



## Алгоритм для многомерной упаковки в контейнеры и его вероятностный анализ

<sup>1,2</sup> Д.О. Лазарев, ORCID: 0000-0002-6253-6447 <lazarev@ispras.ru>

<sup>1</sup> Институт системного программирования им. В.П. Иванникова РАН,  
Россия, 109004, г. Москва, ул. А. Солженицына, д. 25.

<sup>2</sup> НИУ Московский физико-технический институт,  
Россия, 141701, Московская область, г. Долгопрудный, Институтский переулок, д.9.

**Аннотация.** Проводится анализ в среднем задачи упаковки  $d$ -мерных прямоугольных параллелепипедов в контейнеры. Для решения задачи, предложен алгоритм увеличения размерности, позволяющий эффективно строить алгоритм упаковки в  $(d+1)$ -мерные контейнеры, используя алгоритм упаковки в  $d$ -мерные контейнеры. При упаковке, стремимся минимизировать ожидаемый объем незаполненного пространства использованных контейнеров. Ранее известные результаты верхней оценки числа упаковываемых параллелепипедов улучшены. Уменьшение объема незаполненного пространства подтверждают проведенные вычислительные эксперименты. При анализе алгоритмов в среднем, может возникать переобучение. Связано оно с тем, что минимальный ожидаемый объем незаполненного пространства может достигаться для одного конкретного распределения, или для семейства распределений входных параметров алгоритма, в решаемой задаче входные параметры – длины сторон параллелепипедов. Для борьбы с данным нежелательным эффектом, предложенный алгоритм увеличения размерности был обобщен на случай произвольного заранее известного распределения длин сторон параллелепипедов. Эффективность обобщенного алгоритма подтверждена вычислительными экспериментами.

**Ключевые слова:** упаковка в контейнеры; многомерная задача упаковки в контейнеры; вероятностный анализ; переобучение; алгоритм увеличения размерности.

**Для цитирования:** Лазарев Д.О. Алгоритм для многомерной упаковки в контейнеры и его вероятностный анализ. Труды ИСП РАН, том 38, вып. 2, 2026 г., стр. 35–52. DOI: 10.15514/ISPRAS–2026–38(2)–3.

**Благодарности:** Исследования поддержаны фондом отдела теоретической информатики Института системного программирования им. В.П. Иванникова РАН. Результаты получены с использованием услуг Центра коллективного пользования Института системного программирования им. В.П. Иванникова РАН – ЦКП ИСП РАН. Автор выражает особую благодарность Шокурову А.В. за ценные замечания в процессе работы над статьей.

# An Algorithm for Multi-dimensional Bin Packing Problem and Its Average Case Analysis

<sup>1,2</sup> D.O. Lazarev, ORCID: 0000-0002-6253-6447 <lazarev@ispras.ru>

<sup>1</sup> *Ivannikov Institute for System Programming of the Russian Academy of Science, 25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.*

<sup>2</sup> *Moscow Institute of Physics and Technology (National Research Institute), 9, Institute alley, Dolgoprudny, Moscow Region, 141701, Russia.*

**Abstract.** The authors consider a multidimensional bin packing problem. For solving the problem, an algorithm is proposed and its average-case analysis is carried out. The algorithm incrementally increases the dimension of hyperrectangles packed, and allows efficiently pack into  $(d+1)$ -dimensional bins using an algorithm for packing into  $d$ -dimensional bins. During the packing process, our objective is to minimize the expected wasted space of used bins. Best previously known results of the upper bound of the number of hyperrectangles packed were improved. The decrease of expected wasted space is confirmed by computational experiments. The likely problem of average-case analysis is overfitting. It is due to the fact that the minimal expected wasted space is studied for only one distribution or family of distributions of input parameters of the algorithm. In order to deal with overfitting, the proposed algorithm was generalized to the case of any given in advance distribution of rectangles' sides sizes. The efficiency of the generalized algorithm is confirmed by computational experiments.

**Keywords:** bin packing; multidimensional bin packing; average case analysis; overfitting; dimension increment algorithm.

**For citation:** Lazarev D.O. An algorithm for multi-dimensional bin packing problem and its average case analysis. *Trudy ISP RAN/Proc. ISP RAS*, vol. 38, issue 2, 2026, pp. 35-52 (in Russian). DOI: 10.15514/ISPRAS-2026-38(2)-3.

**Acknowledgements.** The research was supported by the fund of the Department of Theoretical Computer Science of Ivannikov Institute for System Programming of the RAS (ISP RAS). The results were obtained using the services of the ISP RAS Data Center. The author is especially grateful to A.V. Shokurov for valuable comments during the work on the article.

## 1. Введение

Задача упаковки многомерных объектов в контейнеры имеет многочисленные практические приложения. Так, одномерная задача упаковки в контейнеры, в англоязычной литературе известная как “bin packing problem”, возникла в силу потребности решения задач форматирования таблиц или аллокации файлов [1]. Также, одномерная задача упаковки в контейнеры, позволяет оптимизировать погрузку грузовых автомобилей с заданным ограничением веса [2]. Двумерная задача упаковки в контейнеры, моделирует проблему раскроя и перевозки материалов, задачу оптимизации размещения объектов на плоскости, а также проблему пакетной обработки вычислительных задач [3]. Трехмерная задача упаковки учитывает еще одно измерение и позволяет моделировать многие промышленные приложения: оптимизацию размещения трехмерных объектов на складах, в вагонах и самолетах [4].

В настоящее время, в силу быстрого роста популярности распределенных вычислений, широкого распространения вычислительных кластеров, грид-технологий, а также облачных вычислений, интерес к задачам упаковки возрастает в связи с новыми приложениями: задачами управления ресурсами распределенных вычислительных систем и развитием техники облачных вычислений [5-7].

## 1.1 История исследования задачи упаковки в контейнеры и классические результаты

Задача упаковки в контейнеры является одной из первых известных NP-трудных в сильном смысле задач [8]. Для ее решения было предложено множество приближенных алгоритмов и получены оценки их качества в наихудшем и в среднем случаях. В настоящей работе рассматриваются алгоритмы, работающие в режиме онлайн, для которых размер и положение очередного упаковываемого объекта неизвестны в момент упаковки предыдущего.

Опишем два важных алгоритма упаковки объектов в одномерные контейнеры:

- **First fit (FF).** При упаковке алгоритмом *first fit*, каждый следующий объект размещается в последний созданный подходящий контейнер. Если же для него не подходит ни один из существующих контейнеров, то для его размещения создается новый контейнер.
- **Best fit (BF).** Каждый следующий объект размещается в наиболее плотно заполненный подходящий контейнер. Если для него не подходит ни один из существующих контейнеров, то для его упаковки создается новый.

Оба приведенных выше алгоритма принадлежат классу алгоритмов *any fit*, в который входят все алгоритмы, работающие в режиме онлайн и размещающие очередной объект в один из уже созданных контейнеров в том случае, если имеется хотя бы один подходящий из созданных контейнеров.

Приведем лучшие известные результаты для задачи упаковки в контейнеры при анализе в среднем и в худшем случаях. При исследовании алгоритма  $A$  упаковки в худшем случае, исследуем асимптотическую точность  $R^\infty$ , или минимальное по всем существующим наборам упаковываемых объектов отношение числа контейнеров, необходимых для упаковки с использованием алгоритма  $A$  к числу контейнеров, в которые оптимальный алгоритм может упаковать эти же наборы параллелепипедов при стремлении к бесконечности числа упаковываемых объектов. При анализе в среднем случае, исследуется математическое ожидание  $W$  объема незаполненного пространства контейнеров после упаковки в случае, когда длины сторон упаковываемых параллелепипедов принадлежат известному распределению или семейству распределений.

При анализе в худшем случае одномерной задачи, было показано [1], что асимптотическая точность работающего в режиме онлайн алгоритма *first fit* составляет 1.7, и что асимптотическая точность любого алгоритма из *any fit* не превосходит асимптотической точности алгоритма *first fit*. Для алгоритмов не из класса *any fit*, алгоритм с лучшей из известных верхних оценок асимптотической точности  $R^\infty \leq 1.541$  был предложен в работе [9]. Наилучшая известная нижняя оценка была получена в [10]. Было доказано, что для любого алгоритма, работающего в режиме онлайн,  $R^\infty \geq 1.536$ .

При анализе худшего случая алгоритмов, работающих в режиме онлайн, для многомерной задачи упаковки в контейнеры, лучшие известные нижние оценки [11] для асимптотической точности, полученные в случае двумерной задачи, равны 1.802, а в случае трехмерной задачи составляют 1.974. Лучший из известных алгоритм [12] для двумерной задачи имеет асимптотическую точность 2.554, для случая  $d$ -мерной задачи в [13] был предложен алгоритм с  $R_d^\infty = P_\infty^d$ , где  $P_\infty \approx 1.691$ .

При анализе в среднем случае алгоритмов, работающих в режиме онлайн, для задачи упаковки в контейнеры выделяют постановки с незадаанным завершением (англ. open-end), когда точное число объектов неизвестно до выпадения символа останова после упаковки последнего объекта, и постановку с известным завершением (англ. closed-end), когда число объектов известно до начала упаковки объектов в контейнеры (подробнее – в разделе “Постановка задачи упаковки в контейнеры”).

В [14] был предложен алгоритм для одномерной задачи с известным завершением с ожидаемым объемом незаполненного пространства контейнеров  $W_N(N) = \theta(\sqrt{N})$ , сначала упаковывающий первые  $\lfloor \frac{N}{2} \rfloor$  объектов в отдельный контейнер каждый, а затем – упаковывающий оставшиеся объекты с использованием алгоритма *BF*. (Здесь  $W_N(N)$  – математическое ожидание объема контейнеров, незаполненного объектами после упаковки  $N$  объектов). Для алгоритмов с незадаанным завершением, также имеется алгоритм [15], достигающий нижней оценки [14], равной  $\Omega(\sqrt{N \ln N})$ . Отметим также, что алгоритм *BF* упаковки с незадаанным завершением, согласно [16], имеет качество  $W_N(N) = \theta(\sqrt{N \ln^3 N})$ , близкое к оптимальному.

При анализе в среднем, для  $d$ -мерной задачи упаковки в контейнеры, был предложен алгоритм [17] с незадаанным завершением с  $W_{d,N}(N) = \theta\left(N^{\frac{d+1}{d+2}}\right)$ . Данный результат улучшен в настоящей работе до  $W_{d,N}(N) = O\left(N^{\frac{d}{d+1}} \ln^{\frac{3}{2(d+1)}} N\right)$ .

## 2. Постановка задачи упаковки в контейнеры

Приведем необходимые определения. Определим, следуя работе [13], многомерную задачу упаковки в контейнеры (англ. multi-dimensional bin packing problem) в постановке упаковки в гиперкубы (англ. box packing). Дан набор, состоящий из  $N$   $d$ -мерных открытых прямоугольных параллелепипедов, которые в дальнейшем для краткости будем называть параллелепипедами. Требуется упаковать параллелепипеды без вращений и попарных пересечений в как можно меньшее число контейнеров. Контейнер – это  $d$ -мерный гиперкуб со стороной единица. Стороны параллелепипедов и контейнеров параллельны осям прямоугольной декартовой системы координат.

Рассматриваем алгоритмы, работающие в режиме онлайн, или онлайн-алгоритмы, на вход которым последовательно поступают параллелепипеды. Для каждого упаковываемого параллелепипеда выделим, возможно пустые, множества параллелепипедов-предшественников и параллелепипедов-последователей, которые подаются алгоритму на вход до и после упаковки рассматриваемого параллелепипеда соответственно. Онлайн-алгоритмы упаковывают каждый следующий параллелепипед, используя лишь размеры параллелепипедов-предшественников и координаты, по которым они расположены, но не обладая информацией ни о размерах, ни о расположении параллелепипедов-последователей. Распределение длин сторон всех параллелепипедов обычно известно алгоритму до начала упаковки при вероятностном анализе алгоритмов.

Проводится вероятностный анализ алгоритмов упаковки в предположении, что длины сторон, расположенных по  $i$ -ой координате каждого из параллелепипедов, независимы в совокупности между собой и с другими длинами сторон и имеют равномерное на  $[0,1]$  распределение  $\forall i \in \{1, \dots, d\}$ . Также приведено естественное обобщение предложенного алгоритма на случай общего распределения длин сторон параллелепипедов.

Рассматриваются онлайн-алгоритмы с незадаанным завершением. В работе [14] приведено следующее описание работы алгоритма с незадаанным завершением:

1. Выберем случайно равномерно  $k$  из набора  $\{1, \dots, N\}$ .
2. Подадим на вход алгоритму последовательность из  $k$  случайных параллелепипедов, которые алгоритм должен упаковать в режиме онлайн.
3. Подадим алгоритму на вход символ останова.

Здесь  $N$  – натуральное число, известное алгоритму до начала работы.

Обозначим через  $A_{d,N}$  алгоритм упаковки в  $d$ -мерные контейнеры, упаковки с заранее известным максимальным количеством  $N$  параллелепипедов, поступающих на вход. Для алгоритма  $A_{d,N}$  за  $W_{d,N}(k)$  обозначим объем незаполненного параллелепипедами пространства частично заполненных контейнеров после упаковки  $k \leq N$  параллелепипедов с использованием алгоритма  $A_{d,N}$ . Плотность заполнения  $d$ -мерных контейнеров в работе [14] исследуется с использованием целевой функции  $V_{d,N}$ :

$$V_{d,N} = \frac{\sum_{k=1}^N \mathbb{E}W_{d,N}(k)}{N},$$

где математическое ожидание вычисляется по распределению длин сторон параллелепипедов.

Онлайновые алгоритмы, работающие по данному принципу и имеющие данную целевую функцию, будем называть алгоритмами с незадачным завершением с целевой функцией типа 1. В настоящей работе, будем также использовать следующую целевую функцию для оценки плотности заполнения контейнеров:

$$W_{d,N} = \max_{1 \leq k \leq N} \mathbb{E}W_{d,N}(k)$$

Данную целевую функцию будем называть целевой функцией типа 2. Оценка плотности заполнения контейнеров по целевой функции типа 1 никогда не превышает оценку по целевой функции типа 2. Таким образом, наиболее строгая оценка получается с использованием целевой функции типа 2.

В работе [17] был предложен алгоритм упаковки параллелепипедов в контейнеры с оценкой целевой функции типа 1:

$$V_{d,N} = \theta \left( N^{\frac{d+1}{d+2}} \right)$$

В настоящей работе, предложен алгоритм  $d$ -мерной упаковки параллелепипедов в контейнеры, работающий в режиме онлайн, с оценкой целевой функции второго типа

$$W_{d,N} = O \left( N^{\frac{d}{d+1}} \ln^{\frac{3}{2(d+1)}} N \right),$$

что асимптотически улучшает оценку из работы [17].

### 3. Алгоритм увеличения размерности и верхние оценки $W_{d,N}$

В настоящем разделе, опишем предложенный алгоритм увеличения размерности для многомерной упаковки в контейнеры и оценим математическое ожидание незаполненного пространства контейнеров при использовании данного алгоритма.

#### 3.1 Алгоритм увеличения размерности

Пусть для  $d$ -мерной задачи упаковки параллелепипедов в контейнеры, имеется множество алгоритмов  $A_{d,M} \forall M \in \mathbb{N}$ , работающих в режиме онлайн с незадачным завершением с оценкой целевой функции типа 2:

$$W_{d,M}(A_{d,M}) = \max_{1 \leq k \leq M} \mathbb{E}W_{d,M}(k) = O \left( M^{\frac{d}{d+1}} \ln^m M \right), m \geq 0$$

Построим для каждого  $N \in \mathbb{N}$  алгоритм  $A_{d+1,N}$ , также работающий в режиме онлайн, с незадачным завершением с

$$W_{d+1,N} = O \left( N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N \right)$$

Идея алгоритма заключается в разбиении пространства контейнеров на области и в последующей упаковке в одну область в контейнерах одного типа параллелепипедов с близкими длинами сторон по первой координате. Пространство контейнера разбивается на 2

области или, только для контейнеров последнего,  $s + 1$ -го типа, контейнеры содержат единственную область.

Число типов областей составляет  $2s + 1$ , эти области расположены в контейнерах одного из  $s + 1$  типов. Внутри области параллелепипеды размещаются с использованием алгоритма упаковки  $A_{d,N_d(N)}$  меньшей размерности, где параметр  $N_d(N)$ , равный верхней оценке числа упаковываемых параллелепипедов размерности  $d$ , будет введен в дальнейшем.

Аналог алгоритма из [17] можно построить также по индукции. Большее асимптотическое значение объема  $W_{d,N}$  незаполненного пространства для  $d$ -мерной задачи упаковки в контейнеры с использованием алгоритма из [17], по сравнению с предложенным в настоящей работе алгоритмом, возникает из-за более высокой асимптотики незаполненного пространства контейнеров для одномерной задачи упаковки в контейнеры:  $W_{1,N} = \theta\left(N^{\frac{2}{3}}\right)$  для алгоритма *hash packing* из [17], тогда как для алгоритма *BF*,  $W_{1,N} = \theta\left(\sqrt{N} \ln^{\frac{3}{4}} N\right)$ .

**Алгоритм увеличения размерности  $A_{d+1,N} = A_{d+1,N}(A_{d,N_d(N)})$ :**

1. Зафиксируем  $s = \left\lfloor N^{\frac{1}{d+2}} \ln^{-\frac{m(d+1)}{d+2}} N \right\rfloor$ .
2. Если выпал параллелепипед  $R$ , длина  $b_1(R)$  которого по первой координате принадлежит полуинтервалу  $\left(\frac{i-1}{2s+1}, \frac{i}{2s+1}\right]$ , то говорим, что параллелепипед  $R$  – типа  $i$ .
3. Размещение параллелепипеда по первой координате.
  - Если тип  $i$  параллелепипеда  $R$  не превосходит  $s$ , то по первой координате, сторону параллелепипеда длиной  $b_1(R)$  размещаем на интервале  $(0, b_1(R))$  в контейнере  $i$ -го типа. Областью  $i$ -го типа назовем подмножество контейнеров  $i$ -го типа, значение первой координаты которых принадлежит отрезку  $\left[0, \frac{i}{2s+1}\right]$ .
  - Если тип  $i$  параллелепипеда  $R$  принадлежит множеству  $\{s + 1, \dots, 2s\}$ , то по 1-ой координате, сторону параллелепипеда  $R$  длиной  $b_1(R)$  размещаем на интервале  $\left(1 - \frac{2s-i}{2s+1}, 1 - \frac{2s-i}{2s+1} + b_1(R)\right)$  в контейнере  $(2s + 1 - i)$ -го типа. Областью  $i$ -го типа назовем подмножество контейнеров  $(2s + 1 - i)$ -го типа, значение первой координаты которых принадлежит полуинтервалу  $\left(\frac{2s-i}{2s+1}, 1\right]$ .
  - Если  $i = 2s + 1$ , то по 1-ой координате, сторону параллелепипеда  $R$  длиной  $b_1(R)$  размещаем на интервале  $(0, b_1(R))$  в контейнере  $s + 1$ -го типа. Областью  $(2s + 1)$ -го типа назовем множество контейнеров  $(s + 1)$ -го типа.

Таким образом, для всех  $i \in \{1, \dots, 2s + 1\}$ , ширина  $i$ -ой области составляет  $\frac{i}{2s+1}$ .

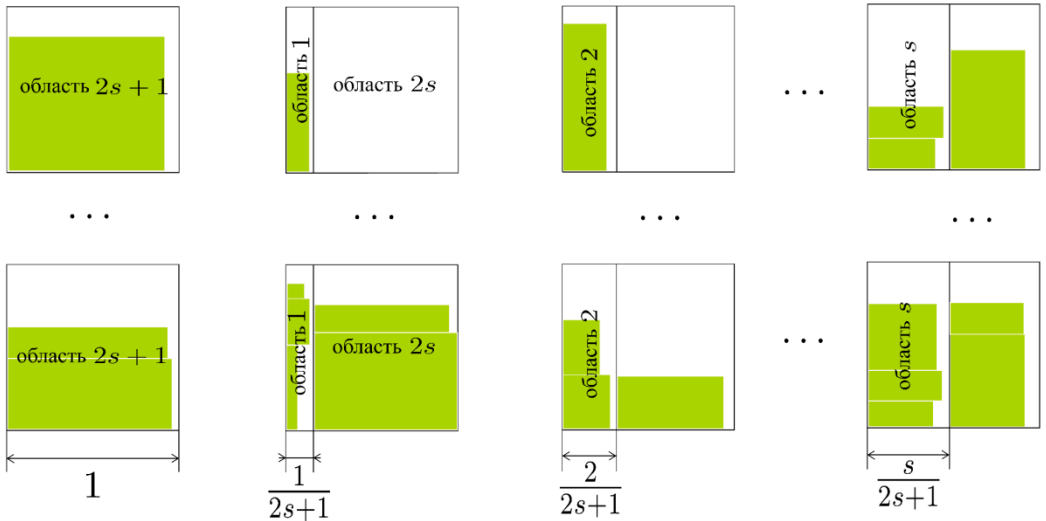
4. Размещение параллелепипеда по последним  $d$  координатам.

Проекцию параллелепипеда типа  $i$  на последние  $d$  координат, где  $i \in \{1, \dots, 2s\}$ , размещаем в контейнер типа  $\min\{i, 2s + 1 - i\}$ , а параллелепипед типа  $2s + 1$  – в контейнер типа  $s + 1$ , используя при этом алгоритм  $A_{d,N_d(N)}$  упаковки  $d$ -мерных параллелепипедов, которому известно, что нужно упаковать не более, чем  $N_d(N) = \left(1 + 4N^{-\frac{d}{2(d+2)}}\sqrt{\ln N}\right) \frac{N}{2s+1}$  параллелепипедов.

Параллелепипеды различных типов упаковываем в попарно непересекающиеся области. Если алгоритм  $A_{d,N_d(N)}$  для размещения проекции параллелепипеда  $R$  типа  $i$ ,  $i \leq 2s$  на последние  $d$  координат  $b_2(R), \dots, b_{d+1}(R)$ , требует создания нового  $d$ -мерного контейнера, и при этом существует  $d + 1$ -мерный контейнер  $B$  типа

$\min\{i, 2s + 1 - i\}$  в который упакованы лишь параллелепипеды  $(2s + 1 - i)$ -го типа, то для размещения  $R$  используется контейнер  $B$ .

На рис. 1 приведен пример построения алгоритма двумерной упаковки в контейнеры-квадраты с помощью одномерного алгоритма упаковки в контейнеры в виде отрезков с использованием алгоритма увеличения размерности.



В каждую область, объекты упаковываем с использованием алгоритма упаковки в контейнеры объектов на 1 меньшей размерности

Рис. 1. Пример построения алгоритма  $A_{2,N}$  двумерной упаковки в контейнеры на основе алгоритма  $A_{1,N_1(N)}$  для одномерной упаковки.

Fig. 1. Example of algorithm  $A_{2,N}$  of two-dimensional bin packing construction using  $A_{1,N_1(N)}$  algorithm for one-dimensional bin packing.

### 3.2 Оценка ожидаемого объема $W_{d,N}$ незаполненного пространства для алгоритма увеличения размерности

В приложении 1, доказана следующая теорема, позволяющая связать плотность заполнения при упаковке с помощью алгоритма  $A_{d+1,N}$  с плотностью заполнения с помощью алгоритма  $A_{d,N_d(N)}$  упаковки прямоугольников в контейнеры меньшей размерности:

**Теорема 1.** Пусть  $A_{d,M} \forall M \in \mathbb{N}$  – семейство алгоритмов с незадачным завершением для  $d$ -мерной задачи упаковки в контейнеры с оценкой целевой функции типа 2, равной

$$W_{d,M}(A_{d,m}) = O\left(M^{\frac{d}{d+1}} \ln^m M\right) \forall M \in \mathbb{N}, m \geq 0.$$

Тогда алгоритм  $A_{d+1,N} = A_{d+1,N}(A_{d,N_d(N)})$  для  $d + 1$ -мерной задачи, полученный в результате применения алгоритма увеличения размерности к алгоритму  $A_{d,N_d(N)}$ , работает в режиме с незадачным завершением и имеет оценку целевой функции типа 2, равную

$$W_{d+1,N}(A_{d+1,N}) = O\left(N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N\right).$$

Как следствие, применяя последовательно  $d - 1$  раз алгоритм увеличения размерности  $A_{d,N} \left( A_{d-1,N_{d-1}(N)} \left( A_{d-2,N_{d-2}(N_{d-1}(N))} (\dots) \right) \right)$  к алгоритму  $A_1$  для одномерной задачи упаковки в контейнеры, получаем следующую теорему:

**Теорема 2.** Пусть для одномерной задачи упаковки в контейнеры, существует алгоритм  $A_{1,N}$  с незадаанным завершением, целевая функция типа 2 для которого равна  $W_{1,N}(A_{1,N}) = O(\sqrt{N} \ln^m N)$ ,  $m \geq 0$  при упаковке не более  $N$  объектов случайного размера, имеющего равномерное на  $(0,1]$  распределение.

Тогда для каждого натурального  $d \geq 1$ , существует алгоритм  $A_{d,N}$  с незадаанным завершением для  $d$ -мерной задачи упаковки в контейнеры, полученный в результате применения алгоритма увеличения размерности  $d - 1$  раза к алгоритму  $A_1$ , со следующей асимптотической оценкой целевой функции типа 2:

$$W_{d,N} = O\left(N^{\frac{d}{d+1}} \ln^{\frac{2m}{d+1}} N\right)$$

Как показано в работе [16], для стандартной онлайн-оной эвристики с незадаанным завершением  $BF$  упаковки в одномерные контейнеры, верна оценка для целевой функции второго типа:

$$W_1^{BF} = W_{1,N}^{BF} = O\left(\sqrt{N} \ln^{\frac{3}{4}} N\right) \forall N,$$

причем верхняя оценка числа упаковываемых объектов  $N$  алгоритму  $BF$  неизвестна. Следовательно, по теореме 2, на основе одномерного алгоритма  $BF$ , можно построить алгоритм с незадаанным завершением  $d$ -мерной упаковки  $A_{d,N}(A_{d-1,N_{d-1}(N)}(\dots(BF)\dots))$  с оценкой целевой функции типа 2:

$$W_{d,N} = O\left(N^{\frac{d}{d+1}} \ln^{\frac{3}{2(d+1)}} N\right)$$

Такие алгоритмы будем называть алгоритмами увеличения размерности от  $BF$  (алгоритмами увеличения размерности  $(BF)$ , или *a.y.p.(BF)* размерности  $d$ .

### 3.3 Обобщение алгоритма увеличения размерности

Основным недостатком анализа алгоритмов упаковки в среднем случае является ограниченная практическая применимость, возникающая вследствие предположения о фиксированном распределении, или семействе распределений длин сторон параллелепипедов. Данное предположение может не выполняться на практике.

Алгоритм увеличения размерности может легко быть обобщен на случай любого заранее известного распределения длин сторон  $d$ -мерных параллелепипедов в том случае, когда длины сторон параллелепипедов независимы в совокупности. Для этого, аналогично случаю равномерного распределения длин сторон параллелепипедов, сначала выберем количество областей для упаковки параллелепипедов по первой координате, равное  $2s + 1$  в зависимости от верхней оценки  $N$  числа упаковываемых параллелепипедов. Ширина  $i$ -ой области изменится с  $\frac{i}{2s+1}$  до значения  $i$ -ого  $2s + 1$ -квантиля распределения 1-ой стороны параллелепипедов (т.е. такого значения, которое первая координата параллелепипеда не превосходит с вероятностью  $\frac{i}{2s+1}$ ) при изменении этого распределения с равномерного на общий случай распределения. Заметим также, что  $\frac{i}{2s+1}$  является  $i$ -ым  $2s + 1$ -квантилем равномерного на  $(0,1]$  распределения. Данные квантили упакуем в режиме оффлайн в как можно меньшее количество контейнеров единичного размера. После чего, аналогично алгоритму увеличения размерности в случае равномерного распределения длин сторон параллелепипедов, в каждую из получившихся областей упаковываем параллелепипеды по оставшимся  $d - 1$  сторонам, используя алгоритм для упаковки объектов меньшей на 1 размерности.

В следующем разделе приведены результаты вычислительных экспериментов, позволяющие сравнить алгоритмы увеличения размерности от  $BF$  с алгоритмом из работы [17] для многомерной задачи упаковки в контейнеры. Также, на примере двумерной задачи, покажем, что предложенный алгоритм увеличения размерности позволяет эффективно упаковывать в контейнеры объекты параллелепипеды с неравномерным распределением длин сторон.

## 4. Результаты вычислительных экспериментов

В приложении 1 при доказательстве теоремы 1, были выделены 3 типа незаполненного пространства контейнеров. При росте числа упаковываемых параллелепипедов  $n \in \{1, \dots, N\}$ , математическое ожидание объема незаполненного пространства контейнеров каждого из 3 типов не убывает. Поэтому, при численном моделировании, достаточно вместо  $W_{d,N} = \max_{1 \leq n \leq N} \mathbb{E}W_{d,N}(n)$ , вычислять  $\mathbb{E}W_{d,N}(N)$ .

Проведем численные эксперименты.

Вычислим математическое ожидание  $W_{2,N}(N)$  для алгоритма двумерной упаковки  $A_{2,N} = A_{2,N}(BF)$  и для алгоритма трехмерной упаковки  $A_{3,N} = A_{3,N}(A_{2,N_2(N)}(BF))$ , полученных применением алгоритма увеличения размерности к алгоритму  $BF$  для одномерной упаковки, и сравним результаты с  $\mathbb{E}W_{d,N}(N)$  для алгоритмов *hash packing* из [17] соответствующей размерности  $d \in \{2,3\}$ . Также сравним адаптивную версию алгоритма увеличения размерности с алгоритмом *hash packing* при упаковке двумерных параллелепипедов, имеющих гауссовское распределение длин сторон.

### 4.1 Анализ алгоритма увеличения размерности для двумерной задачи

Проведем численный анализ алгоритма  $a.u.p.(BF)$ , полученного в результате однократного применения алгоритма увеличения размерности к алгоритму  $BF$  для одномерной задачи упаковки в контейнеры. Сравним плотность заполнения контейнеров после применения алгоритма увеличения размерности от  $BF$  и после применения алгоритма *hash packing*.

Из теоретических результатов следует, что асимптотика  $\mathbb{E}W_{2,N}^{HP}(N)$  алгоритма *hash packing* для двумерной задачи составляет  $\mathbb{E}W_{2,N}^{HP}(N) = \theta(N^{\frac{3}{4}})$ , а асимптотика алгоритма увеличения размерности от  $BF$  составляет  $\mathbb{E}W_{2,N}^{a.u.p.(BF)}(N) = O(N^{\frac{2}{3}\sqrt{\ln N}})$ . Таким образом,  $\mathbb{E}W_{2,N}^{HP}(N) / \mathbb{E}W_{2,N}^{a.u.p.(BF)}(N) = \Omega\left(\frac{1}{\sqrt{\ln N}}\right)$ , причем  $\frac{1}{\sqrt{\ln N}} = 0.85$  при  $N = 10^6$ .

В результате вычислительных экспериментов установлено (рис. 2), что при числе параллелепипедов  $N$  из отрезка  $[50, 10^6]$ , величина  $\frac{\mathbb{E}W_{2,N}^{HP}(N)}{\mathbb{E}W_{2,N}^{a.u.p.(BF)}(N)}$  изменяется от 1.60 при  $N = 50$  до 1.95 при  $N = 10^6$ , причем при  $N = 10^4$ , данная величина составляет 1.69. Для всех  $N$ ,  $a.u.p.(BF)$  позволяет получить меньшее математическое ожидание объема незаполненного пространства, чем алгоритм *hash packing*.

### 4.2 Анализ алгоритма увеличения размерности для трехмерной задачи

Сравним алгоритм увеличения размерности от  $BF$  и алгоритм *hash packing* для упаковки параллелепипедов в трехмерные контейнеры.

Из теоретических результатов следует, что асимптотика  $\mathbb{E}W_{3,N}^{HP}(N)$  алгоритма *hash packing* для трехмерной задачи составляет  $\mathbb{E}W_{3,N}^{HP}(N) = \theta(N^{\frac{4}{3}})$ , а асимптотика алгоритма увеличения

размерности от *BF* составляет  $\mathbb{E}W_{3,N}^{a.y.p.(BF)}(N) = O\left(N^{\frac{3}{4}} \ln^{\frac{3}{8}} N\right)$ . Таким образом,  $\mathbb{E}W_{3,N}^{HP}(N) / \mathbb{E}W_{3,N}^{a.y.p.(BF)}(N) = \Omega\left(N^{\frac{1}{20}} \ln^{-\frac{3}{8}} N\right)$ , причем  $N^{1/20} \ln^{-3/8} N = 0.75$  при  $N = 10^6$ .

В результате вычислительных экспериментов установлено (рис. 3), что при числе параллелепипедов  $N$  из отрезка  $[50, 10^6]$ , величина  $\frac{\mathbb{E}W_{3,N}^{HP}(N)}{\mathbb{E}W_{3,N}^{a.y.p.(BF)}(N)}$  изменяется от 2.81 при  $N = 50$  до 2.55 при  $N = 10^6$ , причем при  $N = 10^4$ , данная величина составляет 2.47.

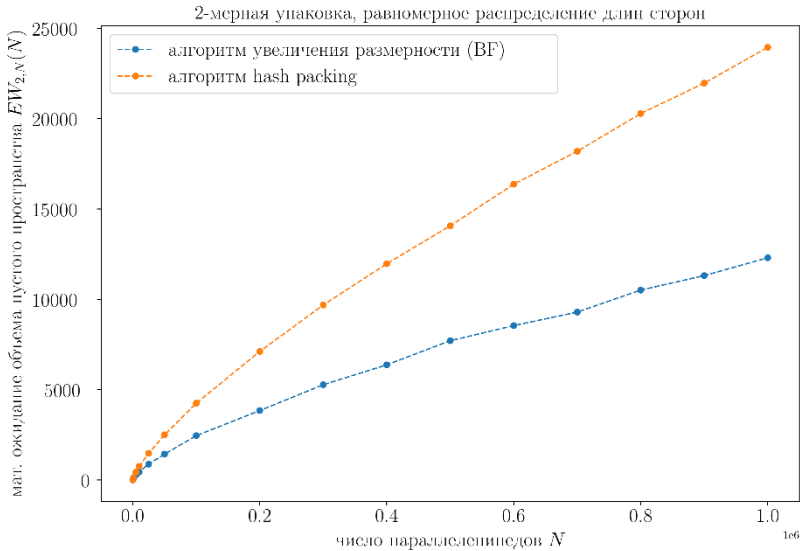


Рис. 2. Сравнение а.у.р.(BF) и hash packing для двумерной задачи упаковки в контейнеры.

Fig. 2. Comparison of the algorithm of increasing dimension from best fit to hash packing algorithm in the case of two-dimensional bin packing problem.

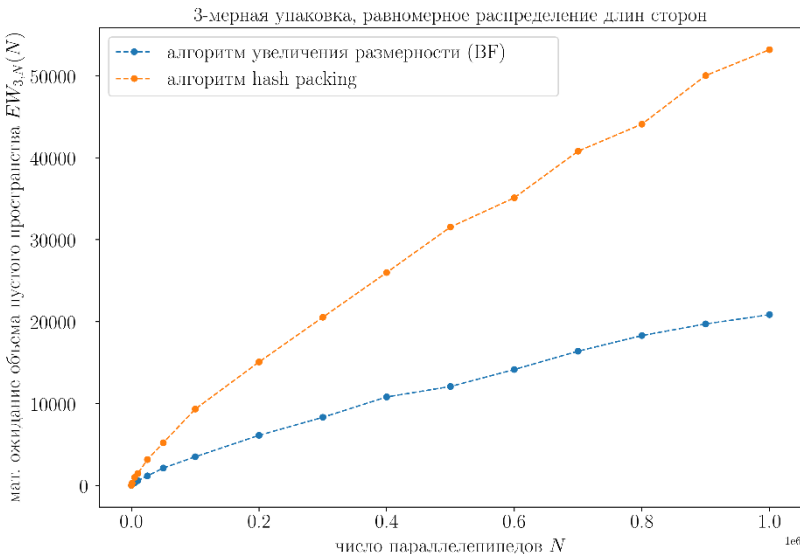


Рис. 3. Сравнение а.у.р.(BF) и hash packing для трехмерной задачи упаковки в контейнеры.

Fig. 3. Comparison of the algorithm of increasing dimension from best fit to hash packing algorithm in the case of three-dimensional bin packing problem.

### 4.3 Анализ алгоритма увеличения размерности в случае неравномерного распределения длин сторон параллелепипедов

При анализе алгоритмов упаковки в среднем случае, делается предположение о том, что входные данные имеют равномерное распределение. Это может вызывать “переобучение” алгоритма, когда малый объем незаполненного пространства достигается лишь в случае фиксированного распределения, или семейства распределений длин сторон параллелепипедов. Это, в свою очередь, может ограничивать практическое применение алгоритмов упаковки, показывающих высокую плотность упаковки при анализе в среднем.

Проведем численные эксперименты для двумерной задачи упаковки в контейнеры, в которых параллелепипеды имеют гауссовское распределение длин сторон. Сравним алгоритм увеличения размерности и *hash packing* в этом случае.

В результате вычислительных экспериментов установлено (рис. 4), что при числе параллелепипедов  $N$  из отрезка  $[50, 10^6]$ , величина  $\frac{EW_{2,N}^{HP}(N)}{EW_{2,N}^{a.u.p.(BF)}(N)}$  изменяется от 2.56 при  $N = 50$  до 21.5 при  $N = 10^6$ , причем при  $N = 10^4$ , данная величина составляет 11.8. Это показывает, что *a.u.p.(BF)* был эффективно обобщен на случай неравномерного распределения длин сторон параллелепипедов, и в этом случае *a.u.p.(BF)* на порядок превосходит алгоритм *hash packing* по плотности упаковки.

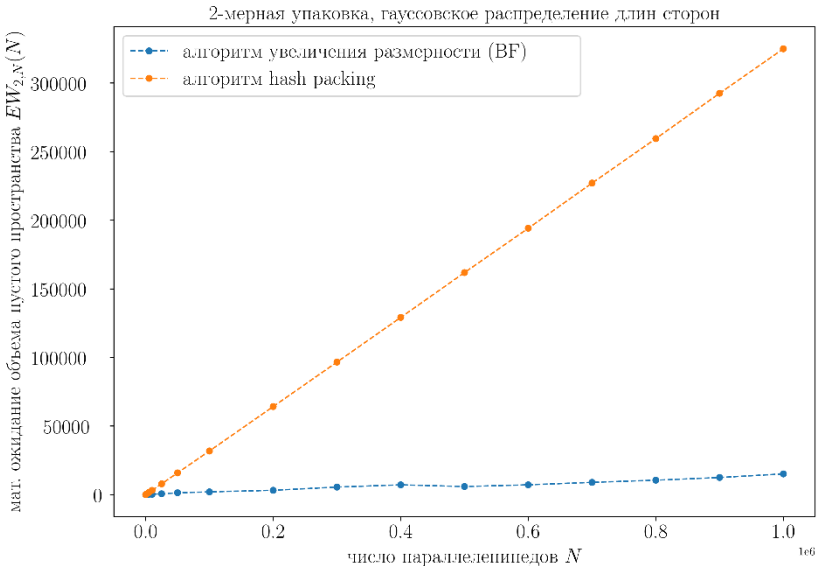


Рис. 4. Сравнение *a.u.p.(BF)* и *hash packing* в случае неравномерного распределения длин сторон параллелепипедов.

Fig. 4. Comparison of the algorithm of increasing dimension from best fit to hash packing algorithm in the case of non-uniform object sizes distribution.

### 5. Заключение

Была рассмотрена задача многомерной упаковки параллелепипедов в контейнеры. Лучшая из известных асимптотических оценок математического ожидания  $d$ -мерного объема не заполненной параллелепипедами части частично заполненных контейнеров при анализе в среднем для алгоритмов, работающих в режиме онлайн с незадаанным завершением для  $d$ -мерной задачи, улучшена с  $\theta\left(N^{\frac{d+1}{d+2}}\right)$  до  $O\left(N^{\frac{d}{d+1}} \ln^{\frac{3}{2(d+1)}} N\right)$ .

Численные эксперименты подтверждают теоретический результат. Показано, что предложенный алгоритм, названный алгоритмом увеличения от алгоритма *BF* размерности  $d$ , имеет меньшее математическое ожидание незаполненного объема контейнеров, чем лучший ранее известный алгоритм *hash packing* для размерностей 2 и 3 при числе испытаний от 50 до  $10^6$ . Также показано, что предложенный алгоритм увеличения размерности может быть эффективно обобщен со случая равномерного распределения длин сторон параллелепипедов на случай общего распределения, позволяя получить меньшее математическое ожидание объема незаполненного пространства, чем алгоритм *hash packing*.

## Список литературы / References

- [1]. Johnson, David S., et al. Worst-case performance bounds for simple one-dimensional packing algorithms. *SIAM Journal on computing* 3.4, 1974, pp. 299-325. DOI: 10.1137/0203025.
- [2]. Coffman Jr, Edward G., Michael R. Garey, and David S. Johnson. *Approximation algorithms for bin-packing—an updated survey. Algorithm design for computer system design.* Springer Vienna, 1984, pp. 49-106. DOI: 10.1007/978-3-7091-4338-4.
- [3]. Li, Xueping, and Kaik Zhang. Single batch processing machine scheduling with two-dimensional bin packing constraints. *International Journal of Production Economics* 196, 2018, pp. 113-121. DOI: 10.1016/j.ijpe.2017.11.015.
- [4]. Paquay, Célia, Michael Schyns, and Sabine Limbourg. A mixed integer programming formulation for the three-dimensional bin packing problem deriving from an air cargo application. *International Transactions in Operational Research* 23.1-2, 2016, pp. 187-213. DOI: 10.1111/itor.12111.
- [5]. Tchernykh, Andrei, et al. On-line hierarchical job scheduling on grids with admissible allocation. *Journal of Scheduling* 13.5, 2010, pp. 545-552. DOI: 10.1007/s10951-010-0169-x.
- [6]. Tchernykh, Andrei, et al. Two level job-scheduling strategies for a computational grid. *International Conference on Parallel Processing and Applied Mathematics.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 774-781. DOI: 10.1007/11752578\_93.
- [7]. Gohil, Bhavesh, et al. A comparative analysis of virtual machine placement techniques in the cloud environment. *International Journal of Computer Applications* 156.14, 2016, pp. 12-18. DOI: 0.5120/ijca2016912530.
- [8]. Garey, Michael R., and David S. Johnson. *Computers and intractability.* Vol. 29. New York: wh freeman, 2002. 338 pp.
- [9]. Brown, Donna J. A Lower Bound for On-Line One-Dimensional Bin Packing Algorithms. No. ACT19. 1979, pp. 1-21. DOI: 10.1016/S0020-0190(80)90077-0.
- [10]. van Vliet, André. An improved lower bound for on-line bin packing algorithms. *Information processing letters* 43.5, 1992, pp. 277-284. DOI: 10.1016/0020-0190(92)90223-1.
- [11]. Galambos, Gabor, and André van Vliet. Lower bounds for 1-, 2-and 3-dimensional on-line bin packing algorithms. *Computing* 52.3, 1994, pp. 281-297. DOI: 10.1007/BF02246509.
- [12]. Han, Xin, et al. A new upper bound 2.5545 on 2d online bin packing. *ACM Transactions on Algorithms (TALG)* 7.4, 2011, pp. 1-18. DOI: 10.1145/2000807.2000818.
- [13]. Csirik, János, and André Van Vliet. An on-line algorithm for multidimensional bin packing. *Operations Research Letters* 13.3, 1993, pp. 149-158. DOI: 10.1016/0167-6377(93)90004-Z.
- [14]. Shor, Peter W. The average-case analysis of some on-line algorithms for bin packing. *Combinatorica* 6.2, 1986, pp. 179-200. DOI: 10.1007/BF02579171.
- [15]. Shor, Peter W. How to pack better than best fit: tight bounds for average-case online bin packing. *Proceedings 32nd Annual Symposium of Foundations of Computer Science, IEEE Computer Society,* 1991, pp. 752-759, DOI: 10.1109/SFCS.1991.185444.
- [16]. Leighton, Frank Thomson, and Peter Shor. Tight bounds for minimax grid matching, with applications to the average case analysis of algorithms. *Proceedings of the eighteenth Annual ACM symposium on theory of computing,* 1986, pp. 91-103, DOI: 10.1145/12130.12140.
- [17]. Chang, Ee-Chien, Weiguo Wang, and Mohan S. Kankanhalli. Multidimensional on-line bin-packing: An algorithm and its average-case analysis. *Information Processing Letters* 48.3, 1993, pp. 121-125. DOI: 10.1016/0020-0190(93)90253-6.

[18]. Chung, Fan, and Linyuan Lu. Connected components in random graphs with given expected degree sequences. Annals of combinatorics 6.2, 2002, pp. 125-145. DOI: 10.1007/PL00012580.

## Приложение 1. Доказательство теоремы 1

Докажем теорему 1. Для этого, приведем и докажем несколько лемм.

В работе [18] было доказано следующее ограничение сверху отклонения суммы биномиальных случайных величин от ее математического ожидания:

**Лемма 1 (Chung, Lu, 2002).** Пусть  $X_1, \dots, X_n$  – независимые в совокупности случайные величины, имеющие распределение Бернулли. Принимаемые значения принадлежат множеству  $\{0, 1\}$ ,  $\Pr(X_i = 1) = p_i$ , а  $\Pr(X_i = 0) = 1 - p_i$ . Выберем значения  $\alpha_i \geq 0 \forall i \in \{1, 2, \dots, n\}$ . Определим значения  $X = \sum_{i=1}^n \alpha_i X_i$  и  $v = \sum_{i=1}^n \alpha_i^2 p_i$ . Тогда для каждого  $\lambda \geq 0$ , выполняется следующее неравенство:

$$\Pr(X \geq \mathbb{E}X + \lambda) \leq \exp \left\{ - \frac{\lambda^2}{2 \left( v + \frac{\lambda \max_i \alpha_i}{3} \right)} \right\}$$

Для доказательства теоремы 1, требуется следующее следствие леммы 1:

**Следствие 1.** Для каждого  $\varepsilon \in [0, 1]$ , для случайной биномиально распределенной величины  $\text{Bin}(n, p)$ , выполняется

$$\Pr\{\text{Bin}(n, p) \geq (1 + \varepsilon)np\} \leq \exp \left\{ - \frac{\varepsilon^2 np}{3} \right\}$$

**Доказательство.** Биномиальная случайная величина  $\text{Bin}(n, p)$  представима в виде суммы независимых в совокупности случайных величин, имеющих распределение Бернулли:  $\text{Bin}(n, p) = \sum_{i=1}^n X_i$ , где  $\Pr(X_i = 1) = p$ ,  $\Pr(X_i = 0) = 1 - p$  для всех  $i \in \{1, \dots, n\}$ .

Используя лемму 1 при значениях параметров  $\alpha_i = 1, p_i = p \forall i \in \{1, \dots, n\}$ , и при значении  $X = \text{Bin}(n, p)$ ,  $\mathbb{E}X = np, \lambda = \varepsilon np, v = np$ , получим

$$\Pr\{\text{Bin}(n, p) \geq (1 + \varepsilon)np\} \leq \exp \left\{ - \frac{(\varepsilon np)^2}{2 \left( np + \frac{\varepsilon np}{3} \right)} \right\} \leq \exp \left\{ - \frac{\varepsilon^2 np}{3} \right\} \quad \square$$

**Лемма 2.** Пусть  $X$  – случайная величина,  $\mathbb{E}X$  – ее математическое ожидание и  $\text{Var}(X)$  – ее дисперсия. Тогда

$$\mathbb{E}^2 |X - \mathbb{E}X| \leq \text{Var}(X)$$

**Доказательство.**  $\text{Var}(|X - \mathbb{E}X|) = \mathbb{E}(X - \mathbb{E}X)^2 - \mathbb{E}^2 |X - \mathbb{E}X| \geq 0$ .

Следовательно,  $\mathbb{E}^2 |X - \mathbb{E}X| \leq \mathbb{E}(X - \mathbb{E}X)^2 = \text{Var}(X) \quad \square$

**Лемма 3.** Обозначим за  $V_i(k)$  сумму  $d$ -мерных объемов проекций  $d + 1$ -мерных параллелепипедов на подпространство, образованное последними  $d$  координатами тех из  $k$  случайных параллелепипедов, которые имеют тип  $i$  для фиксированного типа  $i \in \{1, \dots, 2s + 1\}$  (см. алгоритм увеличения размерности). Тогда верно следующее неравенство:

$$\text{Var}(V_i(k)) \leq 2pk,$$

где  $p = \frac{1}{2s+1}$  – вероятность того, что случайный параллелепипед имеет тип  $i$ .

**Доказательство.**  $V_i(1) = \text{Bin}(1, p) \prod_{i=1}^d \xi_i = \text{Bin}(1, p)U_d$ , где  $\text{Bin}(n, p)$  – случайная величина, имеющая биномиальное распределение с параметрами  $n$  и  $p$ , равными числу экспериментов и вероятности успешности каждого эксперимента соответственно,  $\xi_i$  – случайная величина, имеющая равномерное на  $[0, 1]$  распределение,  $U_d = \prod_{i=1}^d \xi_i$  – случайная величина, равная объему  $d$ -мерного параллелепипеда.

Так как  $\mathbb{E}Bin(1, p) = p$ ,  $0 \leq U_d \leq 1$ , и длины сторон параллелепипедов – независимые в совокупности случайные величины, то

$$\mathbb{E}V_i(1) = \mathbb{E}Bin(1, p) \mathbb{E}U_d \leq p$$

Используя формулу для дисперсии произведения двух независимых случайных величин,  $Var(XY) = Var(X)Var(Y) + Var(X)\mathbb{E}^2Y + Var(Y)\mathbb{E}^2X$ , и соотношения  $Var(Bin(1, p)) = p(1 - p)$ ,  $\mathbb{E}Bin(1, p) = p$  и  $Var(U_d) \leq 1, 0 \leq \mathbb{E}U_d \leq 1$ , получим

$$Var(V_i(1)) = Var(Bin(1, p)U_d) = Var(Bin(1, p))Var(U_d) + Var(Bin(1, p))\mathbb{E}^2U_d + Var(U_d)\mathbb{E}^2(Bin(1, p)) \leq p(1 - p) + p(1 - p) + p^2 \leq 2p$$

В силу независимости в совокупности длин сторон параллелепипедов, получим требуемое неравенство:

$$Var(V_i(k)) = kVar(V_i(1)) \leq 2pk \quad \square$$

**Теорема 1.** Пусть  $A_{d,M}, \forall M \in \mathbb{N}$  – семейство алгоритмов с незадаанным завершением для  $d$ -мерной задачи с оценкой целевой функции типа 2, равной

$$W_{d,M}(A_{d,M}) = O\left(M^{\frac{d}{d+1}} \ln^m M\right), m \geq 0$$

Тогда алгоритм  $A_{d+1,N} = A_{d+1,N}(A_{d,N_d(N)})$  для  $d + 1$ -мерной задачи, полученный в результате применения алгоритма увеличения размерности к алгоритму  $A_{d,N_d(N)}$ , где  $N_d(N)$  – параметр, введенный в разделе 3.1. настоящей работы в алгоритме увеличения размерности. Алгоритм  $A_{d+1,N}$  работает в режиме с незадаанным завершением и имеет оценку целевой функции типа 2

$$W_{d+1,N}(A_{d+1,N}) = O\left(N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N\right).$$

Введем обозначения в предположении, что выпало  $k \leq N$  параллелепипедов:

- $w_i = w_i(k)$  –  $d$ -мерный объем незаполненного пространства контейнеров при упаковке параллелепипедов типа  $i$  алгоритмом  $A_{d,N_d(N)}$  в  $d$ -мерные контейнеры,
- $n_i = n_i(k)$  – число  $d$ -мерных контейнеров, заполненных проекциями параллелепипедов типа  $i$  на последние  $d$  координат при размещении проекций с помощью алгоритма  $A_{d,N_d(N)}$ ,
- $m_i = m_i(k)$  – число выпавших параллелепипедов типа  $i$ ,
- $W_{d+1,N}(k)$  –  $d + 1$ -мерный объем незаполненной части контейнеров после упаковки  $k$  параллелепипедов алгоритмом  $A_{d+1,N}$ ,
- $K_k$  – количество  $d + 1$ -мерных контейнеров, использованных алгоритмом  $A_{d+1,N}$  для упаковки  $k$  параллелепипедов,
- $V_i(k)$  – сумма  $d$ -мерных объемов проекций на подпространство, образованное последними  $d$  координатами тех из  $k$  случайных параллелепипедов, которые имеют тип  $i$  для некоторого фиксированного типа  $i \in \{1, \dots, 2s + 1\}$ .

**Доказательство.**

$$W_{d+1,N}(k) \leq \frac{2K_k}{2s + 1} + \sum_{i=1}^{2s+1} w_i(k) + \sum_{i=1}^s |n_i(k) - n_{2s+1-i}(k)| \quad (1)$$

В формуле (1), первое слагаемое  $\frac{2K_k}{2s+1}$  (потери типа 1), ограничивает сверху объем незаполненной части частично заполненных  $d + 1$ -мерных контейнеров, связанный с тем, что у  $d + 1$ -мерного параллелепипеда  $R$  типа  $i$ , длина  $b_1(R)$  первой стороны принадлежит полуинтервалу  $\left(\frac{i-1}{2s+1}, \frac{i}{2s+1}\right]$  длиной  $\frac{1}{2s+1}$ .

Также из определения  $K_k$ , следует следующее неравенство:

$$K_k \leq k \quad (2)$$

Множитель 2 в первом слагаемом формулы (1) возникает, т.к. каждый контейнер разбит не более, чем на 2 области (см. рис. 1).

Слагаемое  $\sum_{i=1}^{2s+1} w_i$  (потери типа 2) – ограничивает сверху суммарный объем незаполненного пространства контейнеров, заполненных параллелепипедами типа  $i$  без учета потерь типа 1.

Слагаемое  $\sum_{i=1}^s |n_i - n_{2s+1-i}|$  (потери типа 3) – ограничивает сверху объем частей контейнеров, заполненных параллелепипедами типа  $i$ , но не заполненных параллелепипедами типа  $2s + 1 - i$ ,  $i \in \{1, \dots, s\}$ .

$$\begin{aligned} \sum_{i=1}^s |n_i - n_{2s+1-i}| &= \sum_{i=1}^s |(V_i(k) + w_i) - (V_{2s+1-i}(k) + w_{2s+1-i})| \leq \\ &\sum_{i=1}^s |V_i(k) - V_{2s+1-i}(k)| + \sum_{i=1}^s |w_i - w_{2s+1-i}| \leq \\ \sum_{i=1}^s |V_i(k) - V_{2s+1-i}(k)| + \sum_{i=1}^s (w_i + w_{2s+1-i}) &= \sum_{i=1}^s |V_i(k) - V_{2s+1-i}(k)| + \sum_{i=1}^{2s} w_i \quad (3) \end{aligned}$$

Примеры различных типов потерь для пары соседних областей в 2-мерном случае изображены на рис. 5.

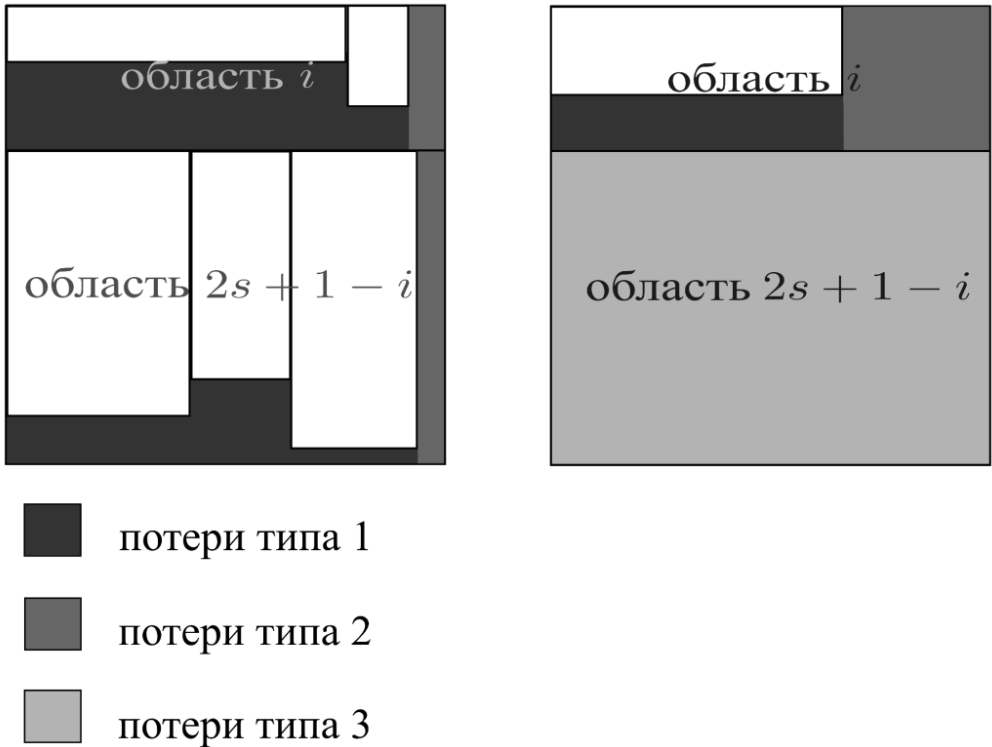


Рис.5. Пример различных типов незаполненного пространства контейнеров.  
Fig.5. Example of different types of missing space of bins.

Из формулы (1) с применением формул (2) и (3), получаем:

$$W_{d+1,N}(k) \leq \frac{2K_k}{2s+1} + 2 \sum_{i=1}^{2s+1} w_i + \sum_{i=1}^s |V_i - V_{2s+1-i}|$$

Следовательно,

$$\mathbb{E}W_{d+1,N}(k) \leq \frac{2K_k}{2s+1} + 2\mathbb{E}\left(\sum_{i=1}^{2s+1} w_i\right) + \mathbb{E}\left(\sum_{i=1}^s |V_i - V_{2s+1-i}|\right) \quad (4)$$

1. Оценим первое слагаемое формулы 4, учитывая, что  $s = \lfloor N^{1/(d+2)} \ln^{-m(d+1)/(d+2)} N \rfloor$ :

$$\frac{2K_k}{2s+1} \leq \frac{2k}{2s+1} \leq \frac{2N}{2s+1} = O\left(N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N\right)$$

2. Оценим второе слагаемое: При достаточно больших  $N$ ,  $4\sqrt{\ln NN}^{-d/2(d+2)} \leq 1$ . Таким образом, согласно следствию 1 леммы 1,

$$\begin{aligned} Pr\left(\text{Bin}\left(k, \frac{1}{2s+1}\right) \geq \left(1 + 4\sqrt{\ln NN}^{-\frac{d}{2(d+2)}}\right) \frac{N}{2s+1}\right) &\leq \\ Pr\left(\text{Bin}\left(N, \frac{1}{2s+1}\right) \geq \left(1 + 4\sqrt{\ln NN}^{-\frac{d}{2(d+2)}}\right) \frac{N}{2s+1}\right) &\leq \\ \exp\left\{-\frac{16}{3} N^{-\frac{d}{d+2}} \frac{N}{2s+1} \ln N\right\} &\leq N^{-5} \end{aligned}$$

для достаточно больших  $N$ . Следовательно, вклад в мат. ожидание  $\mathbb{E}W_{d,N}(k)$  тех случаев, когда хотя бы одного типа  $i$ , число выпавших параллелепипедов  $m_i$  этого типа больше, чем  $\left(1 + 4\sqrt{\ln NN}^{-\frac{d}{2(d+2)}}\right) \frac{N}{2s+1}$ , равен  $o(1)$ . При достаточно больших  $N$ , выполняется  $4\sqrt{\ln NN}^{-d/2(d+2)} \leq 1$ . Таким образом,

$$\begin{aligned} \mathbb{E}\left(\sum_{i=1}^{2s+1} w_i\right) &\leq (2s+1) \max_{1 \leq k \leq \frac{2N}{2s+1}} \mathbb{E}w_i(k) + o(1) = \\ (2s+1)O\left(\left(\frac{2N}{2s+1}\right)^{\frac{d}{d+1}} \ln^m N\right) + o(1) &= O\left(N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N\right) \end{aligned}$$

в предположении, что для алгоритма  $A_{d,N_d(N)}$ , количество упаковываемых параллелепипедов  $N_d(N) = \left(1 + 4\sqrt{\ln NN}^{-\frac{d}{2(d+2)}}\right) \frac{N}{2s+1}$ , не превышает  $2N/(2s+1)$ .

3. Оценим третье слагаемое формулы (4). Так как  $i \in \{1, \dots, 2s+1\}$ ,  $\mathbb{E}V_i(k) = \frac{k}{2s+1}$ ,

$$\begin{aligned} \mathbb{E}|V_i(k) - V_{2s+1-i}(k)| &= \mathbb{E}|(V_i(k) - \mathbb{E}V_i(k)) - (V_{2s+1-i}(k) - \mathbb{E}V_{2s+1-i}(k))| \leq \\ &\mathbb{E}|V_i(k) - \mathbb{E}V_i(k)| + \mathbb{E}|V_{2s+1-i}(k) - \mathbb{E}V_{2s+1-i}(k)| \end{aligned}$$

Таким образом,

$$\begin{aligned} \mathbb{E}\left(\sum_{i=1}^s |V_i(k) - V_{2s+1-i}(k)|\right) &\leq \sum_{i=1}^{2s} \mathbb{E}|V_i(k) - \mathbb{E}V_i(k)| \leq (\text{по лемме 2}) \\ 2s\sqrt{\text{Var } V_i(k)} &\leq (\text{по лемме 3}) \leq 2s\sqrt{\frac{k}{s}} \leq 2s\sqrt{\frac{N}{s}} = O\left(N^{\frac{d+1}{d+2}}\right) \end{aligned}$$

Значит, каждое слагаемое имеет асимптотическую оценку  $O\left(N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N\right)$ .

Следовательно, выполняется искомое соотношение

$$\max_{1 \leq k \leq N} W_{d+1,N}(k) = O\left(N^{\frac{d+1}{d+2}} \ln^{\frac{m(d+1)}{d+2}} N\right) \quad \square$$

### ***Информация об авторах / Information about authors***

Денис Олегович ЛАЗАРЕВ является специалистом отдела теоретической информатики Института системного программирования им. В.П. Иванникова РАН. Научные интересы включают машинное обучение, вероятностный метод и алгоритмы упаковки.

Denis Olegovich LAZAREV – specialist of the Department of Theoretical Computer Science of Ivannikov Institute for System Programming of the RAS. Research interests include machine learning, probabilistic methods and packing algorithms.

