

Анализ одного алгоритма упаковки прямоугольников, связанного с построением расписаний для кластеров¹

А. И. Поспелов

Аннотация. Рассматривается задача минимизации высоты заполнения при упаковке прямоугольников в несколько полубесконечных полос, фиксированной ширины каждая. Предложен эвристический алгоритм упаковки и доказана гарантированная оценка точности этого алгоритма. Описанный алгоритм может использоваться для составления расписания для набора кластеров.

1. Введение

Двумерные задачи упаковки возникают в различных ситуациях. В связи с этим рассматриваются различные модели. Как одна из возможных постановок может быть предложена, например, рассмотренная в [8] задача поиска прямоугольника минимальной площади, в котором без наложений можно разместить заданный набор прямоугольников. Другая интенсивно изучаемая модель — это упаковка заданного списка прямоугольников в минимальное число прямоугольников заданного размера [5]. В данной работе будет рассмотрен третий тип моделей, который заключается в поиске оптимальной в некотором смысле упаковки прямоугольников в полубесконечную полосу.

Задача упаковки прямоугольников в полосу состоит в следующем. Пусть заданы полубесконечная полоса некоторой ширины и набор прямоугольников. Причём каждая сторона каждого прямоугольника параллельна либо основанию, либо вертикальной границы полосы. Требуется упаковать все прямоугольники без наложений и без вращений в полосу, минимизируя высоту заполнения, т.е. максимальную высоту, измеренную от основания полосы, пространства занятого прямоугольниками. Эта задача может возникать в различных ситуациях и интенсивно исследовалась ранее [3]. Некоторые её приложения связаны с задачей составления расписаний для многопроцессорных систем. Прямоугольник соответствует описанию вычислительной задачи при этом, ширина определяет необходимые ресурсы (процессоры/память), а высота — время выполнения

¹Работа выполнена при поддержке РФФИ, проект 02-01-00713.

задачи. Ширина полосы соответствует общему числу процессоров на вычислительном устройстве.

В данной работе мы будем рассматривать обобщение указанной задачи. А именно, мы рассмотрим задачу упаковки прямоугольников из некоторого списка L в несколько полубесконечных полос, заданной ширины каждая. В качестве критерия эффективности упаковки, который минимизируется, рассматривается максимальная среди всех полос высота заполнения (*make-span*).

В [10] показано, что даже более частная задача, заключающаяся в упаковке в полосу прямоугольников равной ширины, NP-полная, поэтому для решения подобных задач обычно рассматривают эвристические алгоритмы, которые позволяют найти приближённое решение. Одна из таких эвристик основана на последовательном выборе прямоугольников из списка, и упаковке их на минимально возможную высоту в самую левую из возможных позиций (*bottom-left*). В общем случае упаковка, получаемая данной эвристикой, может быть в произвольное число раз хуже оптимальной. С другой стороны, для любого упорядоченного списка может быть показано, что гарантированная оценка точности равна трём, т.е. для любого упорядоченного по убыванию ширины списка прямоугольников высота заполнения, полученная с помощью эвристики, превосходит не более чем в три раза высоту оптимального заполнения [3].

Статья организована следующим образом. В разделе 2 описан алгоритм, обобщающий эту эвристику на случай нескольких полос. Оценка точности алгоритма обоснована в разделе 3. В заключение, в разделе 4 описан способ управления вычислительными ресурсами на основе разработанного алгоритма.

2. Формулировка результата

Пусть задано множество полос $\{S_i\}_{i=1}^M$, ограниченных снизу. Ширина i -й полосы — W_i . Требуется упаковать прямоугольники из списка L в полосы. В качестве критерия эффективности упаковки используется максимальная среди всех полос высота заполнения.

Рассмотрим следующий эвристический алгоритм упаковки. Прямоугольники выбираются последовательно из списка L . Для размещения прямоугольника среди всех полос выбирается та, на которой прямоугольник может быть размещён на минимальной высоте. На выбранной полосе прямоугольник размещается на минимальной высоте в самой левой позиции из возможных. Пусть H_{BL} — максимальная высота заполнения среди всех полос при использовании описанной эвристики.

Пусть H_{OP} — минимально возможное значение критерия при заданном списке прямоугольников.

Теорема 1. Если в списке L прямоугольники упорядочены по ширине, тогда

$$\frac{H_{OP}}{H_{BL}} \geq \frac{1}{3}. \quad (1)$$

3. Доказательство теоремы 1

Пусть H_i высота заполнения в i -й полосе при использовании эвристики, тогда

$$H_{BL} = \max_{1 \leq i \leq M} H_i.$$

В каждой полосе S_i выделим какой-нибудь прямоугольник p_i с высотой h_i и шириной w_i , верхняя граница которого находится на высоте H_i . Пусть нижняя граница прямоугольника p_i находится на высоте y_i , тогда

$$H_i = y_i + h_i. \quad (2)$$

Пусть максимальное значение высоты y_i достигается в полосе S_{i_0} , т.е.

$$i_0 \in \operatorname{Argmax}_{1 \leq i \leq M} y_i. \quad (3)$$

Обозначим через Π множество всех полос, в которых возможно разместить p_{i_0} :

$$\Pi = \{S_i \mid 1 \leq i \leq M, W_i \geq w_{i_0}\}.$$

Покажем, что для каждой полосы из Π область, ограниченная высотой y_{i_0} , имеет плотность упаковки не меньше $\frac{1}{2}$ прямоугольниками ширины, не превосходящей w_{i_0} . Для доказательства воспользуемся техникой описанной в [3] для одной полосы. Рассмотрим момент, в который был упакован прямоугольник p_{i_0} . Так как список L упорядочен, то все прямоугольники, упакованные к этому моменту, не меньше p_{i_0} по ширине. Рассмотрим произвольную полосу $S_i \in \Pi$. Любую горизонтальную прямую пересекающую i -ю полосу можно разбить на участки внутри полосы занятые и незанятые прямоугольниками. Степень заполнения можно получить «интегрируя» по высоте относительную длину заполненных участков на каждой из таких прямых. Покажем, что в выбранной полосе S_i для любой горизонтальной прямой проходящей ниже y_{i_0} суммарная длина занятых частей не меньше суммарной длины незанятых. Так как множество прямых, содержащих верхние или нижние стороны прямоугольником, имеет меру нуль,

они не вносят вклад при подсчёте степени заполнения, поэтому, для удобства, можно ограничиться прямыми, не содержащими верхние или нижние стороны прямоугольников.

В силу того, что p_{i_0} упакован выше y_{i_0} , любая горизонтальная прямая ниже y_{i_0} пересекает хотя бы один прямоугольник в полосе S_i . Докажем, что левая сторона первого слева прямоугольник p' , пересекаемого любой прямой, лежит на левой границе полосы. Предположим это не так, тогда, в силу используемого алгоритма упаковки, должен существовать прямоугольник p'' , к которому p' прилегает справа. Ширина незаполненного отрезка прямой лежащего левее p' не меньше $w'' > w_{i_0}$. В силу используемого способа упаковки не один прямоугольник не был упакован над незаполненным отрезком. Следовательно, мы пришли к противоречию, так как прямоугольник p_{i_0} можно было бы разместить над p'' и его нижняя граница оказалась бы ниже высоты y_{i_0} .

Теперь рассмотрим отрезок s незаполненного пространства произвольной прямой, лежащий ниже y_{i_0} . Пусть p — прямоугольник, граничащий с s слева. Так как p_{i_0} упакован выше y_{i_0} , длина s должна быть меньше w_{i_0} (иначе существовал бы прямоугольник, который мог быть упакован ниже). Прямоугольник p упакован раньше p_{i_0} , поэтому ширина p больше w_{i_0} . Таким образом, для каждого отрезка незаполненного пространства существует прямоугольник, имеющий ширину больше, чем длина этого отрезка, примыкающий непосредственно слева к нему отрезку. Следовательно, суммарная длина занятых частей меньше суммарной длины незанятых, откуда получаем, что полоса S_i ниже y_{i_0} заполнена хотя бы наполовину. В силу произвольности выбора полосы мы получаем, что для любой полосы из Π область, лежащая ниже уровня y_{i_0} , имеет плотность заполнения не меньше $\frac{1}{2}$ прямоугольниками, которые нельзя разместить ни в одной из полос не принадлежащих Π . Таким образом, получаем нижнюю оценку на высоту оптимальной упаковки:

$$H_{OP} \geq \frac{1}{2} \left(\sum_{S_i \in \Pi} W_i y_{i_0} \right) \left(\sum_{S_i \in \Pi} W_i \right)^{-1} = \frac{y_{i_0}}{2}. \quad (4)$$

Среди всех полос выберем полосу с максимальной высотой заполнения:

$$k \in \operatorname{Argmax}_{1 \leq i \leq M} H_i = \operatorname{Argmax}_{1 \leq i \leq M} (y_i + h_i).$$

Тогда с одной стороны можно выразить значения высоты упаковки при использовании эвристики,

$$H_{BL} = H_k = y_k + h_k, \quad (5)$$

с другой стороны оценить снизу оптимальную высоту упаковки,

$$H_{OP} \geq h_k. \quad (6)$$

Собирая вместе (2), (3), (4), (5) и (6), получаем

$$\frac{H_{OP}}{H_{BL}} \geq \frac{\max(\frac{1}{2}y_{i_0}, h_k)}{H_k} = \frac{\max(\frac{1}{2}y_{i_0}, h_k)}{y_k + h_k} \geq \frac{\max(\frac{1}{2}y_{i_0}, h_k)}{y_1 + h_k} \geq \frac{1}{3},$$

откуда следует утверждение теоремы.

4. Составление расписаний для кластеров

Рассмотрим следующую упрощённую модель управления вычислительными ресурсами. Имеется набор кластеров, каждый с заданным числом процессоров. Распределитель (брокер) распределяет имеющиеся у него задачи среди этих кластеров. Для каждой задачи известно необходимые для её выполнения время и число процессоров. При этом брокер не занимается напрямую составлением расписаний на каждом кластере. Тогда описанный в разделе 2 алгоритм может быть использован для составления расписаний следующим образом.

Брокер последовательно выбирает задачи из имеющегося у него списка, и опрашивает кластеры, определяя, когда каждый из кластеров может выполнить задачу в соответствии с очередью на нем. Для размещения задачи брокер выбирает тот кластер, на котором задача завершится раньше всего. Каждый из кластеров, получая задачу от брокера, размещает её как можно раньше, используя по возможности процессоры с меньшими номерами.

Описанную процедуру можно представить как упаковку прямоугольников в полубесконечные полосы, где прямоугольники соответствуют задачам, а полосы кластерам. Ширина и высота прямоугольника представляет необходимые для выполнения задачи число процессоров и время, соответственно. Ширина полосы определяет число процессоров, доступное на соответствующем кластере. Основное отличие при составлении расписаний от размещения прямоугольников заключается в том, что при размещении задачи на кластере её можно размещать не только на подряд идущих процессорах, но и разбивать на вертикальные части, каждая из которых работает на своём процессоре в едином для всех частей временном интервале.

Аналогично тому, как это было сделано в предыдущей части, можно оценить точность эвристики. При этом для оценки плотности заполнения можно воспользоваться тем, что суммарная длина свободного пространства вдоль каждой горизонтальной прямой меньше ширины p_{i_0} , и таким образом получить, что плотность заполнения ниже y_{i_0} не меньше половины. Так как оценка на плотность не изменится, то и не изменится оценка

на высоту оптимального заполнения. Таким образом, останется в силе гарантированная оценка на точность упаковки.

Литература

- [1] A. Abraham, R. Buyya, B. Nath, Nature's heuristics for scheduling jobs on Computational Grids, International Conference on Advanced Computing and Communications (2000).
- [2] B.S. Baker, D.J. Brown and H.P. Katseff, A 5/4 algorithm for two-dimensional packing, J. of Algorithms, 1981, v. 2, pp. 348–368.
- [3] B.S. Baker, E. J. Coffman and R.L. Rivest, Orthogonal packings in two dimensions, SIAM J. Computing, 1980, v. 9, pp. 846–855.
- [4] P. Brucker, Scheduling Algorithms, Springer Verlag (1998), 217–218.
- [5] E.G. Coffman, Jr., M.R. Garey, and D.S. Johnson. Approximation Algorithms for Bin-Packing – An Updated Survey. In Algorithm Design for Computer System Design, ed. by Ausiello, Lucertini, and Serafini. Springer-Verlag, 1984.
- [6] E. J. Coffman, M. R. Garey, D. S. Johnson and R. E. Tarjan, Performance bounds for level-oriented two-dimensional packing algorithms, SIAM J. Computing, 1980, v. 9, pp. 808–826.
- [7] C. Kenyon and E. Remila, A near optimal solution to a two-dimensional cutting stock problem, Mathematics of Operations Research, 25 (2000), 645–656.
- [8] R. E. Korf, Optimal rectangle packing: Initial results, in Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS-03), Trento, Italy, June 2003, pp. 287-295.
- [9] D.D. Sleator, A 2.5-times optimal algorithm for bin packing in two dimensions, Inf. Processing Letters, 1980, v. 10, pp. 37–40.
- [10] J.D. Ullman, *Complexity of sequencing problems*, Computer and Job-shop Scheduling Theory, E. G. Coffman, Jr. (ed), John Wiley and Sons, New York, 1975.