Михайлович,
Уфимский государственный авиационный технический университет,
ул. К. Маркса, 12,
450000, г. Уфа, Россия
E-mail: kvmail@mail.ru

Картак Вера Валерьевна,
Башкирский государственный университет,
ул. З. Валиди, 32,
450074, г. Уфа, Россия
E-mail: kvera@mail.ru