The paper is organized as follows: first, we describe the Martel algorithm and give a simpler analysis to prove his main result. In Section 3 we present our heuristic H_7. Section 4 contains the proof of our main result, and we finish this paper with a short conclusion.

2. THE MARTEL RESULT

We will strongly exploit the results in [5]. Therefore, we rewrite briefly the algorithm by Martel. It is based on a simple classification of the elements in a given list. He defined the following sets:

$$C_0 = \{x_i \mid 1 \geqslant x_i > \tfrac{2}{3}\},$$
$$C_1 = \{x_i \mid \tfrac{2}{3} \geqslant x_i > \tfrac{1}{2}\},$$
$$C_2 = \{x_i \mid \tfrac{1}{2} \geqslant x_i > \tfrac{1}{3}\},$$
$$C_3 = \{x_i \mid \tfrac{1}{3} \geqslant x_i > \tfrac{1}{4}\},$$
$$C_4 = \{x_i \mid \tfrac{1}{4} \geqslant x_i > 0\}.$$

Let $c_i = |C_i|$, $i = 0, ..., 4$, where an $x \in C_i$ will be called a C_i-item or C_i-piece. As Martel mentioned, "the motivation for this positioning is to allow items to be packed based on the set to which they belong." We denote this algorithm by H_4. It works in the following way:

1. Form the sets C_i, $i = 0, ..., 4$.

2. Let $k = \lceil \min(c_1, c_2)/2 \rceil$. Split C_1 into two subsets: C_1^s contains the smallest k elements of C_1, and C_1^b contains the remaining elements. We split similarly C_2 into C_2^s and C_2^b. Arbitrarily select pairs from C_1^s and C_2^s. If they fit in a bin, create a bin containing both elements. If the pair does not fit, we put only C_1^s into an empty bin.

3. Put each C_0-piece and C_1^b-piece into separate bins.

4. Put the remaining C_2-pieces (C_2^s- and C_2^b-pieces) into bins, two into each.

5. Until we run out of C_3-pieces, pack C_3-elements into those bins which contain a single C_1-piece.

6. Any remaining C_3-pieces are put three to an empty bin.

7. Put C_4-pieces into bins using the Next-Fit rule.

We denote this algorithm by H_4. For proving the main result Martel used an important lemma which we state here in a more general form.

Lemma 2.1 (Martel [5]). *For two arbitrary disjoint item sets C_i, C_j let $k = \lceil \min(c_i, c_j)/2 \rceil$. Let H be any heuristic which splits C_i and C_j into two subsets each. C_i^s shall contain the k smallest elements of C_i, and C_i^b contains the remaining elements. An analogous splitting is done for C_j. Then, H arbitrarily selects pairs from C_i^s and C_j^s. If they fit in a bin, it creates a bin containing both elements. Let us suppose that H pairs $m(\leqslant k)$ elements in this way. Then the maximum number of pairs consisting of elements of C_i and C_j in an optimum packing does not exceed $m+k$.*

Proof. If $m=k$, the statement is obvious. Now suppose that $m<k$. Therefore, there are elements $x_i \in C_i^s$ and $x_j \in C_j^s$ such that $x_i + x_j > 1$. The best possible pairing technique is to put together at most m elements of C_i^b with elements of C_j^s, m elements of C_i^s with elements of C_j^b, and the remaining $k-m$ elements of C_j^s. Thus, we can not pack together more than $2m+(k-m)=m+k$ elements. ▮

While having analyzed the algorithm of Martel, we have found a simpler proof for the main theorem which was stated by Martel as follows:

THEOREM 2.2 (Martel [5]). *For any list* L *we obtain* $H_4(L) \leqslant \frac{4}{3}L^* + 2$.

Proof. Let us suppose, that our statement is not true. Then we assume the existence of a *minimal counterexample*, i.e., a list L of items with $H_4(L) > \frac{4}{3}L + 2$ and the cardinality of L minimal. It is obvious that this list does not contain any C_4-item. We will distinguish two different subcases:

Case A. We suppose that Step 6 creates at least one bin. In this case

$$L^* \geqslant c_0 + c_1 + \left\lceil \frac{(c_2 + c_3) - (c_0 + c_1)}{3} \right\rceil,$$

and therefore,

$$\begin{aligned} H_4(L) &= c_0 + c_1 + \left\lceil \frac{c_2 - m}{2} \right\rceil + \left\lceil \frac{c_3 - (c_1 - m)}{3} \right\rceil \\ &\leqslant c_0 + \frac{2}{3} c_1 + \frac{1}{2} c_2 + \frac{1}{3} c_3 - \frac{m}{6} + 2 \\ &\leqslant L^* + \frac{1}{3} c_0 + \frac{1}{6} c_2 - \frac{m}{6} + 2 \\ &\leqslant L^* + \frac{1}{3} \left(c_0 + \frac{1}{2} c_2 \right) + 2 \\ &\leqslant \frac{4}{3} L^* + 2, \end{aligned}$$

which is a contradiction.

Case B. Let us suppose that Step 6 does not create new bins. Applying Lemma 2.1 to Step 2, we get

$$L^* \geqslant c_0 + c_1 + \left\lceil \frac{c_2 - (k+m)}{2} \right\rceil \geqslant c_0 + c_1 + \frac{1}{2} c_2 - \frac{k}{2} - \frac{m}{2}.$$

Since $m \leqslant k = \min\{c_2/2, c_1/2\}$, we can conclude

$$\begin{aligned} H_4(L) &= c_0 + c_1 + \left\lceil \frac{c_2 - m}{2} \right\rceil \\ &\leqslant c_0 + c_1 + \frac{1}{2} c_2 - \frac{m}{2} + 1 \\ &\leqslant L^* + \frac{k}{2} + 1 \\ &\leqslant L^* + \frac{1}{4} c_1 + 1 \\ &\leqslant \frac{5}{4} L^* + 1, \end{aligned}$$

which is again a contradiction. ∎

There are many lists which prove that this upper, bound is tight. For example, we can consider the following list L_n with $n = 2m$ and $m \in \mathbb{N}$. L_n contains m pieces of C_0-items and m pieces of C_3-items with sizes $3/4 - \varepsilon$ and $1/4 + \varepsilon$, respectively. Then $H_4(L_n) = m + \frac{m}{3} = \frac{4}{3}m$, and $L_n^* = m$. This implies $H_4(L_n)/L_n^* = 4/3$ for each n.

3. THE H_7 ALGORITHM

Considering the H_4 algorithm by Martel we can realize that it works better than in its worst-case for many subcases (see the proof of Case B). Martel mentioned that "if we could handle C_3 and C_4 better (perhaps by splitting C_4 into two sets, one of which has elements in $(0, \frac{1}{5}]$, the other in $(\frac{1}{5}, \frac{1}{4}]$), we might be able to improve the worst-case ratio."

In spite of the fact that this idea seemed to be very easy, almost 10 years passed and the Martel result has not been improved. Now we give a new linear time algorithm with a $\frac{5}{4}$ worst-case ratio which we denote by H_7. Before presenting our heuristic we classify the elements in a given list as follows:

$C_0 = \{x_i \mid 1 \geqslant x_i > \frac{4}{5}\}$,

$C_1 = \{x_i \mid \frac{4}{5} \geqslant x_i > \frac{2}{3}\}$,

$C_2 = \{x_i \mid \frac{2}{3} \geqslant x_i > \frac{1}{2}\}$,

$C_3 = \{x_i \mid \frac{1}{2} \geqslant x_i > \frac{3}{8}\}$,

$C_4 = \{x_i \mid \frac{3}{8} \geqslant x_i > \frac{1}{3}\}$,

$C_5 = \{x_i \mid \frac{1}{3} \geqslant x_i > \frac{1}{4}\}$,

$C_6 = \{x_i \mid \frac{1}{4} \geqslant x_i > \frac{1}{5}\}$,

$C_7 = \{x_i \mid \frac{1}{5} \geqslant x_i > 0\}$.

Let $c_i = |C_i|$, $i = 0, ..., 7$. Note that during the description of the algorithm c_i always denotes the number of elements of C_i which have not yet been assigned to any bin.