

Алгоритм имитации отжига для упаковки прямоугольных предметов в параллельные полосы

Акимов Алексей Александрович

Научный руководитель: Кочетов Ю.А. д-р физ.-мат. наук, проф.

Новосибирский государственный университет

23 июня 2022

Неформальная постановка задачи

Имеется сервер облачного сервиса.

На сервере расположено несколько ядер и ограниченное количество виртуальных машин, которые нужно разместить на параллельно стоящих ядрах так, чтобы сервер завершил работу за минимальное время. Виртуальные машины рассматриваются как прямоугольники.

Высота такого прямоугольника равна времени работы виртуальной машины, а ширина прямоугольника - мощности соответствующей виртуальной машины. Ядра считаются параллельными полосами. Нужно минимизировать итоговую высоту этих полос, т.е. максимальную из высот сделать как можно меньше.

Сформулированная задача является как минимум NP-трудной, поскольку ее частный случай SPP - NP-трудная задача.

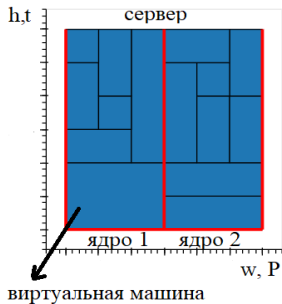


Рис. 1: Неформальная постановка задачи

Математическая постановка задачи

$I = \{1, 2, \dots, n\}$ — множество прямоугольников;

$J = \{1, 2, \dots, m\}$ — множество полос;

W - ширина полосы

w_i и h_i — ширина и высота i -го прямоугольника.

x_i и y_i — координаты левого нижнего угла i -го прямоугольника;

H - высота упаковки;

$l_{ik} \in \{0, 1\}$ равняется единице, если i -й прямоугольник находится левее k -го, и нулю в противном случае;

$b_{ik} \in \{0, 1\}$ равняется единице, если i -й прямоугольник находится ниже k -го, и нулю в противном случае.

$c_{ij} \in \{0, 1\}$ равняется единице, если i -й прямоугольник находится в j -ой полосе

$$\min H$$

$$y_i + h_i \leq H, i \in I \quad (1)$$

$$y_i \geq 0, i \in I \quad (2)$$

$$\sum_{j=1}^m c_{ij} = 1, i \in I \quad (3)$$

$$x_i + w_i \leq W \cdot (j - 1) + W \cdot (m \cdot (1 - c_{ij}) + c_{ij}), i \in I, j \in J \quad (4)$$

$$x_i \geq (j - 1) \cdot W \cdot c_{ij}, i \in I, j \in J \quad (5)$$

$$y_i + h_i \leq y_k + (1 - b_{ik}) \cdot B, i, k \in I \quad (6)$$

$$x_i + w_i \leq x_k + (1 - l_{ik}) \cdot B, i, k \in I \quad (7)$$

$$l_{ik} + l_{ki} + b_{ik} + b_{ki} = 1, k \neq i, k, i \in I \quad (8)$$

Кодировка решения O-tree

Упаковка называется L-компактной, если ни один прямоугольник нельзя сдвинуть влево при условии, что остальные прямоугольники остаются неподвижными. Упаковку можно представить с помощью ордерова, если она удовлетворяет одному из четырех определений компактности. В дальнейшем будем использовать в основном L-компактные упаковки.

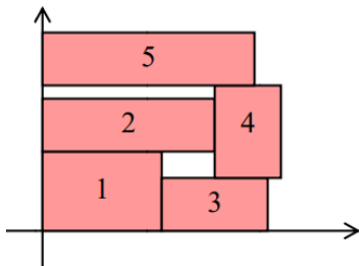


Рис. 2: LB-компактная упаковка.

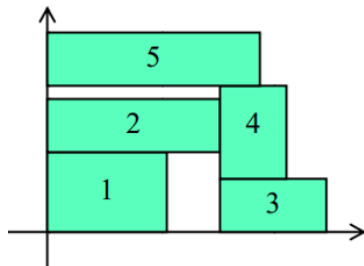


Рис. 3: B-компактная упаковка.

Кодировка упаковки в L-дерево

Корнем дерева будет являться левая граница полосы, дуги направлены от корня к листьям. Каждый прямоугольник обозначаем вершиной. Дуги ставятся между вершинами, соответствующим таким прямоугольникам, которые соприкасаются по оси x и имеют пересечение проекций на ось y . Родительской из возможных всегда берется вершина, соответствующая прямоугольнику с наименьшим значением координаты y . По заданной L-компактной упаковке дерево можно построить за время $O(n^2)$.

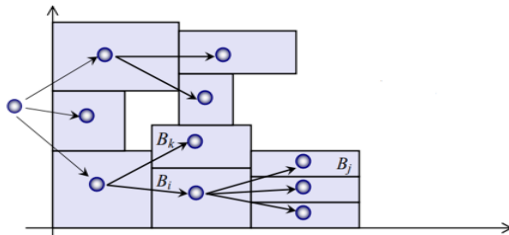


Рис. 4: L-дерево для упаковки

Декодировка L-дерева в упаковку

Преобразование L-дерева в упаковку осуществляется с помощью алгоритма обхода в глубину. Новый прямоугольник добавляется в соответствии с порядком обхода в глубину, с вершиной $(x, y) := (x_p + w_p, y_m + h_m)$. Где x_p, w_p параметры прямоугольника, соответствующего родительской вершине, а y_m, h_m - это параметры прямоугольника, имеющего пересечение проекций с текущим на ось x , а также имеющего наибольшую сумму по $y + h$. То есть самого высокого прямоугольника, который располагается под новым. Процедура декодирования занимает $O(n)$ времени.

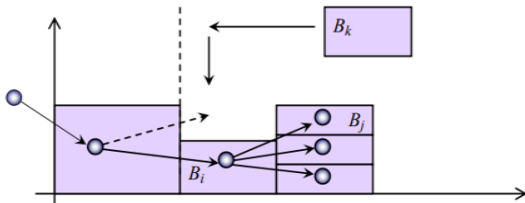


Рис. 5: Декодировка упаковки.

Обобщение на несколько полос

В случае нескольких полос кодировка происходит последовательно для каждой, т.е. упаковке прямоугольников в данной полосе ставится в соответствие свое дерево.

Аналогично для декодировки: дерево для каждой из полос последовательно декодируется.

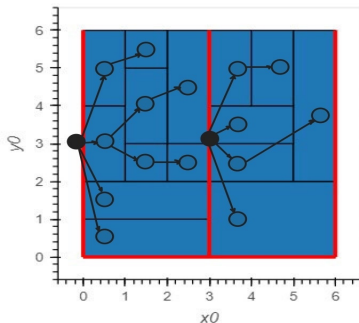


Рис. 6: Два дерева для двух полос

Окрестность решения

Окрестностью текущего решения называется множество всевозможных решений, полученных путем однократного применения некоторой операции к текущему решению. Рассмотрим следующие операции над ориентированными деревьями:

O1 Перестановка вершин деревьев вместе с соответствующими прямоугольниками.

O2 Перемещение листа дерева и соответствующего ему прямоугольника, в новую позицию листа.

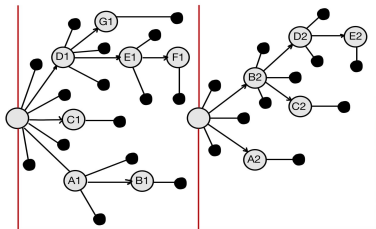


Рис. 7: Позиции для вставки листа

Декодировка приведенная ранее, на самом деле, не гарантирует, что упакованные предметы не пересекают допустимую ширину. Поэтому введем понятие допустимого решения.

Решение будем называть допустимым, если ни один прямоугольник из упаковки не выходит за данную ширину полосы, в которой он находится.

Если решение недопустимое, то целевая функция домножается на штраф $f = \sum_{j=1}^m (W_{maxj} - W \cdot j), j \in J$.

Конечным результатом может являться только допустимое решение.

SA представляет собой вероятностную процедуру локального поиска. Он может совершать шаги как улучшающие, так и ухудшающие целевую функцию, что позволяет переходить от локальных оптимумов к другим решениям.

Алгоритм:

1. Выбрать начальное решение i и вычислить $F(i)$
2. Задать начальную температуру T и коэффициент охлаждения r
3. Пока не выполнен критерий остановки делать следующее:
 - 3.1. Выполнить цикл L раз
 - 3.1.1. Выбрать в $N(i)$ случайным образом решение j
 - 3.1.2. Положить $\Delta = F(j) - F(i)$
 - 3.1.3. Если $\Delta \leq 0, i := j$
 - 3.1.4. Если $\Delta > 0$, то с вероятностью $e^{-\frac{\Delta}{T}}$ положить $i := j$
 - 3.2. Понизить температуру $T := T \cdot r$

Типичное поведение алгоритма

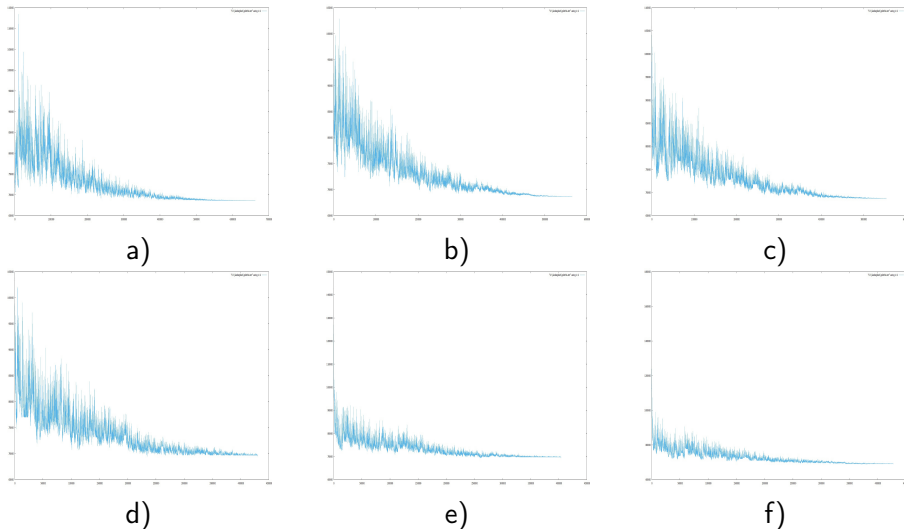
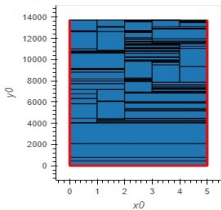
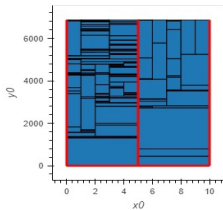


Рис. 8: Графики поведения алгоритма на: а) одной полосе, б) двух полосах, в) трех полосах, г) четырех полосах, д) пяти полосах, е) шести полосах.

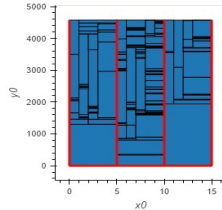
Графические результаты работы



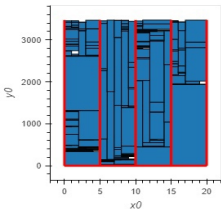
a)



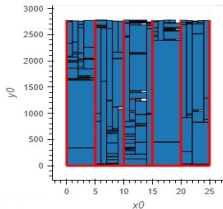
b)



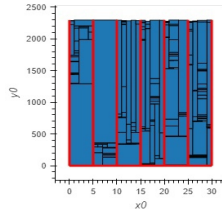
c)



d)



e)



f)

Рис. 9: Упаковки набора в 209 прямоугольников для: а) одной полосы, б) двух полос, с) трех полос, д) четырех полос, е) пяти полос, ф) шести полос.

Сравнение результатов для 1 полосы

Набор	59	76	100	118	164	172	209
SA + O-tree							
Лучшее, %	0,001	0,013	0	0,013	0,003	0,005	0,005
Среднее, %	0,273	0,112	0,254	0,142	0,06	0,01	0,008
Худшее, %	1,925	0,389	1,31	1	0,693	0,027	0,013
Упаковок, тыс.	132	245	650	995	1989	2220	3358
GA + Skyline							
Лучшее, %	0,001	0	0	0	0,003	0,006	0,006
Среднее, %	0,001	0,0001	0	0	0,003	0,006	0,006
Худшее, %	0,001	0,007	0	0	0,003	0,006	0,006
Упаковок, тыс.	158	225	241	396	570	482	440
Gurobi							
Dead space, %	0	1,863	4,05	2,919	5,974	2,585	58,12

Gurobi на всех примерах(кроме 59) работал 1 час.

Сравнение результатов для 2 полос

Набор	59	76	100	118	164	172	209
SA + O-tree							
Лучшее, %	0,063	0,118	0,059	0,013	0,024	0,027	0,02
Среднее, %	1,053	0,777	1,071	0,953	0,319	0,447	0,134
Худшее, %	2,956	2,205	2,531	2,214	1,564	2,186	0,915
Упаковок, тыс.	190	178	437	609	1638	1779	2959
GA + Skyline							
Лучшее, %	0,009	0,007	0,012	0,013	0,01	0,013	0,02
Среднее, %	0,026	0,039	0,017	0,151	0,025	0,044	0,06
Худшее, %	0,096	0,105	0,107	0,8	0,082	0,145	0,238
Упаковок, тыс.	158	270	366	465	410	170	265
Gurobi							
Dead space, %	0,009	3,509	2,666	2,263	10,658	16,252	9,057

Gurobi на всех примерах(кроме 59) работал 1 час.

Сравнение результатов для 4 полос

Набор	118	164	172	209	213
SA + O-tree					
Лучшее, %	19,8297	0,169	0,086	0,093	3,37124
Среднее, %	19,8297	4,77	1,725	0,962	3,37124
Худшее, %	19,8297	12,047	3,71	2,284	3,37124
Упаковок, тыс.	72	648	1162	2169	1263
GA + Skyline					
Лучшее, %	19,895	0,082	0,087	0,209	3,397
Среднее, %	19,895	0,238	0,263	0,702	3,4
Худшее, %	19,895	0,6	0,584	1,414	3,424
Упаковок, тыс.	145	280	114	86	151
Gurobi					
Dead space, %	19,8297	9,936	6,953	37,937	3,37124

Gurobi на всех примерах работал 1 час.

Алгоритм имитации отжига для упаковки прямоугольных предметов в параллельные полосы

Акимов Алексей Александрович

Научный руководитель: Кочетов Ю.А. д-р физ.-мат. наук, проф.

Новосибирский государственный университет

23 июня 2022

Контур упаковки

Используется дополнительная структура данных для хранения информации о текущем контуре упаковки. Контур состоит из уже размещенных прямоугольников, покрывающих частичную упаковку сверху.

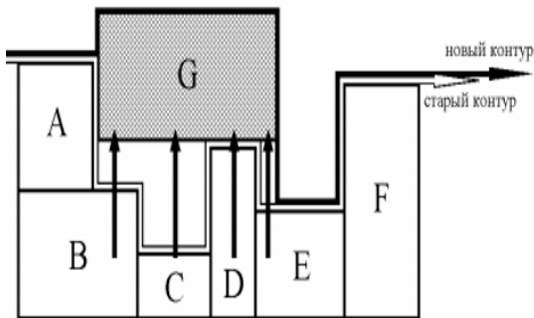
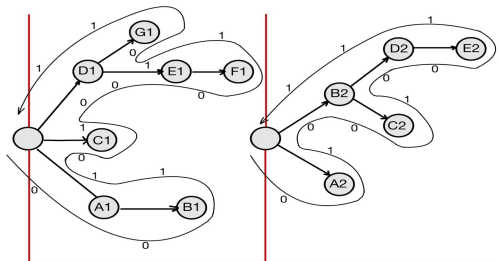


Рис. 10: Использование контура.

Кодировка ордерова

Структуру ориентированного дерева можно представить с помощью двоичной последовательности τ и перестановки π . Такая кодировка производится с помощью обхода в глубину, в порядке которого заполняются π и τ . В перестановке π хранятся проименованные вершины дерева, а двоичная последовательность τ , в свою очередь, отображает структуру дерева. Цифра «0» в ней соответствует направлению движения вглубь дерева, а цифра «1» соответствует обратному направлению.



$$\pi_1 = A_1 B_1 C_1 D_1 E_1 F_1 G_1$$

$$\tau_1 = 00110100011011$$

$$\pi_2 = A_2 B_2 C_2 D_2 E_2$$

$$\tau_2 = 0100100111$$

Прямоугольники случайным образом поровну распределяются по полосам. Далее сортируются по невозрастанию высоты и самый высокий располагается в левом нижнем углу текущей полосы, тем самым инициализируя первый уровень, по высоте равный ему. Остальные прямоугольники располагаются слева направо пока есть место на текущем уровне. Прямоугольник, который не поместился на уровне, помещается сверху, образуя следующий уровень и так далее.

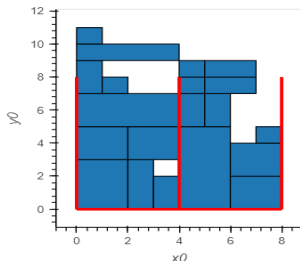


Рис. 11: Пример работы NFDH

Сравнение с LS skyline

Набор	59	76	100	118	164	172	209
LS							
Лучшее, %	0,212	0,105	0,154	0,8	0,025	0,028	0,079
Среднее, %	2,001	1,001	1,536	4,798	0,712	0,966	0,622
Худшее, %	4,743	6,581	3,024	11,854	2,989	4,563	3,276
SA + o-tree							
Лучшее, %	0,063	0,118	0,059	0,013	0,024	0,027	0,02
Среднее, %	1,053	0,777	1,071	0,953	0,319	0,447	0,134
Худшее, %	2,956	2,205	2,531	2,214	1,564	2,186	0,915

Таблица 1: Сравнение с алгоритмом локального поиска