アルゴリズム 80－7
（2001．9．25）

最悪性能比が 2.7834 二次元調和算法の提案と評価

HAN Xin[1,2]，藤田 聡[1]，郭 禾[2]
1. 広島大学　2. 大連理工大学

アブストラクト：箱詰め問題は 組み合わせ最適化問題における基本的な問題の１つである。60,70 年代から、ずっと 注目され、たくさんの結果が出た本論では、二次元の箱詰め問題について考察する。以下では 二次元調和算法の改良版である RTDH を 提案し、その性能を理論的に評価する。評価の結果、提案アルゴリズムの最悪性能比が 2.7834 以下であることが示される。

A Two-Dimensional Harmonic Algorithm with Performance Ratio 2.7834

Xin HAN[1,2], Satoshi FUJITA[1], He GUO[2]
1. Hiroshima University, Japan
2. Dalian University of Technology, The Peoples Republic of China

In this paper, we study an on-line version of the two-dimensional bin packing problem that is the problem of packing a list of rectangular items into a minimum number of unit-square bins in an on-line manner. An on-line algorithm RTDH (Refined Two Dimensional HARMONIC) is proposed and analyzed. We show that RTDH can achieve an asymptotic worst case ratio of less than 2.7834, that beats the best known bound 2.85958.

1 introduction

The bin packing problem is one of the basic problems in the fields of theoretical computer science and combinatorial optimization. It has many important real-world applications, such as memory allocation and job scheduling, and is well-known to be NP-hard [3]; that is a main motivation of the study and development of approximation algorithms for solving the problem.

The classical (one-dimensional) on-line bin packing problem is the problem of, given a list L of items $\langle x_1, x_2, \ldots, x_n \rangle$ where $x_i \in (0, 1]$, packing all items in L into a minimum number of unit-capacity bins. Note that term "on-line" implies that items in L are consecutively input, and the packing of an item must be determined before the arrival of the next item.

In this paper, we consider a two-dimensional version of the problem, that is the problem of packing all rectangular items in L into a minimum number of unit-square bins in such a way that 1) each item is entirely contained inside its bin with all sides parallel to the sides of the bin, 2) no two items in a bin overlap, and 3) the orientation of any item is the same as the orientation of the bin. In the literature, it is known that the worst case ratio[1] of an optimal on-line bin packing algorithm OPT, denoted by R^{∞}_{OPT}, which fulfills $1.907 \leq R^{\infty}_{OPT} \leq 2.85958$ $(= 1.69103^2)$ [1, 2].

In this paper, we propose an on-line algorithm that beats the above upper bound 2.85958; more precisely, our algorithm achieves the worst case ratio of 2.7834. The basic idea of the algorithm is to pack several big items into the same bin as much as possible. Our analysis of the worst case ratio is based on the notion of maximum size of the area that should be spent for each item and an exhaustive case enumerations. The remainder of this paper is organized as follows. Section 2 introduces some basic definitions. The classifications of items and strips used in the proposed algorithm will also be given. Section 3 describes the proposed algorithm, and Section 4 is devoted to the analysis of the worst case ratio. Finally, Section 5 concludes the paper with some future directions of research.

[1] A formal definition will be given in Section 2.

2 Preliminaries

2.1 Basic Definitions

Let $\mathcal{A}$ be an on-line bin packing algorithm, and $\mathcal{A}(L)$ the number of bins used by algorithm $\mathcal{A}$ for input sequence L. The *asymptotic worst case ratio* (or simply, *worst case ratio*) of algorithm $\mathcal{A}$, denoted as $R^{\infty}_{\mathcal{A}}$, is defined as follows:

$$R^{\infty}_{\mathcal{A}} \stackrel{\text{def}}{=} \limsup_{n\to\infty} R^{n}_{\mathcal{A}}$$
$$\text{where } R^{n}_{\mathcal{A}} \stackrel{\text{def}}{=} \max\left\{\frac{\mathcal{A}(L)}{OPT(L)} \mid OPT(L) = n\right\}.$$

where $OPT(L)$ denotes the number of bins used by an optimal bin packing problem provided that the input is L.

2.2 Classification

Let M be an integer that is greater than or equal to 6. As will be shown later, we will fix M to 6, as in [4]. Let T be the set of items that can be packed into a unit-square bin; i.e., $T \stackrel{\text{def}}{=} (0,1] \times (0,1]$. In the proposed algorithm, set T is partitioned into several subsets, as in Harmonic [4] and ROUND [2].

First, T is partitioned into four subsets A, B, C, D as

$$\begin{aligned} A &= \{(x,y) \mid 0 < x \le 1/M \text{ and } 0 < y \le 1/M\} \\ B &= \{(x,y) \mid 1/M < x \le 1 \text{ and } 0 < y \le 1/M\} \\ C &= \{(x,y) \mid 0 < x \le 1/M \text{ and } 1/M < y \le 1\} \text{ and} \\ D &= \{(x,y) \mid 1/M < x \le 1 \text{ and } 1/M < y \le 1\}. \end{aligned}$$

Next, subsets B, C and D are partitioned into smaller subsets as follows:

$$\begin{aligned} B_i &= \{(x,y) \in B \mid 1/(i+1) < x \le 1/i\} \\ C_j &= \{(x,y) \in C \mid 1/(j+1) < y \le 1/j\} \quad \text{and} \\ D_{i,j} &= \{(x,y) \mid 1/(i+1) < x \le 1/i \\ &\qquad \text{and } 1/(j+1) < y \le 1/j\} \end{aligned}$$

for $1 \le i, j < M$, and in addition, subsets $D_{1,1}$, $D_{1,2}$ and $D_{2,1}$ are further partitioned into smaller subsets as follows (see Figure 1 for illustration):

$$\begin{aligned} D^1_{1,1} &= \{(x,y) \mid 1/2 < x \le 3/5, 1/2 < y \le 3/5\} \\ D^0_{1,1} &= D_{1,1} - D^1_{1,1} \\ D^1_{1,2} &= \{(x,y) \mid 1/2 < x \le 3/5, 1/3 < y \le 2/5\} \\ D^0_{1,2} &= D_{1,2} - D^1_{1,2} \\ D^1_{2,1} &= \{(x,y) \mid 1/3 < x \le 2/5, 1/2 < y \le 3/5\} \\ D^0_{2,1} &= D_{2,1} - D^1_{2,1}. \end{aligned}$$

For brevity of the notation, in the following, we denote subsets $D^1_{1,1}, D^1_{1,2}, D^1_{2,1}$ by α, β, and γ, respectively.

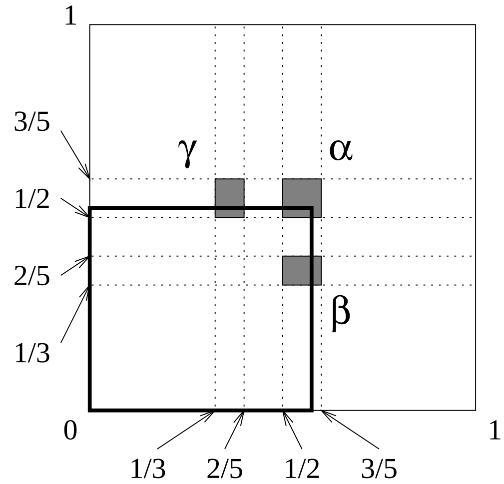

Figure 1: Subsets α, β, and γ (an item drawn by a thick line is an α-item).

Let $\mathcal{S}$ be the set of all subsets defined above; i.e.,

$$\mathcal{S} \stackrel{\text{def}}{=} \{A, B_1, \ldots, B_{M-1}, C_1, \ldots, C_{M-1}, \alpha, \beta, \gamma, D^0_{1,1}, D^0_{1,2}, D^0_{2,1}, D_{2,2}, \ldots, D_{M-1,M-1}\}.$$

Note that the number of subsets in $\mathcal{S}$ is $M^2 + 3$ $(= 1 + (M-1) + (M-1) + (M-1)^2 + 3)$. In what follows, an item in subset ξ $(\in \mathcal{S})$ will be referred to as an ξ-item.

In the proposed algorithm, the set of unit-square bins is also classified into several categories according to the classification of items, and each bin is dedicated to a specific subset of items, except for subsets α, β, and γ; As will be explained below, those three subsets will be treated in a mixed manner (note that an α-item, a β-item, and a γ-item can be packed together into one bin). In the algorithm, a bin that can accommodate (merely) ξ-items will be referred to as a ξ-bin. As for the packing of α-, β-, and γ-items, on the other hand, we will distinguish nine independent cases shown in Table 1, where, e.g., (α, β) denotes the type of bin that accommodates one α-item and one β-item. Note that the type of a bin can change dynamically during the execution of the algorithm; e.g., when an empty bin accommodates an α-item as its first item, it becomes an (α)-bin, and when it accommodates a β-item as its second item, it becomes an (α, β)-item, and so on.

3 Algorithm Description

In the proposed algorithm RTDH_M (Refined Two Dimensional $\mathsf{HARMONIC}_M$) a bin becomes "active" when it accommodates its first item, and once it is declared to be "closed" it does not become active again. An outline of the algorithm is described as follows.

Table 1: Nine kinds of type of bins containing items in $\alpha \cup \beta \cup \gamma$.

type of bin	α	β	γ
(α)	1	0	0
(β)	0	1	0
(γ)	0	0	1
(α, β)	1	1	0
(β, β)	0	2	0
(α, γ)	1	0	1
(γ, γ)	0	0	2
(β, γ)	0	1	1
(α, β, γ)	1	1	1

Algorithm RTDH_M
Let $L = (a_1, a_2, \ldots, a_n)$ be the list of items that is given in an on-line manner. The packing of item a_k to an active bin is determined in the following manner:

Case 1: If $a_k \in A$, then pack it into an active A-bin by using a procedure shown in Subsection 3.1.

Case 2: When it accommodates its first item, each B_i-bin is split into strips of width $1/i$ and height 1, and each C_j-bin is split into strips of width 1 and height $1/j$. If $a_k \in B_i$ (resp. C_j), then pack the item into its corresponding strip of a B_i-bin (resp. C_j-bin) in a Next Fit manner (by regarding each strip as a bin).

Case 3: If $a_k \in D_{i,j} - (\alpha \cup \beta \cup \gamma)$, then pack the item into an active $D_{i,j}$-bin; if the bin becomes to contain $i \times j$ items, then after closing it, open a new bin as a new active $D_{i,j}$-bin.

Case 4: If $a_k \in \alpha \cup \beta \cup \gamma$, then pack it into an active bin by using a procedure shown in Subsection 3.2.

In the following two subsections, we will give a detailed description of packing procedures used in Cases 1 and 4.

3.1 Packing of A-items

The packing of A-items is done in a similar way to ROUND [2]. Let Y_1, Y_2, Y_3, Y_4 and Y_5 be five sets of reals in $(0, 1]$ defined as follows:

$$\begin{aligned} Y_1 &= \{1/2^i \mid i = 3, 4, \ldots\} \\ Y_2 &= \{1/(3 \times 2^i) \mid i = 1, 2, 3, \ldots\} \\ Y_3 &= \{1/(5 \times 2^i) \mid i = 1, 2, 3, \ldots\} \\ Y_4 &= \{1/(7 \times 2^i) \mid i = 0, 1, 2, 3, \ldots\} \text{ and} \\ Y_5 &= \{1/(9 \times 2^i) \mid i = 0, 1, 2, 3, \ldots\}. \end{aligned}$$

For real number $a \leq 1/M$, where $M \geq 6$, let $\bar{a}$ denote the smallest real in $\bigcup_{i=1}^{5} Y_i$ that is not smaller than a, and in what follows, we say that value a is "rounded up" to $\bar{a}$. For example, when $M = 6$, $9/60$ is rounded up to $1/6$ ($\in Y_2$) and $10/101$ is rounded up to $1/10$ ($\in Y_3$). In the following, an A-item (x, y) is said to be of "type-i" if $\bar{y} \in Y_i$ for $i = 1, 2, \ldots, 5$. (Note that $Y_i \cap Y_j = \emptyset$ for $i \neq j$.)

In algorithm RTDH_M, each A-bin is split into strips of width 1 and with a height that is drawn from set $\bigcup_{i=1}^{5} Y_i$. Those strips are used in such a way that items of different types cannot be packed into the same bin. A strip becomes active when it accommodates its first item, and once it is declared to be closed, it does not become active again. In the following, we show a concrete procedure for the packing of an A-item $a_k = (x, y)$ of type 1, i.e., $\bar{y} = 2^{-m}$ for some $m > 2$. The packing of items of the other types can be done in a similar manner.

Packing of A-item of type 1

Step 1: Find an active strip of height $\bar{y}$ that can accommodate the input item $a_k = (x, y)$. If there is no such active strips, then after closing all active strips of height $\bar{y}$ (if any), execute the following operations:

(1) If there exists an empty strip of height $\bar{y} = 2^{-m}$, then make it as a new active strip of height $\bar{y}$.

(2) Otherwise, after finding an empty strip of height $2^{-m'}$ such that $m' < m$ and m' is largest possible (it includes the case in which $m' = 0$), partition the strip into $m - m' + 1$ empty strips of height $2^{-(m'+1)}, 2^{-(m'+2)}, \ldots, 2^{-(m-1)}, 2^{-m}$ and 2^{-m}, respectively, then make a strip of height 2^{-m} as an active one.

Step 2: Pack the item into the active strip (of height $\bar{y}$) in a "bottom-left" manner.

The efficiency of the packing of A-items can be evaluated as follows. First, let us consider the final packing of type-1 items. Given a final packing, "repack" items in such a way that all the empty strips are moved to former bins, and all active strips are moved to latter bins. By the description of the algorithm, a single bin is enough for each purpose. Hence, by regarding those areas as a waste, we can conclude that the amount of "type-1 waste" is at most as large as two bins. A similar claim holds for waste of the other types. On the other hand, since each "non-waste" strip (of each type) is filled with items in such a way that the total width is at least $(M-1)/M$ (recall that the width of any A-item is smaller than $1/M$), and in addition, since our rounding strategy guarantees that every A-item is rounded up its height by no more than a factor of $6/5$, we can conclude that the occupation ratio of non-waste strips is at least $5(M-1)/6M$; or conversely, each A-item of size s (asymptotically) occupies an area of size at most $\frac{6M}{5(M-1)} \times s$. Hence we have the following lemma.

Lemma 1 *Asymptotically, each A-item of size s can occupy an area of size at most* $\frac{6M}{5(M-1)} \times s$ *under* RTDH.

3.2 Packing of α-, β-, or γ-items

In the following, we use $m(x)$ to denote the number of (x)-bins that is currently used by the algorithm; $m(x)$ is initialized to zero and it may change dynamically. Let $m'(\beta)$ (resp. $m'(\gamma)$) denote the number of bins containing β-item (resp. γ-item), except for (β, β)-bins (resp. (γ, γ)-bins); i.e.,

$$m'(\beta) \stackrel{\text{def}}{=} m(\alpha, \beta) + m(\beta) + m(\beta, \gamma) + m(\alpha, \beta, \gamma) \text{ and}$$
$$m'(\gamma) \stackrel{\text{def}}{=} m(\alpha, \gamma) + m(\gamma) + m(\beta, \gamma) + m(\alpha, \beta, \gamma).$$

By using the above notations, the packing of α-, β-, or γ-items can be described as follows.

Packing of α-, β-, or γ-items

Packing of α-item: If there is a bin of type (β), (γ), or (β, γ), then pack the α-item into one of such bins and change its type accordingly; i.e., to (α, β), (α, γ), and (α, β, γ), respectively; otherwise, open a new bin as an (α)-bin and pack the item into the bin.

Packing of β-item: If there is a (β, β)-bin that contains one β-item, then pack the β-item into the bin and close it; otherwise, execute the following operations:

- If $m(\beta, \beta) < 4m'(\beta)$, then open a new bin as a (β, β)-bin and pack the item into the bin (note that the (β, β)-bin contains only one β-item at this time).
- Else if there is a bin of type (γ) or (α, γ) then pack the item into one of such bins and change its type accordingly; i.e., to (β, γ), (α, β, γ), respectively.
- Else if there is a bin of type (α) then pack the item into this bin and change its type to (α, β).
- Otherwise, open a new bin as a (β)-bin and pack the item into the bin.

Packing of γ-item: If there is a (γ, γ)-bin that contains only one γ-item, then pack the item into the bin and close it; otherwise, execute the following operations:

- If $m(\gamma, \gamma) < 4m'(\gamma)$, then open a new bin as a (γ, γ)-bin and pack the item into the bin (note that the (γ, γ)-bin contains only one γ-item at this time).
- Else if there is a bin of type (β) or (α, β) then pack the item into one of such bins and change its type accordingly; i.e., to (β, γ), (α, β, γ), respectively.
- Else if there is a bin of type (α) then pack the item into this bin and change its type to (α, γ).
- Otherwise, open a new bin as a (γ)-bin and pack the item into the bin.

4 Analysis

Let $s(b)$ denote the size of item b, and $h(b)$ denote the (maximum) size of the area that is spent for packing item b under algorithm RTDH_M, where value $h(b)$ (resp. $s(b)$) will also be referred to as the h-value (resp. s-value) of item b. For example, if $b \in D_{2,2}$, since each $D_{2,2}$-bin can contain at most four $D_{2,2}$-items (regardless of the size of those items), the size of area spent for item b is determined as $1/4$. For brevity, in what follows, we denote $\sum_{b \in L} h(b)$ by $h(L)$, and $\sum_{b \in L} s(b)$ by $s(L)$ for any set L.

In the following, we will prove an upper bound on the worst case ratio of RTDH_M, in terms of function h. Let $\mathcal{T}$ be the set of all possible sets of rectangular items that can be packed into a unit-square bin, and let

$$\bar{H} \stackrel{\text{def}}{=} \sup_{L' \in \mathcal{T}} h(L').$$

The following theorem relates function h with an upper bound on the worst case ratio.

Theorem 1

$$r(\mathsf{RTDH}_M(L)) \leq \bar{H}.$$

Proof. Let $L_1, L_2, \ldots, L_{OPT(L)}$ be sublists of given list L each of which is packed into a single bin under an optimal (off-line) algorithm OPT. Since $\mathsf{RTDH}_M(L) \leq h(L) + M^2 - 1 + 10$, and since $h(L_k) \leq \bar{H}$ for each $1 \leq k \leq OPT(L)$, we have $\mathsf{RTDH}_M(L) \leq \sum_{k=1}^{OPT(L)} h(L_k) + M^2 + 9 \leq OPT(L) \times \bar{H} + M^2 + 9$. Hence, we have

$$\begin{aligned} r(\mathsf{RTDH}_M(L)) &= \lim_{OPT(L) \to \infty} \sup_L \left(\frac{\mathsf{RTDH}_M(L)}{OPT(L)} \right) \\ &\leq \bar{H} \end{aligned}$$

which completes the proof. □

4.1 h-values for each item

In this subsection, we will derive an upper bound on the h-value for each class of items. If b is not an item in $\alpha \cup \beta \cup \gamma$, we will have the following bounds on $h(b)$.

Lemma 2 *For any item* $b = (x, y)$,

$$h(b) \leq \begin{cases} \dfrac{6s(b)M}{5(M-1)} & \text{if } b \in A \\ \dfrac{yM}{i(M-1)} & \text{if } b \in B_i \\ \dfrac{xM}{j(M-1)} & \text{if } b \in C_j \\ \dfrac{1}{i \times j} & \text{if } b \in D_{i,j} - (\alpha \cup \beta \cup \gamma) \end{cases}$$

Proof. The case for $b \in A$ is clear from Lemma 1. Since each (closed) B_i-bin is filled with B_i-items with total size at least $\frac{i}{i+1} \times \frac{M-1}{M}$, we can say that each B_i-item of height y occupies an area of size at most $\frac{yM}{M-1} \times \frac{1}{i}$. A similar claim holds for C_j-items. Finally, since each $D_{i,j}$-item trivially occupies $1/(i \times j)$ of a bin, the lemma follows. □

If b is an item in $\alpha \cup \beta \cup \gamma$, upper bound on the h-value can be derived in a more complicated manner. Let $n_L(\xi)$ denote the number of ξ-items in list L. First, we give an upper bound on the number of bins used by RTDH_M.

Lemma 3 *When the packing of all items in L completes, it holds*

$$\begin{aligned} m(\beta,\beta) &\leq (4/9)\,(n_L(\beta)+1) \\ m'(\beta) &\leq (1/9)\,(n_L(\beta)+1) \\ m(\gamma,\gamma) &\leq (4/9)\,(n_L(\gamma)+1) \quad \textit{and} \\ m'(\gamma) &\leq (1/9)\,(n_L(\gamma)+1) \end{aligned}$$

Proof. Initially, it holds $m(\beta,\beta) = 4m'(\beta) = 0$. Since the input of a β-item increases $m(\beta,\beta)$ by one if $m(\beta,\beta) < 4m'(\beta)$, and increases $m'(\beta)$ by one otherwise, we have $4m'(\beta) - m(\beta,\beta) = \delta_1$, where $0 \leq \delta_1 \leq 4$. On the other hand, by the description of the algorithm, we have $n_L(\beta) = 2m(\beta,\beta) + m'(\beta) + \delta_2$, where $-1 \leq \delta_2 \leq 0$. By solving the above equations, we have

$$\begin{aligned} m'(\beta) &= \frac{n_L(\beta)}{9} - \frac{2\delta_1 - \delta_2}{9} \quad \text{and} \\ m(\beta,\beta) &= \frac{4n_L(\beta)}{9} - \frac{\delta_1 + 4\delta_2}{9}. \end{aligned}$$

Since $0 \leq \delta_1 \leq 4$ and $-1 \leq \delta_2 \leq 0$, the first two inequalities hold. The proof for γ can be done in a similar manner. □

By using the above lemma, we have the following bound on the h-values of α-, β-, and γ-items.

Lemma 4 *If list L contains an α-item b_1, a β-item b_2, and a γ-item b_3, then one of the following three conditions holds:*

$$\begin{aligned} &(h(b_1) = 0, \quad h(b_2) \leq 4/9, \quad \textit{and} \quad h(b_3) \leq 5/9) \\ &(h(b_1) = 0, \quad h(b_2) \leq 5/9, \quad \textit{and} \quad h(b_3) \leq 4/9) \quad \textit{or} \\ &(h(b_1) \leq 1, \quad h(b_2) \leq 4/9, \quad \textit{and} \quad h(b_3) \leq 4/9). \end{aligned}$$

Proof. First, we show that for any list L, one of the following three cases holds:

Case 1: $m(\alpha) = m(\beta) = m(\alpha,\beta) = 0$,
Case 2: $m(\beta) = m(\gamma) = m(\beta,\gamma) = 0$, or
Case 3: $m(\alpha) = m(\gamma) = m(\alpha,\gamma) = 0$.

By the description of algorithm RTDH_M, (α)- and (β)-bins can never coexist. Similarly, (α)- and (γ)-bins never coexist, and (α)- and (β,γ)-bins never coexist. Hence we have

$$m(\alpha)(m(\beta) + m(\gamma) + m(\beta,\gamma)) \;=\; 0. \tag{1}$$

By a similar reason, we have the following equalities:

$$\begin{aligned} m(\beta)(m(\gamma) + m(\alpha,\gamma)) &= 0, \quad \text{and} \qquad (2) \\ m(\gamma)(m(\beta) + m(\alpha,\beta)) &= 0. \qquad (3) \end{aligned}$$

Next, let us prove the following equality:

$$m(\alpha,\beta)m(\alpha,\gamma) \;=\; 0. \tag{4}$$

Assume Equality (4) does not hold. Then, an (α,β)-bin ξ_1 and an (α,γ)-bin ξ_2 must coexist. Without loss of generality, let us assume ξ_2 is created before ξ_1. There are two possible cases to create an (α,β)-bin; i.e., an α-item is put into a (β)-bin, or a β-item is put into an (α)-bin. If ξ_1 is created by the former case, $m(\beta) \geq 1$ and $m(\alpha,\gamma) \geq 1$ must hold simultaneously, which contradicts to Equation (2). On the other hand, if ξ_1 is created by the latter case, by the description of RTDH_M, the existence of (α,γ)-bin must be checked before checking the existence of (α)-bin, a contradiction. Hence the claim follows.

By Equation (1), if $m(\alpha) \neq 0$, then $m(\beta) + m(\gamma) + m(\beta,\gamma) = 0$ must hold (it corresponds to Case 2). Assume $m(\alpha) = 0$. Then, if $m(\beta) \neq 0$, by Equation (2), $m(\gamma) + m(\alpha,\gamma) = 0$ must hold (since $m(\alpha) = 0$ is assumed, it corresponds to Case 3). Otherwise, that is, if $m(\alpha) = 0$ and $m(\beta) = 0$ simultaneously hold, by Equation (3), $m(\gamma) = 0$ or $m(\alpha,\beta) = 0$ must hold. Since the latter case corresponds to Case 1, the remaining case to be examined is that of $m(\alpha) = m(\beta) = m(\gamma) = 0$. Now, consider Equation (4). The equation implies that either $m(\alpha,\beta) = 0$ or $m(\alpha,\gamma) = 0$ holds; where the former implies Case 1, and the latter implies Case 3. Hence the claim holds.

Let $\tilde{m}$ denote the total number of bins used for items except for α-, β-, or γ-items in RTDH_M. Then, the total number of bins used by RTDH_M for packing all items in list L is given by

$$\begin{aligned} \mathsf{RTDH}_M(L) \;\leq\; & \tilde{m} + m(\alpha) + m(\beta) + m(\gamma) + m(\alpha,\beta) \\ & + m(\beta,\gamma) + m(\alpha,\gamma) + m(\beta,\beta) \\ & + m(\gamma,\gamma) + m(\alpha,\beta,\gamma). \end{aligned} \tag{5}$$

Hence, for Case 1, we have

$$\begin{aligned} \mathsf{RTDH}_M(L) &\leq \tilde{m} + m(\beta,\beta) + m'(\gamma) + m(\gamma,\gamma) \\ &\leq \tilde{m} + \left(\frac{4}{9}\right) n_L(\beta) + \left(\frac{5}{9}\right) n_L(\gamma) + 1, \end{aligned}$$

where the last inequality is due to Lemma 3. Similarly, for Case 2, we have

$$\begin{aligned}
&\mathsf{RTDH}_M(L) \\
\leq\ & \tilde{m} + n_L(\alpha) + m(\beta,\beta) + m(\gamma,\gamma) \\
\leq\ & \tilde{m} + n_L(\alpha) + \left(\frac{4}{9}\right) n_L(\beta) + \left(\frac{4}{9}\right) n_L(\gamma) + \frac{8}{9},
\end{aligned}$$

and for Case 3, we have

$$\begin{aligned}
\mathsf{RTDH}_M(L) &\leq \tilde{m} + m(\beta,\beta) + m'(\beta) + m(\gamma,\gamma) \\
&\leq \tilde{m} + \left(\frac{5}{9}\right) n_L(\beta) + \left(\frac{4}{9}\right) n_L(\gamma) + 1.
\end{aligned}$$

Hence the lemma follows. □

4.2 List L' with a big h-value

By Theorem 1, in the following, we will try to estimate an upper bound on $\bar{H}$ as precisely as possible. Let L' be any list in $\mathcal{T}$ that can be packed into a unit-square bin under an optimal algorithm OPT.

In the following proofs, for brevity, we use symbols X_1, Y_1, X_2 and Y_2, that is defined as follows:

- X_1 is the total width of C_1-items contained in L',
- Y_1 is the total height of B_1-items contained in L'.
- X_2 is the total width of C_2-items contained in L', and
- Y_2 is the total height of B_2-items contained in L'.

Note that if L' contains $B_1 \cup C_1$-items, then the total h-values of them is at most $6(X_1+Y_1)/5$ and the total s-values of them is at least $(X_1+Y_1)/2$. Similarly, if L' contains $B_2 \cup C_2$-items, then the total h-values of them is at most $3(X_2+Y_2)/5$ and the total s-values of them is at least $(X_2+Y_2)/3$.

Note that any $D_{1,5} \cup D_{5,1}$-item can be replaced by $B_1 \cup C_1$ items without changing h/s value; thus in the following, we will omit the selection of $D_{1,5} \cup D_{5,1}$-items.

Lemma 5 *If L' contains no $D_{1,1}$-items, $h(L') \leq 2.7778$.*

Proof. By Lemma 4, without loss of generality, we may assume $h(b) = 5/9$ for $b \in \beta$ and $h(b) = 4/9$ for $b \in \gamma$.

Hence, in the following, we may consider the following six cases separately:

	β	γ	$D_{1,3}$	$D_{3,1}$	$h(L') <$
Case 1	0	*	*	*	2.67
Case 2	1	*	*	*	2.7778
Case 3	2	1	1	0	2.7514
Case 4	2	1	0	0	2.7656
Case 5	2	0	1	1	2.7653
Case 6	2	0	1	0	2.7645
Case 7	2	0	0	1	2.7645
Case 8	2	0	0	0	2.7778

For an instance , we olny give the concrete computation of the case 5 , the other cases are simular to this .

Case 5: Since $X_1+Y_1 < 1/3$, L' can contain at most one $D_{4,1}$-itemand any other item b fulfills $h(b)/s(b) < 9/4$,

$$\begin{aligned}
h(L') &< h' + \frac{1}{3} + \frac{1}{3} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\
&\quad + \left(\frac{9}{4}\right)\left(s' - \frac{1}{8} - \frac{1}{8} - \frac{1}{10} - \frac{X_1+Y_1}{2}\right) \\
&< 2.7653.
\end{aligned}$$

Hence the lemma follows. □

Lemma 6 *If L' contains one $D^0_{1,1}$-item, $h(L') \leq 2.78$.*

Proof. Let p (resp. q or r or o) be the number of $\beta \cup \gamma$-items (resp. $D^0_{1,2} \cup D^0_{2,1}$-items or $D_{1,3} \cup D_{3,1}$-items or $D_{2,2}$-itmes) in L'. In the following we consider the following cases separately:

	(p,q,r,o)	$h(L') <$
Case 1	$(2,0,0,1)$	2.76
Case 2	$(2,0,0,0)$	2.78
Case 3	$(1,1,0,1)$	2.7756
Case 4	$(1,1,0,0)$	2.7689
Case 5	$(1,0,1,1)$	2.7789
Case 6	$(1,0,1,0)$	2.7789
Case 7	$(1,0,0,\leq 2)$	2.7734
Case 8	$(0,0,2,*)$	2.7567
Case 9	$(0,*,\leq 1,*)$	2.7709

We also take one case (case 7) as an example .

Case 7: Let $h' \stackrel{\text{def}}{=} 1 + 5/9\ (= 14/9)$ and $s' \stackrel{\text{def}}{=} 1 - 3/10 - 1/6\ (= 8/15)$. We may consider three subcases; i.e., (1) L' contains two $D_{1,4} \cup D_{4,1}$-items and one $D_{2,2}$-item; (2) L' contains at most one $D_{1,4} \cup D_{4,1}$-item and one $D_{2,2}$-item; and (3) L' contains no $D_{1,4} \cup D_{4,1}$-items and at most two $D_{2,2}$-items. In the first case, since $X_1 + Y_1 < 1/6$,

$$\begin{aligned}
h(L') < h' + 2 \times \frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\
+ \left(\frac{9}{5}\right)\left(s' - 2 \times \frac{1}{10} - \frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\
< 2.7556
\end{aligned}$$

in the second case, since $X+Y < 11/30$, $h(L') < 2.7734$. and in the third case, since $X_1 + Y_1 < 7/30$, $h(L') < 2.7245$.

Hence the lemma follows. □

Lemma 7 *If L' contains an α-item, then $h(L') \leq 2.7834$*

Proof. If an α-item costs zero bin, $h(L') < \frac{3}{4} \times \frac{10}{3} = 2.5$. Hence, in the following, without loss of generality, we assume an α-item costs one bin, which implies that a β-item costs 4/9-bin, and a γ-item costs 4/9-bin (see Lemma 4).

Let p (resp. q or r) be the number of $\beta \cup \gamma$-items (resp. $D^0_{1,2} \cup D^0_{2,1}$-items or $D_{1,3} \cup D_{3,1}$-items) in L'. In the following we consider the following cases separately:

	(p,q,r)	$h(L') <$
Case 1	$(2,0,0)$	2.7834
Case 2	$(0,2,0)$	2.78
Case 3	$(1,1,0)$	2.7812
Case 4	$(1,0,1)$	2.7823
Case 5	$(1,0,0)$	2.7823
Case 6	$(0,1,1)$	2.7813
Case 7	$(0,1,0)$	2.7812
Case 8	$(0,0,2)$	2.7812
Case 9	$(0,0,1)$	2.7812
Case 10	$(0,0,0)$	2.7678

Case 1: Let $h' \stackrel{\text{def}}{=} 1+4/9+4/9 \ (=17/9)$ and $s' \stackrel{\text{def}}{=} 1-1/4-1/6-1/6 \ (=5/12)$. Note that $X_1+Y_1 < 1/3$ and $X_2+Y_2 < 2/3-(X_1+Y_1)$. If it contains one $D_{2,2}$-item,

$$\begin{aligned} h(L') &< h' + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} + \frac{3(X_2+Y_2)}{5} \\ &\quad + \left(\frac{8}{5}\right)\left(s' - \frac{1}{9} - \frac{X_1+Y_1}{2} - \frac{X_2+Y_2}{3}\right) \\ &< 2.7834 \end{aligned}$$

and if not, since it can contain at most one $D_{2,3} \cup D_{3,2}$-item, $h(L') < 2.7681$.

Case 4: Let $h' \stackrel{\text{def}}{=} 1+4/9+1/3 \ (=16/9)$ and $s' \stackrel{\text{def}}{=} 1-1/4-1/6-1/8 \ (=11/24)$. First, consider the case in which L' contains one $D_{1,4} \cup D_{4,1}$-item. Note that $X_1+Y_1 < 13/60$. If it contains one $D_{2,2}$-item, since $X_2+Y_2 < 11/20-(X_1+Y_1)$,

$$\begin{aligned} h(L') &< h' + \frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} + \frac{3(X_2+Y_2)}{5} \\ &\quad + \left(\frac{8}{5}\right)\left(s' - \frac{1}{9} - \frac{1}{10} - \frac{X_1+Y_1}{2} - \frac{X_2+Y_2}{3}\right) \\ &< 2.7823, \end{aligned}$$

and if not, since it can contain at most one $D_{2,3} \cup D_{3,2}$-item,

$$\begin{aligned} h(L') &< h' + \frac{1}{4} + \frac{1}{6} + \frac{6(X_1+Y_1)}{5} \\ &\quad + \left(\frac{15}{8}\right)\left(s' - \frac{1}{10} - \frac{1}{12} - \frac{X_1+Y_1}{2}\right) \\ &< 2.7670. \end{aligned}$$

Next, consider the case in which L' contains no $D_{1,4} \cup D_{4,1}$-items. Since $X_1+Y_1 < 5/12$, if L' contains one $D_{2,2}$-item,

$$\begin{aligned} h(L') &< h' + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ &\quad + \left(\frac{9}{5}\right)\left(s' - \frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\ &< 2.7622, \end{aligned}$$

and if not,

$$\begin{aligned} h(L') &< h' + \frac{6(X_1+Y_1)}{5} + 2\left(s' - \frac{X_1+Y_1}{2}\right) \\ &< 2.7778. \end{aligned}$$

Case 5: Let $h' \stackrel{\text{def}}{=} 1+4/9 \ (=13/9)$ and $s' \stackrel{\text{def}}{=} 1-1/4-1/6 \ (=7/12)$. First, consider the case in which L' contains two $D_{1,4} \cup D_{4,1}$-items. Since $X_1+Y_1 < 4/15$, if L' contains one $D_{2,2}$-item

$$\begin{aligned} h(L') &< h' + 2\times\frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ &\quad + \left(\frac{9}{5}\right)\left(s' - 2\times\frac{1}{10} - \frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\ &< 2.7645, \end{aligned}$$

and if not,

$$\begin{aligned} h(L') &< h' + 2\times\frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ &\quad +2\left(s' - 2\times\frac{1}{10} - \frac{X_1+Y_1}{2}\right) \\ &< 2.7645, \end{aligned}$$

Next, consider the case in which L' contains one $D_{1,4} \cup D_{4,1}$-item. Since $X_1+Y_1 < 7/15$ and L' can contain at most one $D_{2,2}$-item,

$$\begin{aligned} h(L') &< h' + \frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ &\quad +2\left(s' - \frac{1}{10} - \frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\ &< 2.7823. \end{aligned}$$

Finally, consider the case in which L' contains no $D_{1,4} \cup D_{4,1}$-items and two $D_{2,2}$-items. Since $X_1+Y_1 < 1/3$,

$$\begin{aligned} h(L') &< h' + 2\times\frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ &\quad +2\left(s' - 2\times\frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\ &< 2.7334. \end{aligned}$$

Case 6: Let $h' \stackrel{\text{def}}{=} 1+1/2+1/3 \ (=11/6)$ and $s' \stackrel{\text{def}}{=} 1-1/4-1/5-1/8 \ (=17/40)$. First, consider the case in which L' contains one $D_{1,4} \cup D_{4,1}$-item. Note that $X_1 + Y_1 < 13/60$. If L' contains one $D_{2,2}$-item, since $X_2 + Y_2 < 29/60 - (X_1 + Y_1)$,

$$\begin{aligned} & h(L') \\ < \; & h' + \frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} + \frac{3(X_2+Y_2)}{5} \\ & + \left(\frac{8}{5}\right)\left(s' - \frac{1}{9} - \frac{1}{10} - \frac{X_1+Y_1}{2} - \frac{X_2+Y_2}{3}\right) \\ < \; & 2.78, \end{aligned}$$

and if not,

$$\begin{aligned} h(L') \; < \; & h' + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ & +2\left(s' - \frac{1}{10} - \frac{X_1+Y_1}{2}\right) \\ < \; & 2.7767. \end{aligned}$$

Next, consider the case in which L' contains no $D_{1,4} \cup D_{4,1}$-items. Since $X_1 + Y_1 < 5/12$, if L' contains one $D_{2,2}$-item,

$$\begin{aligned} h(L') \; < \; & h' + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ & + \left(\frac{15}{8}\right)\left(s' - \frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\ < \; & 2.7813, \end{aligned}$$

and if not,

$$\begin{aligned} h(L') \; < \; & h' + \frac{6(X_1+Y_1)}{5} + 2\left(s' - \frac{X_1+Y_1}{2}\right) \\ < \; & 2.7667. \end{aligned}$$

Case 8: Let $h' \stackrel{\text{def}}{=} 1 + 1/3 + 1/3 \ (= 5/3)$ and $s' \stackrel{\text{def}}{=} 1 - 1/4 - 1/8 - 1/8 \ (= 1/2)$. First, consider the case in which L' contains two $D_{1,4} \cup D_{4,1}$-items. Note that $X_1 + Y_1 < 1/10$. If L' contains one $D_{2,2}$-item, since $X_2 + Y_2 < 13/30 - (X_1 + Y_1)$,

$$\begin{aligned} & h(L') \\ < \; & h' + 2 \times \frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} + \frac{3(X_2+Y_2)}{5} \\ & + \left(\frac{8}{5}\right)\left(s' - 2 \times \frac{1}{10} - \frac{1}{9} - \frac{X_1+Y_1}{2} - \frac{X_2+Y_2}{3}\right) \\ < \; & 2.7812, \end{aligned}$$

and if not, since it can contain at most one $D_{2,3} \cup D_{3,2}$-item,

$$\begin{aligned} h(L') \; < \; & h' + 2 \times \frac{1}{4} + \frac{1}{6} + \frac{6(X_1+Y_1)}{5} \\ & + \left(\frac{15}{8}\right)\left(s' - 2 \times \frac{1}{10} - \frac{1}{12} - \frac{X_1+Y_1}{2}\right) \\ < \; & 2.7659. \end{aligned}$$

Next, consider the case in which L' contains one $D_{1,4} \cup D_{4,1}$-item. Since $X_1 + Y_1 < 3/10$ and L' can contain at most one $D_{2,2}$-item,

$$\begin{aligned} h(L') \; < \; & h' + \frac{1}{4} + \frac{1}{4} + \frac{6(X_1+Y_1)}{5} \\ & + \left(\frac{9}{5}\right)\left(s' - \frac{1}{10} - \frac{1}{9} - \frac{X_1+Y_1}{2}\right) \\ < \; & 2.7767. \end{aligned}$$

The computation of the case 2,3,7,9 and 10 are also likely, so here ,we omit it .

Hence the lemma follows. □

From the above lemmas, we have the following theorem.

Theorem 2

$$r(\mathsf{RTDH}_M) \; < \; 2.7834.$$

5 Concluding Remarks

In this paper, we proposed an on-line algorithm for two-dimensional bin packing problem. This algorithm is an extension of algorithm ROUND and HARMONIC, and for $M = 6$, it has a worst case ratio strictly less than 2.7834, which is better than any on-line two-dimensional bin-packing algorithm for known to date. Obviously, when $M > 6$, it is plausible to improve the asymptotic worst case ratio. A natural question is left open, that is, "can we apply this algorithm RTDH to solve on-line three dimensional packing problem."

References

[1] D. Blitz, A. Van Valiet, and G. J. Woeginger. Lower bounds on the asymptotic worst-case ratio of on-line bin packing algorithms. unpublished manuscript, 1996.

[2] J. Csirik and A. Van Vliet. An on-line algorithm for multidimensional bin packing. *Oper. Res. Lett.*, 13:149–158, 1993.

[3] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide for the Theory of NP-Completeness.* Freeman, San Francisco, CA, 1979.

[4] C. C. Lee and D. T. Lee. A simple on-line bin packing algorithm. *J. Assoc. Comput. Mach.*, 32:562–572, 1985.