ON THE NUMBER OF BINS REQUIRED TO PACK n PIECES USING THE NEXT-FIT ALGORITHM

by

Micha Hofri

Technical Report #203

(revised March 1981)
Abstract

Bin packing into identical bins of size 1 and an infinite supply of pieces, with i.i.d. sizes, is considered under the Next-Fit packing procedure. The first two moments of $A_n$, the number of bins used to pack $n$ pieces are computed when the pieces sizes are uniformly distributed. A simple approximate pgf for $A_n$ is also given.
1. Introduction

1.1. The problem of bin packing can be defined as follows: Given a set of \( n \) pieces \( \{ R_i \} \), each of length \( X_i \), and a set of bins \( B_i \), each with the same size \( b \), determine which subsets of \( \{ R_i \} \) should be packed in each bin such that

a) The capacity of each bin is not exceeded;

b) The minimum number of bins is used.

This problem has been shown to be "NP-hard", i.e. it essentially requires the (possibly implicit) enumeration of all packing schemes [2].

An example with \( n = 8 \), \( X_i = \{3, 1, 4, 2, 4, 4, 4, 4\} \), \( b = 9 \) is in Fig. 1a.

Recently there is increasing interest in estimating the performance of simple packing (and scheduling) algorithms, especially where the simplicity of those is traded-off consciously against the higher complexity of optimal procedures. See e.g. [1] and further references there. Two venues have been taken: a) estimating worst case behavior, which implies the consideration of some specific set of pieces (tasks) contrived to show the algorithm's poorest performance; b) defining some probability measure over the possible sets of pieces and computing distributions (or at least moments) of performance criteria with respect to this measure. The latter is the road here taken.

1.2. This note is in a sense a continuation of [1]; there the main effort was to prove that subject to the assumptions detailed below, the processes that describe the Next-Fit packing procedure are

---

*This is a one-dimensional problem; the width or any other descriptors of the pieces are immaterial.*
Fig. 1 Example of bin packing

a. Optimal Packing  
b. Next Fit Packing
well behaved. Few calculations were attempted. Here the main thrust is computational.

1.3 Below we define precisely the environment and the Next-Fit (NF) algorithm. From these we derive equations for the number of bins used under NF to pack \( n \) pieces, \( A_n \). It seems to be rather easy to evaluate \( \mathbb{E}(A_n) \), but the evaluation of higher moments and certainly the distribution is less transparent. For the former we outline a procedure and use it to calculate \( \text{Var}(A_n) \); for the distribution we suggest an approximate generating function which turns out to be easy to compute and quite accurate (except that it over-estimates the tails of the distribution.

1.4 Specifically we consider the following situation: An infinite sequence of pieces \( \{R_i, i \geq 1\} \) of sizes \( \{X_i, i \geq 1\}, X_i \sim F \), is to be packed into an infinite set of identical bins \( B_i \), each of size \( 1 \), using the Next-Fit algorithm; nearly the most primitive one possible:

**Next-Fit Packing**

1) \( k = 1, i = 1 \)

2) Pack pieces \( R_k, R_{k+1}, \ldots, R_j \) into \( B_i \), as long as

\[
\sum_{i=k}^{j} X_i \leq 1.
\]

3) \( k = j+1, i = i+1 \), repeat step 2.

Thus a 'discarded' bin is never reexamined. An example is in Fig. 1b.

We note that this algorithm is 'real-time' - i.e. for each piece it requires approximately the same (little here) processing. Its low efficiency is directly related to this.

1.5 In [1] the processes \( \{L_i\} \) - occupancy of discarded bins \( B_i \) and \( \{N_i\} \) - number of pieces in such bins were considered. Here we focus on:
A_n - the number of bins used to pack n pieces.

It will become apparent that the following variable is of interest as well:

T_n - occupancy of the last bin used, when n pieces are packed.

The basic relations will be written for a general piece size distribution, but useful results are only obtained here for the case X_i \sim U(0,1), independently. More on this in Section 4.
The complete dynamics of the packing process are expressed in the following relation, considering the manners by which the packing of \( n \) pieces can give rise to \( \ell \) occupied bins with the last one filled to level \( x \):

\[
A_n = A_{n-1} + \delta_n
\]  

where

\[
\delta_n = \begin{cases} 
1 & \text{when } R_n \text{ overflows to a new bin} \\
0 & \text{otherwise.} 
\end{cases}
\]

Here the auxiliary variable \( T_n \) defined above is useful, in relating \( \delta_n \) to \( A_{n-1} \):

Define

\[
P_\ell(Q, x) = \text{Prob}(A_n = \ell, T_n \leq x), \quad 1 \leq \ell \leq n, \; 0 < x \leq 1.
\]

Thus \( P_1(\ell, x) = F(x) \) for \( \ell = 1 \) and zero otherwise.

In the sequel we shall always assume \( X_i \) to have a probability density function (pdf), and then so does \( T_n \). We shall use the notations

\[
f(x) = \frac{dF(x)}{dx}
\]

\[
p_n(\ell, x) = \frac{dP_n(\ell, x)}{dx}
\]

\[
p_n(\ell) = P_n(\ell, 1)
\]

\[
F_{T_n}(x) = \sum_{\ell=1}^{n} P_n(\ell, x)
\]

\[
f_{T_n}(x) = \frac{dF_{T_n}(x)}{dx}.
\]

The basic equation

2.1 The process \( A_n \) evolves according to

\[
A_n = A_{n-1} + \delta_n
\]
The value for for n=1 is evident from (3). The first part in (4) corresponds to the case where \( R_n \) fits into \( A_{n-1} \), and the second one takes care of an overflowing piece.

For uniformly distributed pieces sizes (4) reduces to

\[
p_n(\ell, x) = P_{n-1}(\ell, x) + P_{n-1}(\ell-1) - P_{n-1}(\ell-1,1-x). \tag{5}
\]

The marginal density of \( T_n \) is, from (4), using (3)

\[
f_T(x) = f* f_{T_{n-1}}(x) + f(x)[1-F_{T_{n-1}}(1-x)], \quad f_{T_{n-1}}(x) = f(x). \tag{6}
\]

For \( X_i \sim U(0,1) \) (6) has the solution

\[
f_T(x) \begin{cases} 
1 & n = 1 \\
2x & n > 1 \quad 0 \leq x \leq 1 
\end{cases} \tag{7}
\]

No closed form is available for general \( f(\cdot) \). For this reason, mainly, in the sequel only uniformly distributed piece sizes are considered, unless otherwise stated.

2.2 For some computations the following generating functions are convenient:

\[
\beta^n(z, x) = \sum_{\ell=1}^{n} p_n(\ell, x)z^\ell
\]
Equation (5) in terms of $\beta$ becomes

$$\beta^n(x, z) = \beta^{n-1}(x, z) + z[\beta^{n-1}(1, z) - \beta^{n-1}(1, 1-x)], \quad n \geq 1, \quad (8)$$

where subscript $x$ denotes differentiation,

$$\beta^0(x, z) = z, \quad \beta^n(z, 0) = 0.$$ 

No closed form solution has been obtained for (8), although the calculation of 3.3 points to a feasible way; however, its form is simple enough to calculate $\beta^n(z, x)$ recursively for moderate $n$; for $n$ beyond five or six hand calculation is inadvisable.

A calculation of $\beta^n(\cdot, \cdot)$ for $n \leq 32$ was done in an attempt to find a common pattern. This required a small fraction of a second on a rather fast computer. We return to this function below, in 3.5.

3. Moments and approximations

3.1 If (5) is integrated over $0 \leq x \leq 1$ one obtains

$$p_n(\ell) = \int_0^1 p_{n-1}(\ell, x)dx + p_{n-1}(1) - \int_0^1 p_{n-1}(1, x)dx \quad (9)$$

multiplying by $\ell$ and summing for $\ell \geq 1$ yields

$$E(A_n) = \sum_{\ell \geq 1} \ell \int_0^1 p_{n-1}(\ell, x)dx + \sum_{\ell \geq 1} (\ell-1)p_{n-1}(\ell-1) + \sum_{\ell \geq 1} p_{n-1}(\ell-1)$$

$$- \sum_{\ell \geq 1} (\ell-1)\int_0^1 p_{n-1}(\ell-1, x)dx - \sum_{\ell \geq 1} \int_0^1 p_{n-1}(\ell-1, x)dx$$

$$= E(A_{n-1}) + 1 - \int_0^1 \tilde{F}_n(x)dx$$
Since $E(T_{n-1})$ is available from (7) we have

$$E(A_n) = \begin{cases} 1 & n = 1 \\ \frac{2}{3} n + \frac{1}{6} & n > 1. \end{cases}$$

(11)

3.2 An alternative way to obtain (10) is to note that (this is correct for general $f(\cdot)$ as well) taking the expectation of (1),

$$E(A_n) = E(A_{n-1}) + \text{Prob}(R_n \text{ starts a new bin})$$

$$= E(A_{n-1}) + \int_0^1 f_{T_{n-1}}(u)P(X>1-u)du$$

$$= E(A_{n-1}) + 1 - f_{T_{n-1}}^{-1} F(1)$$

(12)

For $f(\cdot) = 1$ (12) reduces to (10). This is also the only nontrivial case where $E(A_n) - E(A_{n-1})$ is constant (for $n>2$).

3.3 The interest in the second moment is nearly as great as in $E(A_n)$, since it will give a probabilistic statement on the extent to which we may assume $E(A_n)$ to characterize the evolution of $A_n$, especially for large $n$. From (1), on squaring and taking expectation we get

$$V(A_n) = V(A_{n-1}) + E(\delta_n)[1 - E(\delta_n) - 2E(A_{n-1})] + 2E(A_{n-1} \delta_n)$$

(13)

For this recursion the initial value is $V(A_1) = 0$. The last term in (13) is the only one not known so far:

$$V_{n-1} = E(A_{n-1} \delta_n) = \sum_{\ell=1}^n \ell \int_0^1 x p_{n-1}(\ell,x)dx.$$ 

(14)
Multiplying equation (5) by $\ell x$, summing for all $\ell \geq 1$ and integrating $x$ over $(0,1)$ one obtains after some routine manipulations

\[ y_n = y_{n-1} + \frac{1}{2} \{ E(A_{n-1}) - E(T_{n-1}^2) + 2E(T_{n-1}^2) \} - \sum_{\ell=1}^{n} \ell \int_0^1 x^2 p_{n-1}(\ell, x) dx \]

(15)

so we only netted a more complicated term! Call this last term in (15) $\theta_{n-1}$, and use (5) multiplied by $\ell x^2$ to write a recursion for it. This time, because of the even power of $x$ in the definition of $\theta_n$ no higher degree terms are preserved, and we obtain the following relation

\[ \theta_n = y_{n-1} - \theta_{n-1} + \{ E(T_{n-1}) - E(T_{n-1}^2) + \frac{1}{3} \{ E(A_{n-1}) + E(T_{n-1}^3) \} \} \]

(16)

Denoting the terms within braces in (15) and (16) by $f_1(n)$ and $f_2(n)$ respectively, we immediately obtain

\[ y_n = f_1(n-1) - f_2(n-1) + f_1(n) \]  

(17)

Collecting all these and evaluating (17) produces

\[ y_n = \begin{cases} 
1/2 & n=1 \\
1 & n=2 \\
17/12 & n=3 \\
(20n+4)/45 & n \geq 4 
\end{cases} \]

(18)

and (13) duly results in
\( V(A_n) = \begin{cases} 
0 & n=1 \\
1/4 & n=2 \\
17/36 & n=3 \\
(32n-13)/180 & n \geq 4 
\end{cases} \) (19)

3.4 Since both \( E(A_n) \) and \( V(A_n) \) \( \propto n \), it means that for not too small \( n \)
likely deviations of \( A_n \) from \( E(A_n) \) are \( O(n^{1/2}) \), and their relative
magnitude is therefore \( O(n^{-1/2}) \). This allows us to say that \( A_n \) is well
represented by \( E(A_n) \) of (11), for sufficiently large \( n \).

3.5 The method of 3.3 seems to be applicable to higher moments as well,
but with rapidly increasing complexity. Thus while the possibility
exists to thus calculate \( \beta_n(\cdot) \) of paragraph 2:2 term by term, the pro-
spect is not attractive. Examining equation (1) we note that the ob-
stacle is the dependence between \( A_{n-1} \) and \( \delta_n \), or alternatively \( A_{n-1} \) and
\( T_{n-1} \). Intuitively, for not too small \( n \), unless \( A_n \) is at - or near - its
extreme values (1 and \( n \)), \( T_n \) should not be too sensitive to its exact
value. Thus the following approximation suggests itself:

Consider again equation (5). Note that

\[ \int_0^1 p_{n-1}(\ell, x)dx = p_{n-1}(\ell)[1 - \alpha_{n-1}(\ell)] \] (20)

where \( \alpha_n(\ell) = E(T_n \mid A_n = \ell) \).

Integrating \( x \) out in (5) we have

\[ p_{n}(\ell) = p_{n-1}(\ell)[1 - \alpha_{n}(\ell)] + p_{n-1}(\ell-1)\alpha_{n-1}(\ell-1). \] (21)

Using the near "insensitivity" of \( T \) to \( A \) we write

\[ \hat{p}_{n}(\ell) = \hat{p}_{n-1}(\ell)[1 - \alpha_{n-1}] + \hat{p}_{n-1}(\ell-1)\alpha_{n-1} \] (22)
where the carat denotes the approximate values. The unconditional $\alpha_n$ are simply $E(T_n)$, available from (10), and thus

$$\hat{p}_n(\ell) = \frac{1}{3} \hat{p}_{n-1}(\ell) + \frac{2}{3} \hat{p}_{n-1}(\ell-1) \quad n>2. \quad (23)$$

Defining

$$\hat{\beta}_n(z) = \sum_{\ell=1}^{n} \hat{p}_n(\ell)z^\ell$$

one gets

$$\hat{\beta}_n(z) = \hat{\beta}_{n-1}(z) \frac{1+2z}{3} \quad (1+2z)^{n-2}$$

$$\hat{\beta}_2(z)(\frac{1+2z}{3})^{n-2} \quad (24)$$

where for $\hat{\beta}_2(z)$ we actually take the accurate $\beta^2(z,1)$, as available from (8)

$$\beta^2(z,1) = \frac{1}{2} (z+z^2). \quad (25)$$

3.6 Thus we have an approximation for $p_n(\ell)$ in compact form, which except for $\ell=1$ or $n$ is expected to be reasonable. Actually - as a brief reflection will show it should - it turns out to overestimate those tail probabilities. Thus for $n=8$ we find the typical pattern

<table>
<thead>
<tr>
<th>$\ell$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_8(\ell)$</td>
<td>$2.48 \cdot 10^{-4}$</td>
<td>$3.1498 \cdot 10^{-2}$</td>
<td>$3.7426 \cdot 10^{-1}$</td>
<td>1.5305</td>
</tr>
<tr>
<td>$\hat{p}_8(\ell)$</td>
<td>$6.858 \cdot 10^{-3}$</td>
<td>$8.9163 \cdot 10^{-2}$</td>
<td>$4.938 \cdot 10^{-1}$</td>
<td>1.50891</td>
</tr>
</tbody>
</table>
One might expect it will improve, in some sense, as the extreme values become less likely, with increasing n.

If (24) is differentiated at \( z=1 \) we obtain the correct result, (11) for \( \text{E}(A_n) \). This can more simply be gleaned from (23).

3.7 As the table above suggests we should expect the variance to be overestimated by \( \hat{\text{p}}_n(\ell) \). From (13) we obtain for \( n>2 \) \( \text{V}(A_n) = \frac{8n-7}{36} \).

Thus \( \text{V}(A_n) \) would be overestimated by 25% even for large \( n \). Higher moments will be probably even further off, but the probabilities near the mode (\( \sim \frac{2}{3} n \)) are quite well approximated.

4. Comments

4.1 The unique role of the U(0,1) distribution in the foregoing may seem a bit surprising. Note that we insisted not only on the uniformity, but that the support of the piece size distribution equals the bin size. Any deviation seems to land one in a quagmire of complex calculations. While these can be pushed recursively (in (4) or (6)), no pattern seems to emerge. In particular, eq. (17) has no apparent equivalent. We tend to ascribe this specificity to the simple form of the distribution of \( S_n \) - the sums of n piece sizes: \( F_{S_n}(x)=x^n! \), on \( x \in [0,1] \), but it is probably a matter of taste.
4.2 The form of eqs. (12) and (19) suggests a simple diffusion approximation for $A_n$, but it does not seem to serve any use for our purposes.

4.3 While Next-Fit seems to be the only reasonable "real time" algorithm, i.e., one that packs pieces as they come, and requires approximately the same effort for each, with a slightly higher expenditure of processing the packing can be materially improved. Examples are: sorting the piece in increasing (or decreasing) piece size - and then effecting the Next-Fit procedure; or when not all the pieces are immediately available for sorting, scanning with each piece for the first bin (the one with the lowest index) that will accommodate it. This is the First-Fit packing algorithm. Both algorithms and numerous variations have been shown to improve materially over Next-Fit in terms of worst case behavior, and can be expected to do so on the average as well. Their stochastic analysis however is quite intricate.
References
