On the Sum-of-Squares Algorithm for Bin Packing

JANOS CSIRIK
University of Szeged, Szeged, Hungary
DAVID S. JOHNSON
AT&T Labs - Research, Florham Park, New Jersey
CLAIRE KENYON
Brown University, Providence, Rhode Island
JAMES B. ORLIN and PETER W. SHOR
MIT, Cambridge, Massachusetts
and
RICHARD R. WEBER
University of Cambridge, Cambridge, England

In this paper we present a theoretical analysis of the on-line *Sum-of-Squares* algorithm (SS) for bin packing along with several new variants. SS is applicable to any instance of bin packing in which the bin capacity B and item sizes $s(a)$ are integral (or can be scaled to be so), and runs in time $O(nB)$. It performs remarkably well from an average case point of view: For any discrete distribution in which the optimal expected waste is sublinear, SS also has sublinear expected waste. For any discrete distribution where the optimal expected waste is bounded, SS has expected waste at most $O(\log n)$. We also discuss several interesting variants on SS, including a randomized $O(nB \log B)$-time on-line algorithm SS^* whose expected behavior is essentially optimal for all discrete distributions. Algorithm SS^* depends on a new linear-programming-based pseudopolynomial-time algorithm for solving the NP-hard problem of determining, given a discrete distribution F, just what is the growth rate for the optimal expected waste.

Categories and Subject Descriptors: F.2.2 [**Nonnumerical Algorithms and Problems**]: Computations on discrete structures; Sequencing and scheduling

General Terms: Algorithms, Experimentation, Theory

Additional Key Words and Phrases: Approximation algorithms, average case analysis, bin packing

---

A preliminary version of this article appeared in the *Proceedings of the 32nd Annual ACM Symposium on the Theory of Computing*, 2000, ACM, New York, 208–217.

Authors' addresses: J. Csirik, Dept. of Computer Sciences, University of Szeged, Szeged, Hungary, email: csirik@inf.u-szeged.hu; D. S. Johnson, AT&T Labs - Research, Room C239, 180 Park Avenue, Florham Park, NJ 07932, email: dsj@research.att.com; C. Kenyon, Computer Science Department, Brown University, Providence, RI 02912, email: claire@cs.brown.edu; J. B. Orlin, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02139, email: jorlin@mit.edu; P. W. Shor. Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139, email: shor@math.mit.edu; R. R. Weber, Statistical Laboratory, University of Cambridge, Cambridge CB3 0WB, England, email: rrw1@cam.ac.uk.

Permission to make digital/hard copy of all or part of this material without fee for personal or classroom use provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.
© 20YY ACM 0004-5411/20YY/0100-0002 $5.00

1. INTRODUCTION

In the classical one-dimensional bin packing problem, we are given a list $L = (a_1, ..., a_n)$ of items, a bin capacity B, and a size $s(a_i) \in (0, B]$ for each item in the list. We wish to pack the items into a minimum number of bins of capacity B, i.e., to partition the items into a minimum number of subsets such that the sum of the sizes of the items in each subset is B or less. Many potential applications, such as packing small information packets into somewhat larger fixed-size ones, involve integer item sizes, fixed and relatively small values of B, and large values of n.

The bin packing problem is NP-hard, so research has concentrated on the design and analysis of polynomial-time approximation algorithms for it, i.e., algorithms that construct packings that use relatively few bins, although not necessarily the smallest possible number. Of special interest have been *on-line* algorithms, i.e., ones that must permanently assign each item in turn to a bin without knowing anything about the sizes or numbers of additional items, a requirement in many applications. In this paper we shall analyze the *Sum-of-Squares* algorithm, an on-line bin packing algorithm recently introduced in Csirik et al. [1999] that is applicable to any instance whose item sizes are integral (or can be scaled to be so), and is surprisingly effective.

1.1 Notation and Definitions

In what follows, we assume that all items have integer size. Let P be a packing of list L and for $0 \le h \le B$ let $N_P(h)$ be the number of partially-filled bins in P whose contents have total size equal to h. We shall say that such a bin has *level h*. Note that by definition $N_P(0) = N_P(B) = 0$. We call the vector $\langle N_P(1), N_P(2), \ldots, N_P(B-1)\rangle$ the *profile* of packing P.

Definition 1.1. *The* sum of squares $ss(P)$ *for packing* P *is* $\sum_{h=1}^{B-1} N_P(h)^2$.

The *Sum-of-Squares Algorithm* (SS) introduced in Csirik et al. [1999] is an on-line algorithm that packs each item according to the following simple rule: Let a be the next item to be packed and let P be the current packing. A *legal* bin for a is one that is either empty or has current level no more than $B - s(a)$. Place a into a legal bin so as to yield the minimum possible value of $ss(P')$ for the resulting packing P', with ties broken in favor of the highest level, and then in favor of the newest bin with that level. (Our results for SS hold for any choice of the tie-breaking rule, but it is useful to have a completely specified version of the algorithm.)

Note that in deciding where to place an item of size s under SS, the explicit calculation of $ss(P)$ is not required, a consequence of the following lemma.

Lemma 1.2. *Suppose an item of size s is added to a bin of level h of packing P, thus creating packing P', and that $N_P(h+s) - N_P(h) = d$. Then*

$$ss(P') - ss(P) \;=\; \begin{cases} 2d+1, & \text{if } h = 0 \text{ or } h = B - s, \\ 2d+2, & \text{otherwise.} \end{cases}$$

Proof. Straightforward calculation using the facts that $d = N_P(h+s)$ when $h = 0$ and $d = -N_P(h)$ when $h = B - s$. ■

Thus to find the placement that causes the least increase in $ss(P)$ one simply needs to find that i with $N_P(i) \neq 0$ that minimizes $N_P(i+s) - N_P(i)$, $0 \le i \le B - s$

under the convention that $N_P(0)$ and $N_P(B)$ are re-defined to be $1/2$ and $-1/2$ respectively. We currently know of no significantly more efficient way to do this in general than to try all possibilities, so the running time for SS is $O(nB)$ overall.

In what follows, we will be interested in three measures of L and P.

Definition 1.3. *The* size $s(L)$ *of a list* L *is the sum of the sizes of all the items in* L.

Definition 1.4. *The* length $|P|$ *of a packing* P *is the number of nonempty bins in* P.

Definition 1.5. *The* waste $W(P)$ *of packing* P *is* $\sum_{h=1}^{B-1} N_P(h) \cdot \frac{B-h}{B} = |P| - s(L)/B$.

Note that these quantities are related since $|P| \geq s(L)/B$ and hence $W(P) \geq 0$.

We are in particular interested in the average-case behavior of SS for discrete distributions. A *discrete distribution* F consists of a bin size $B \in \mathbb{Z}^+$, a sequence of positive integral sizes $s_1 < s_2 < \cdots < s_J \leq B$, and an associated vector $\bar{p}_F = \langle p_1, p_2, \ldots, p_J \rangle$ of positive rational probabilities such that $\sum_{j=1}^{J} p_j = 1$. In a list generated according to this distribution, the ith item a_i has size $s(a_i) = s_j$ with probability p_j, independently for each $i \geq 1$. We consider two key measures of average-case algorithmic performance. For any discrete distribution F and any algorithm A, let $P_n^A(F)$ be the packing resulting from applying A to a random list $L_n(F)$ of n items generated according to F. Let OPT denote an algorithm that always produces an optimal packing. We then have

Definition 1.6. *The* expected waste rate *for algorithm* A *and distribution* F *is*

$$EW_n^A(F) \equiv E\left[W\left(P_n^A(F)\right)\right] .$$

Definition 1.7. *The* asymptotic expected performance ratio *for* A *and* F *is*

$$ER_\infty^A(F) \equiv \limsup_{n\to\infty} \left(E\left[\frac{|P_n^A(F)|}{|P_n^{OPT}(F)|} \right] \right) .$$

1.2 Our Results

Let us say that a distribution F is *perfectly packable* if $EW_n^{OPT}(F) = o(n)$ (in which case almost all of the bins in an optimal packing are perfectly packed). By a result of Courcoubetis and Weber [1990] that we shall describe in more detail later, the possible growth rates for $EW_n^{OPT}(F)$ when F is perfectly packable are quite restricted: the only possibilities are $\Theta\left(\sqrt{n}\right)$ and $O(1)$. In the latter case we say F is not only perfectly packable but is also a *bounded waste* distribution. In this paper, we shall present the following results.

(1) For any perfectly packable distribution F, $EW_n^{SS}(F) = O(\sqrt{n})$ [Theorem 2.5].
(2) If F is a bounded waste distribution, then $EW_n^{SS}(F)$ is either $O(1)$ or $\Theta(\log n)$ and there is a simple combinatorial property that F must satisfy for the first case to hold [Theorems 3.4 and 3.11].

(3) There is a simple $O(nB)$-time deterministic variant SS' on SS that has bounded expected waste for all bounded waste distributions and $O(\sqrt{n})$ waste for all perfectly packable distributions [Theorem 3.10].

(4) There is a linear program (LP) of size polynomial in B for determining whether F is perfectly packable, and if not, computing the value of $\limsup_{n\to\infty}(EW_n^{OPT}/n)$. If F is perfectly packable, one can determine whether it is a bounded waste distribution by solving no more than B additional LP's obeying similar size bounds. Using current polynomial time algorithms for linear programming, the total time for this process is polynomial in B and the number of bits required to describe the probability vector $\bar{p}_F$ [Theorems 5.3, 5.2, and 5.6]. Note that since the running time is polynomial in B rather than in $\log B$, the algorithm technically runs in pseudopolynomial time. We cannot hope for a polynomial time algorithm unless P = NP since the problem solved is NP-hard [Coffman, Jr. et al. 2000a].

(5) For the case where F is not perfectly packable, there are lower bound examples and upper bound theorems showing that $1.5 \le \max_F ER_\infty^{SS}(F) \le 3$, and that for all lists L, we have $SS(L) \le 3OPT(L)$, where $A(L)$ is the number of bins used when algorithm A is applied to list L [Theorems 4.1 and 4.2].

(6) For any fixed F, there is a randomized $O(nB)$-time on-line algorithm SS_F such that $EW_n^{SS_F}(F) \le EW_n^{OPT}(F)+O(\sqrt{n})$ and hence $ER_\infty^{SS_F}(F) = 1$. Algorithm SS_F is based on SS and, given F, can be constructed using the algorithm of (4) above [Theorem 6.1].

(7) There is a randomized $O(nB)$-time on-line algorithm SS^* that for any F with bin capacity B has $EW_n^{SS^*}(F) = \Theta(EW_n^{OPT}(F))$ and also $EW_n^{SS^*}(F) \le EW_n^{OPT}(F) + O(n^{1/2})$, the latter implying that $ER_\infty^{SS^*}(F) = 1$. This algorithm works by learning the distribution and using the algorithms of (4) and (6) [Theorem 6.3].

(8) SS can maintain its good behavior even in the face of a non-oblivious adversary who gets to choose the item size distribution at each step (subject to appropriate restrictions) [Theorems 7.1 and 7.2].

(9) The good average case behavior of SS is at least partially preserved under many (but not all) natural variations on its sum-of-squares objective function and the accuracy with which it is updated. Moreover, there is a variant of SS that runs in time $O(B^2+n\log B)$ instead of $\Theta(nB)$ and has the same qualitative behavior as specified for SS' above in (3) [Theorems 8.1 through 8.11].

Several of these results were conjectured based on experimental evidence in Csirik et al. [1999], which also introduced the main linear program of (4).

1.3 Previous Results

The relevant previous results can be divided into two classes: (1) results for practical algorithms on specific distributions, and (2) more general (and less practical) results about the existence of algorithms. We begin with (1).

The average case behavior under discrete distributions for standard heuristics has been studied in Albers and Mitzenmacher [1998], Coffman, Jr. et al. [1991], Coffman, Jr. et al. [1993], Coffman, Jr. et al. [1997], Coffman, Jr. et al. [2000a], and Kenyon et al. [1998]. These papers concentrated on the discrete uniform distributions $U\{j,k\}$, where the bin capacity $B = k$ and the item sizes are $1, 2, \ldots, j < k$, all equally likely. If $j = k-1$, the distribution is symmetric and we have by earlier results that the optimal packing and the off-line First and Best Fit Decreasing algorithms (FFD and BFD) all have $\Theta(\sqrt{n})$ expected waste, as do the on-line First Fit (FF) and Best Fit (BF) algorithms [Coffman, Jr. et al. 1991; Coffman, Jr. et al. 1997].

More interesting is the case when $1 \le j \le k-2$. Now the optimal expected waste is $O(1)$ [Coffman, Jr. et al. 1991; Coffman, Jr. et al. 2000a; 2002b], and the results for traditional algorithms do not always match this. In Coffman, Jr. et al. [1991] it was shown that BFD and FFD have $\Theta(n)$ waste for $U\{6,13\}$, and Coffman, Jr. et al. [] identifies a wide variety of other $U\{j,k\}$ with $j < k-1$ for which these algorithms have linear waste. For the on-line algorithms FF and BF, the situation is no better. Although they can be shown to have $O(1)$ waste when $j = O(\sqrt{k})$ [Coffman, Jr. et al. 1991], when $j = k-2$ [Albers and Mitzenmacher 1998; Kenyon et al. 1998], and (in the case of BF) for specified pairs (j,k) with $k \le 14$ [Coffman, Jr. et al. 1993], for most values of (j,k) it appears experimentally that their expected waste is linear. This has been proved for BF and the pairs $(8,11)$ and $(9,12)$ [Coffman, Jr. et al. 1993] as well as all pairs j,k with $j/k \in [0.66, 2/3)$ when k is sufficiently large [Kenyon and Mitzenmacher 2002]. In contrast, $EW_n^{SS}(U\{j,k\}) = O(1)$ whenever $j < k-1$. On the other hand, our current best implementation of basic SS runs in time $\Theta(nB)$ assuming $\log n$ and $\log B$ are no more than the computer word size, compared to $O(n \log B)$ for BF, $O(n + B^2)$ for FFD and BFD [Coffman, Jr. et al.]. (The fastest known implementation of FF is $\Theta(n \log n)$ and so FF is asymptotically slower than SS for fixed B.)

Turning to less distribution-specific results, the first relevant results concerned off-line algorithms. In the 1960's, Gilmore and Gomory [1961; 1963] introduced a deterministic approach to solving the bin packing problem that used linear programming, column generation, and rounding to find a packing that for any list L with J or fewer distinct item sizes is guaranteed to use no more than $OPT(L) + J - 1$ bins. Since $J < B$ for any discrete distribution, this implies an average-case performance that is at least as good as that specified for SS^* in (7) of the previous section, and is in some cases better. This approach, when implemented using the simplex algorithm, seems to work well in practice, but the worst-case running time for such an implementation is conceivably exponential in B: We have no polynomial bound either on the time for simplex to solve the LP's or on the number of LP's that need to be solved. (The LP being implicitly solved can have exponentially many variables.) One can obtain a time bound for this approach that is polynomial in B, but for this one must use the ellipsoid method applied to the dual of the implicit LP, with the column generation subroutine converted to a separation oracle, as in Grötschel et al. [1981] and Karp and Papadimitriou [1982]. Such an approach is unfortunately unlikely to be practical.

A simpler way to produce such good packings in time polynomial in B is to directly solve the basic LP of (4) above and then greedily extract a packing from the variables of a basic optimal solution, as explained in Applegate et al. [2003]. For this we are not restricted to the ellipsoid method and can use the fastest available polynomial time linear programming algorithm, currently that of Vaidya [1989]. A simplistic analysis of the resulting running time yields a worst-case bound of $O(n + (JB)^{4.5} \log^2 n)$, which is linear but with an additive constant that for many distributions would still render the algorithm impractical. As with the Gilmore-Gomory approach, solving the LP's using the simplex method tends to work well in practice, although it again raises the possibility of worst-case times that are exponential in B Applegate et al. [2003].

Theoretically the best approach along these lines is the off-line deterministic algorithm of Karmarkar and Karp [1982] that for any list L never uses more than $OPT(L) + O(\log^2 J)$ bins and takes time $O(n + J^8 \log J \log^2 n)$. Although these guarantees are asymptotically stronger than those for the previous two approaches, the Karmarkar-Karp algorithm is substantially more complicated and once again requires the performance of ellipsoid method steps. (This Karmarkar-Karp algorithm is closely related to the more famous one from the same paper that guarantees a packing within $OPT(L) + O(\log^2(OPT))$ for *all* lists L, independent of the number of distinct item sizes, but for which the best current running time bound is $O(n^8 \log^3 n)$.)

For on-line algorithms, the most general results are those of Rhee [1988], Rhee and Talagrand [1993a] and Rhee and Talagrand [1993b]. Rhee and Talagrand [1993a] proved that for any distribution F (discrete or not) there exists an $O(n \log n)$ on-line randomized algorithm A_F satisfying $EW_n^{A_F}(F) \le EW_n^{OPT}(F) + O(\sqrt{n} \log^{3/4} n)$ and hence $ER_\infty^{A_F}(F) = 1$. (For distributions with irrational sizes and/or probabilities, their results assume a real-number RAM model of computation.) This is a more general result than (6) above, and although the additive error term is worse than the one in (6), the extra factor of $\log^{3/4} n$ appears to reduce to a constant depending only on B when F is a discrete distribution, making the two bounds comparable. Unfortunately, Rhee and Talagrand only prove that such algorithms *exist*. The details of the algorithms depend on a non-constructive characterization of F and its packing properties given in Rhee [1988].

Rhee and Talagrand [1993b] present a single (constructive) on-line randomized algorithm A that works for all distributions F (discrete or not) and has $EW_n^A(F) \le EW_n^{OPT}(F) + O(\sqrt{n} \log^{3/4} n)$, again with the $\log^{3/4} n$ factor likely to reduce to a function of B for discrete distributions. Even so, for discrete distributions this algorithm is not quite as good as our algorithm SS^*, which itself has $EW_n^{SS^*}(F) \le EW_n^{OPT}(F) + O(\sqrt{n})$ for all discrete distributions and in addition gets bounded waste for bounded waste distributions. Moreover, the algorithm of Rhee and Talagrand [1993b] is unlikely to be practical since it uses the Karmarkar-Karp algorithm (applied to the items seen so far) as a subroutine.

The fastest on-line algorithms previously known that guarantee an $O(\sqrt{n})$ expected waste rate for perfectly packable discrete distributions are due to Courcoubetis and Weber, who used them in the proof of their characterization theorem in Courcoubetis and Weber [1990]. These algorithms are distribution-dependent, but for fixed F run in linear time. At each step, the algorithm must solve a linear program whose number of variables is potentially exponential in B, but for fixed F this takes constant time, albeit potentially a large constant. Moreover, for bounded waste distributions, the Courcoubetis-Weber algorithms have

$EW_n^A(F) = O(1)$, whereas the Rhee-Talagrand algorithms cannot provide any guarantee better than $O(\sqrt{n})$. On the other hand, the algorithms of Rhee and Talagrand [1993a] and Rhee and Talagrand [1993b] guarantee $ER_\infty^A(F) = 1$ for all distributions, while Courcoubetis and Weber [1990] only do this for those distributions in which $EW_n^{OPT}(F) = O(\sqrt{n})$.

Thus, although these earlier general approaches rival the packing effectiveness of SS and its variants, and in the case of the offline algorithms actually can do somewhat better, none are likely to be as widely usable in practice (certainly none of the online rivals will be), and none has the elegance and simplicity of the basic SS algorithm.

As to our LP-based approach for evaluating distributions (4), this too has relevant precursors. The main linear program we use turns out to be essentially equivalent to the *arc flow* model for bin packing previously introduced by Valério de Carvalho [1999], although that paper did not address questions of average case behavior. Moreover, the classic LP-based approach to the cutting stock problem of Gilmore and Gomory [1961] and Gilmore and Gomory [1963] once again applies and can be implemented to run in time polynomial in B using the ellipsoid algorithm and the separation oracle results of Grötschel et al. [1981] and Karp and Papadimitriou [1982]. The Gilmore-Gomory based approach is significantly more complicated than ours, however, and at least implicitly involves LP's that are of size exponential rather than polynomial in B. Moreover, no previous paper has even implicitly presented algorithms for identifying bounded-waste distributions.

1.4 Outline of the Paper

The remainder of this paper is organized as follows. In Section 2 we present the details of the Courcoubetis-Weber characterization theorem and prove our result about the behavior of SS under perfectly packable distributions. In Section 3 we prove our results for bounded waste distributions. Section 4 covers our linear-programming-based algorithm for characterizing $EW_n^{OPT}(F)$ given F. In Section 5 we discuss our results about the behavior of SS under linear waste distributions. In Section 6 we discuss our results about how SS can be modified so that its expected behavior is asymptotically optimal for such distributions. Section 7 presents our results about how SS behaves in more adversarial situations. Section 8 covers our results about the effectiveness of algorithms that use variants on the sum-of-squares objective function or trade accuracy in measuring that function for improved running times. We conclude in Section 9 with a discussion of open problems and related results, such as the recent extension of the Sum-of-Squares algorithm to the bin covering problem in Csirik et al. [2001].

2. PERFECTLY PACKABLE DISTRIBUTIONS

In order to explain why the Sum-of-Squares algorithm works so well, we need first to understand the characterization theorem of Courcoubetis and Weber [1990], which we now describe.

Given a discrete distribution F, a *perfect packing configuration* is a length-J vector $\bar{b} = \langle b_1, b_2, \ldots, b_J \rangle$ of nonnegative integers such that $\sum_{j=1}^{J} b_j s_j = B$. Such a configuration corresponds to a way of completely filling a bin with items from

F. That is, if we take b_i items of size s_i, $1 \le i \le J$, we will precisely fill a bin of capacity B. Let Λ_F be the cone generated by the set of all perfect packing configurations for F, that is, the closure under convex combinations and positive scalar multiplication of the set of all such configurations.

Definition 2.1. *A vector $\bar{x} = \langle x_1, \ldots, x_J \rangle$ is* in the interior *of a cone Λ if and only if there exists an $\epsilon > 0$ such that all nonnegative vectors $\bar{y} = \langle y_1, \ldots, y_J \rangle$ satisfying $|\bar{x} - \bar{y}| \equiv \sum_{i=1}^{J} |x_i - y_i| \le \epsilon$ are in Λ.*

Theorem [Courcoubetis and Weber 1990]. Let $\bar{p}_F$ denote the vector of size probabilities $\langle p_1, p_2, \ldots, p_J \rangle$ for a discrete distribution F.

(a) $EW_n^{OPT}(F) = O(1)$ *if and only if $\bar{p}_F$ is in the* interior *of Λ_F.*

(b) $EW_n^{OPT}(F) = \Theta(\sqrt{n})$ *if and only if $\bar{p}_F$ is on the* boundary *of Λ_F, i.e., is in Λ_F but not in its interior.*

(c) $EW_n^{OPT}(F) = \Theta(n)$ *if and only if $\bar{p}_F$ is outside Λ_F.*

The Courcoubetis-Weber Theorem can be used to prove the following lemma, which is key to many of the results that follow:

Lemma 2.2. *Let F be a perfectly packable distribution with bin size B, P be an arbitrary packing into bins of size B, x be an item randomly generated according to F, and P' be the packing resulting if x is packed into P according to SS. Then $E[ss(P')|P] < ss(P) + 2$.*

Proof. The proof relies on the following claim.

Claim 2.3. *If F is a perfectly packable distribution with bin size B, then there is an algorithm A_F such that given any packing P into bins of size B, A_F will pack an item randomly generated according to F in such a way that for each bin level h with $N_P(h) > 0$, $1 \le h \le B-1$, the probability that $N_P(h)$ increases is no more than the probability that it decreases.*

Proof of Claim. The algorithm A_F depends on the details of the Courcoubetis-Weber Theorem. Since F is perfectly packable, $\bar{p}_F$ must be in Λ_F and so there must exist some number m of length-J nonnegative integer vectors $\bar{b}_i$ and corresponding positive numbers α_i satisfying

$$\sum_{j=1}^{J} (b_{i,j} \cdot s_j) \;=\; B, \;\; 1 \le i \le m, \tag{2.1}$$

$$\sum_{i=1}^{m} (\alpha_i \cdot b_{i,j}) \;=\; p_j, \;\; 1 \le j \le J. \tag{2.2}$$

Note that since the $b_{i,j}$ are integral and the p_j are rational, there exists a set of values for the α_i that are rational as well, so assume that they are. Now since the α_i and p_j are all rational, there exists an integer Q such that $Q \cdot \alpha_i$ and $Q \cdot p_j$ are integral for all i and j. Consider the *ideal packing* P^* which has $Q\alpha_i$ copies of bins of type $\bar{b}_i$. We will use P^* to define A_F. Note that by (2.2) P^* contains Qp_j items

(1) Let the set U of as-yet-unordered items initially be set to Y and let $S = 0$ be the initial total size of ordered items.
(2) While $U \neq \emptyset$ and $last(Y)$ is undefined, do the following:
 2.1 If there is an item x in U such that P has a partially filled bin of level $S + s(x)$
  2.1.1 Choose such an x, put it next in the ordering, and remove it from U
  2.1.2 Set $S = S + s(x)$.
 2.2 Otherwise, set $last(Y)$ to be the number of items ordered so far and exit *While* loop.
(3) Complete the ordering by appending the remaining items in U in arbitrary order.

Fig. 1. Procedure for ordering items in bin Y given a packing P

of size j, $1 \leq j \leq J$, and hence a total of Q items. Let $L_F = \{x_1, x_2, \ldots, x_Q\}$ denote the Q items packed into P^*, and denote the bins of P^* as $Y_1, Y_2, \ldots, Y_{|P^*|}$.

Now let P be an arbitrary packing of integer-size items into bins of size B. We claim that for each bin Y of the packing P^*, there is an ordering $y_1, y_2, \ldots, y_{|Y|}$ of the items contained in Y and a special threshold index $last(Y)$, $0 \leq last(Y) < |Y|$, such that if we set $S_i \equiv \sum_{j=1}^{i} s(y_j)$, $0 \leq i \leq |Y|$, then the following holds:

(1) P has partially filled bins with each level S_i, $0 \leq i \leq last(Y)$.
(2) P has no partially filled bin of level $S_{last(Y)} + s(y_i)$ for any $i > last(Y)$.

That such an ordering and threshold index always exist can be seen from Figure 1, which presents a greedy procedure that, given the current packing P, will compute them. Assume we have chosen such an ordering and threshold index for each bin in P^*. Note that $S_{|Y|} = B$ for all such bins Y, since each is by definition perfectly packed.

Our algorithm A_F begins the processing of an item a by first randomly identifying it with an appropriate element $r(a) \in L_F$. In particular, if a is of size s_j, then $r(a)$ is one of the $Q \cdot p_j$ items in L_F of size s_j, with all such choices being equally likely. Note that this implies that for each i, $1 \leq i \leq Q$, the probability that a randomly generated item a will be identified with x_i is $1/Q$.

Having chosen $r(a)$, we then determine the bin into which we should place a as follows. Suppose that in P^*, item $r(a)$ is in bin Y and has index j in the ordering of items in that bin.

(i) If $j = 1$, place a in an empty bin, creating a new bin with level $s(a) = S_1$.
(ii) If $1 < j \leq last(Y)$, place a in a bin with level S_{j-1}, increasing its level to S_j.
(iii) If $j > last(Y)$, place a in a bin of size $S_{last(Y)}$ (or in a new bin if $last(Y) = 0$).

For example, suppose that the items in Y, in our constructed order, are of size 2, 3, 2, and 4 and $last(Y) = 2$. Then $S_1 = 2$, $S_2 = 5$, $S_3 = 7$, $S_4 = B = 11$, $N_P(2)$, $N_P(5) > 0$, and $N_P(7) = N_P(9) = 0$. If $r(a) \in Y$, then it is with equal probability the first 2, the 3, the second 2, or the 4. In the first case it starts a new bin, creating a bin of level 2 and increasing $N_P(2)$ by 1. In the second it goes in a bin of level 2, converting it to a bin of level 5, thus decreasing $N_P(2)$ by 1 and increasing $N_P(5)$ by 1. In the third and fourth cases it goes in a bin of level 5, converting it to a bin of level 7 or 9, depending on the case, and decreasing $N_P(5)$ by 1. Thus when $r(a) \in Y$, the only positive level counts that can change are those for $h \in \{2,5\} = \{S_1, S_2 = S_{last(Y)}\}$, counts can only change by 1, and each count is at least as likely to decline as to increase.

More generally, for any bin Y in P^*, if a is randomly generated according to F and $r(a) \in Y$, then by the law of conditional probabilities $r(a)$ will take on each of the values y_i, $1 \le i \le |Y|$ with probability $p = 1/|Y|$. Thus if $r(a) \in Y$ the probability that the count for level S_i increases equals the probability that it decreases when $1 \le i < last(Y)$. The probability that the count for $S_{last(Y)}$ decreases is at least as large as the probability that it increases (greater if $last(Y) \le |Y| - 2$). And for all other levels with positive counts, the probability that a change occurs is 0. Since this is true for all bins Y of the ideal packing P^*, the Claim follows. ∎

Claim 2.3 is used to prove Lemma 2.2 as follows. Note that the claim implies a bound on the expected increase in $ss(P)$ when a new item is packed under A_F. For any level count $x > 0$, the expected increase in $ss(P)$ given that this particular count changes is, by the claim, at most

$$\frac{1}{2}\Big((x+1)^2 - x^2\Big) + \frac{1}{2}\Big((x-1)^2 - x^2\Big) = 1.$$

More trivially, the expected increase in $ss(P)$ given that a 0-count changes is also at most 1. Since a placement changes at most two counts, this means that the expected increase in $ss(P)$ using algorithm A_F is at most 2. Since SS explicitly chooses the placement of each item so as to minimize the increase in $ss(P)$, we thus must also have that the expected increase in $ss(P)$ under SS is at most 2 at each step. ∎

Lemma 2.2 is exploited using the following result.

LEMMA 2.4. *Suppose P is a packing under SS of a randomly generated list $L_n(F)$, where F is a discrete distribution with bin size B and $n > 0$. Then*

$$E[W(P)] \le \sqrt{(B-1)E[ss(P)]}.$$

PROOF. For $1 \le i \le n$ let $C_i = \sum_{h=1}^{B-1} p[N_P(h) = i]$, i.e., the expected number of levels whose count in P equals i. Then $\sum_{i=1}^{n} C_i = B - 1$ and

$$E[ss(P)] = \sum_{h=1}^{B-1} E\left[N_P(h)^2\right] = \sum_{i=1}^{n} C_i \cdot i^2. \tag{2.3}$$

We now apply the Cauchy-Schwartz inequality, which says that

$$\left(\sum x_i y_i\right)^2 \le \left(\sum x_i^2\right)\left(\sum y_i^2\right).$$

Let $x_i = \sqrt{C_i}$ and $y_i = i\sqrt{C_i}$, $1 \le i \le n$. We then have

$$\left(\sum_{i=1}^{n} C_i \cdot i\right)^2 \le \left(\sum_{i=1}^{n} C_i\right)\left(\sum_{i=1}^{n} C_i i^2\right).$$

Taking square roots and using (2.3), we get

$$E\left[\sum_{h=1}^{B-1} N_P(h)\right] \le \sqrt{(B-1)E[ss(P)]}. \tag{2.4}$$

Since no partially full bin has more than $(B-1)/B < 1$ waste, the claimed result follows. ■

THEOREM 2.5. *Suppose F is a discrete distribution satisfying $EW_n^{OPT}(F) = O(\sqrt{n})$. Then $EW_n^{SS}(F) < \sqrt{2nB}$.*

PROOF. By Lemma 2.2 and the linearity of expectations, we have

$$E[ss(P_n^{SS}(F))] \le 2n.$$

The result follows by Lemma 2.4. ■

3. BOUNDED WASTE DISTRIBUTIONS

In order to distinguish the broad class of bounded waste distributions under which SS performs well, we need some new definitions. If F is a discrete distribution, let U_F denote the set of sizes associated with F.

DEFINITION 3.1. *A level h, $1 \le h \le B-1$, is a* dead-end *level for F if there is no collection of items with sizes in U_F whose total size is $B-h$.*

In other words, if h is a dead-end level then it is impossible to use items whose sizes are in U_F to completely fill a bin whose current contents have total size h. Note that the dead-end levels for F depend only on U_F and can be identified in time $O(|U_F|B)$ by dynamic programming.

OBSERVATION 3.2. *For future reference, note the following easy consequences of the definition of dead-end level.*

(a) *The algorithms A_F of Claim 2.3 in the proof of Lemma 2.2 never create bins that have dead-end levels. (This is because the levels of the bins they create are always the sums of item sizes from a perfectly packed bin.)*
(b) *If F is a perfectly packable distribution, then for no $s_j \in U_F$ is s_j a dead-end level. (Otherwise, no bin containing items of size s_j could be perfectly packed. Since the expected number of such bins in an optimal packing is at least np_j/B, this means that the expected waste would have to be at least np_j/B^2. Since we must have $p_j > 0$ by definition, this implies that we would have linear waste, contradicting the assumption that F is a perfectly packable distribution.)*
(c) *No distribution with $1 \in U_F$ can have a dead-end level, so that in particular the $U\{j,k\}$ do not have dead-end levels.*

A simple example of a distribution that does have dead-end levels is any F that has $B = 6$ and $U_F = \{2,3\}$. Here 5 is a dead-end level for F while 1,2,3,4 are not. There is a sense, however, in which this distribution is still fairly benign.

DEFINITION 3.3. *A level h is* multiply-occurring *for a distribution F if there is some list L with item sizes from U_F such that the SS packing P of L has $N_P(h) > 1$.*

It is easy to verify that there are *no* multiply-occurring levels, dead-end or otherwise, in the above $B = 6$ example.

We shall divide this section into three parts. In subsection 3.1 we show that SS has bounded expected waste for bounded waste distributions with no multiply-occurring dead-end levels. In subsection 3.2 we show that a simple variant on SS has bounded expected waste for *all* bounded waste distributions. In subsection 3.3 we characterize the behavior of SS for bounded waste distributions that do have multiply-occurring dead-end levels.

3.1 A Bounded Expected Waste Theorem for SS

THEOREM 3.4. *If F is a bounded waste distribution with no multiply-occurring dead-end levels, then $EW_n^{SS}(F) = O(1)$.*

To prove this result we rely on the Courcoubetis-Weber Theorem, Lemma 2.2, and the following specialization of a result of Hajek [1982].

Hajek's Lemma. *Let S be a state space and let $\mathcal{F}_k$, $k \geq 1$, be a sequence of functions, where $\mathcal{F}_k$ maps S^{k-1} to probability distributions over S. Let $X_1, X_2, \ldots$ be a sequence of random variables over S generated as follows: X_1 is chosen according to $\mathcal{F}_1(\cdot)$ and X_k is chosen according to $\mathcal{F}_k(X_1, \ldots, X_{k-1})$. Suppose there are constants $b > 1$, $\Delta < \infty$, $D > 0$, and $\gamma > 0$ and a function ϕ from S to $[0, \infty)$ such that*

(a) [Initial Bound Hypothesis]. $E\left[b^{\phi(X_1)}\right] < \infty$.

(b) [Bounded Variation Hypothesis]. *For all $N \geq 1$, $|\phi(X_{N+1}) - \phi(X_N)| \leq \Delta$.*

(c) [Expected Decrease Hypothesis]. *For all $N \geq 1$,*

$$E[\phi(X_{N+1}) - \phi(X_N) | \phi(X_N) > D] \leq -\gamma.$$

Then there are constants $c > 1$ and $T > 0$ such that for all $N \geq 1$, $E\left[c^{\phi(X_N)}\right] < T$.

Note that the conclusion of this lemma implies that there is also a constant T' such that $E[\phi(X_N)] < T'$ for all N. A weaker version of the lemma was used in the analyses of the Best and First Fit bin packing heuristics in Albers and Mitzenmacher [1998], Coffman, Jr. et al. [1993], and Kenyon et al. [1998]. The added strength is not needed for Theorem 3.4, but will be used in the proof of Theorem 3.11.

We prove Theorem 3.4 by applying Hajek's Lemma with the following interpretation. The state space S is the set of all length-$(B-1)$ vectors of non-negative integers $\bar{x} = \langle x_1, x_2, \ldots, x_{B-1} \rangle$, where we view $\bar{x}$ as the profile of a packing that has x_i bins with level i, $1 \leq i \leq B-1$. X_0 is then the profile of the empty packing and X_{i+1} is the profile of the packing obtained by generating a random item according to F and packing it according to SS into a packing with profile X_i. The potential function is

$$\phi(\bar{x}) = \sqrt{\sum_{i=1}^{B-1} x_i^2}.$$

Note that if the hypotheses of Hajek's Lemma are satisfied under this interpretation, then the lemma's conclusion would say that there is a T' such that for all

N,

$$E\left[\sqrt{\sum_{i=1}^{B-1} x_{N,i}^2}\,\right] < T',$$

which implies that $E[x_{N,i}]$ is bounded by T' as well, $1 \le i \le B-1$. Thus the expected waste is less than the constant BT' and Theorem 3.4 would be proved.

Hence all we need to show is that the three hypotheses of Hajek's lemma apply. The Initial Bound Hypothesis applies since the profile of an empty packing is all 0's and hence $\phi(X_0) = 0$. The following lemma implies that Bounded Variation Hypothesis also holds.

LEMMA 3.5. *Let $\bar{x}$ be the profile of a packing into bins of size B, and let $\bar{x}'$ be the profile of the packing obtained from $\bar{x}$ by adding an item to the packing in any legal way. Then*

$$|\phi(\bar{x}') - \phi(\bar{x})| \le 1.$$

PROOF. Consider the case when $\phi(\bar{x}') > \phi(\bar{x})$ and suppose that i is the level whose count increases when the item is packed is level i. We have

$$\begin{aligned}
\phi(\bar{x}') - \phi(\bar{x}) &\le \sqrt{\phi(\bar{x})^2 + (x_i+1)^2 - x_i^2} - \phi(\bar{x}) \\
&= \frac{\left(\sqrt{\phi(\bar{x})^2 + 2x_i + 1} - \phi(\bar{x})\right)\left(\sqrt{\phi(\bar{x})^2 + 2x_i + 1} + \phi(\bar{x})\right)}{\sqrt{\phi(\bar{x})^2 + 2x_i + 1} + \phi(\bar{x})} \\
&= \frac{2x_i + 1}{\sqrt{\phi(\bar{x})^2 + 2x_i + 1} + \phi(\bar{x})} \\
&\le \frac{2x_i + 1}{\sqrt{x_i^2 + 2x_i + 1} + x_i} = 1\,.
\end{aligned}$$

A similar argument handles the case when $\phi(\bar{x}') < \phi(\bar{x})$. ■

To complete the proof of the theorem, we need to show that the Expected Decrease Hypothesis of Hajek's Lemma applies. For this we need the following three combinatorial lemmas.

LEMMA 3.6. *Suppose y is any number and $a > 0$. Then*

$$y - a \le \frac{y^2 - a^2}{2a}.$$

PROOF. Note that $y - a = (y^2 - a^2)/(y + a)$, and then observe that no matter whether $y \ge a$ or $y < a$, this is less than or equal to $(y^2 - a^2)/2a$. ■

LEMMA 3.7. *Let F be a distribution with no multiply-occurring dead-end levels and let P be any packing that can be created by applying SS to a list of items all of whose sizes are in U_F. If $\bar{x}$ is the profile of P and $\phi(\bar{x}) > 2B^{3/2}$, then there is a size $s \in U_F$ such that if an item of size s is packed by SS into P, the resulting* *profile $\bar{x}'$ satisfies*

$$\phi(\bar{x}')^2 \le \phi(\bar{x})^2 - \frac{\phi(\bar{x})}{B^{3/2}}.$$

PROOF. Suppose $\bar{x}$ is as specified and let h be the index for a level at which $\bar{x}$ takes on its maximum value. It is easy to see that

$$x_h \ge \phi(\bar{x})/\sqrt{B}\,. \tag{3.5}$$

Thus $x_h > 2B > 1$ and so by definition h is a multiply-occurring level and hence by hypothesis cannot be a dead-end level for F. Hence there must be a sequence of levels $h = \ell_0 < \ell_1 < \cdots < \ell_m = B$, $m \le B$, such that for $1 \le i \le m$, $\ell_i - \ell_{i-1} \in U_F$. Taking $x_B = 0$ by convention, we have

$$x_h = \sum_{i=0}^{m-1} (x_{\ell_i} - x_{\ell_{i+1}})\,. \tag{3.6}$$

Let $\Delta = \max\{x_{\ell_i} - x_{\ell_{i+1}} : 0 \le q < m\}$, let q be an index which yields this maximum value, and let $s = \ell_{q+1} - \ell_q$. Then by (3.6) we have $\Delta \ge x_h/m \ge x_h/B \ge \phi(\bar{x})/B^{3/2}$, where the last inequality follows from (3.5). By Lemma 1.2 this means that if an item of size s arrives, $\phi(\bar{x})^2$ must decline by at least

$$2(\Delta - 1) \ge 2\left(\frac{\phi(\bar{x})}{B^{3/2}} - 1\right) \ge \frac{\phi(\bar{x}) + 2B^{3/2}}{B^{3/2}} - 2 \ge \frac{\phi(\bar{x})}{B^{3/2}}$$

as claimed. ■

LEMMA 3.8. *Let F be a bounded waste distribution with $U_F = \{s_1, s_2, \ldots, s_J\}$ and let $p_{min} = \min\{p_i : 1 \le i \le J\}$. (Recall that by definition we must have $p_{min} > 0$.) For each i, $1 \le i \le J$, and ϵ, $0 < \epsilon < p_{min}$, let $F[i, \epsilon]$ be the distribution which decreases p_i to $p'_i = (p_i - \epsilon)/(1 - \epsilon)$ and increases all other probabilities p_j to $p'_j = p_j/(1 - \epsilon)$. Then there is a constant $\epsilon_0 > 0$ such that $F[i, \epsilon]$ is a perfectly packable distribution for all i, $1 \le i \le J$, and ϵ, $0 < \epsilon \le \epsilon_0$.*

PROOF. Since F is a bounded waste distribution and $p_i > 0$, $1 \le i \le J$, this follows from the Courcoubetis-Weber theorem, part (a). ■

We can now prove that the Expected Decrease Hypothesis of Hajek's Lemma applies, which will complete the proof of the Theorem 3.4. Let F be a bounded waste distribution with no multiply-occurring dead-end levels, and let ϵ_0 be the value specified for F by Lemma 3.8. Without loss of generality we may assume that $\epsilon_0 < 2$. Let P be a packing as specified in Lemma 3.7 but with profile $\bar{x}$ satisfying $\phi(\bar{x}) > 4B^{3/2}/\epsilon_0 > 2B^{3/2}$. Let i be the index of the size $s \in U_F$ whose existence is proved in Lemma 3.7, and let F_i be the distribution that always generates an item of size s_i.

Consider the two-phase item generation process that first randomly chooses between distributions F_i and $F[i, \epsilon_0]$, the first choice being made with probability ϵ_0 and the second with probability $1 - \epsilon_0$. It is easy to see that this process is just a more complicated way of generating items according to distribution F. Now consider what happens when this process is used to add one item to packing P. If F_i is chosen, then by Lemma 3.7, the value of ϕ^2 declines by at least $\phi(\bar{x})/B^{3/2}$. If

$F[i,\epsilon_0]$ is chosen, the expected value of ϕ^2 increases by less than 2 as a consequence of Lemma 2.2 and the fact that $F[i,\epsilon_0]$ is a perfectly packable distribution (Lemma 3.8). Thus if $\bar{x}'$ is the resulting profile, we have by applying Lemma 3.6 for $a = \phi(\bar{x})$ and $b = \phi(\bar{x}')$ and taking expectations

$$\begin{aligned} E\left[\phi(\bar{x}') - \phi(\bar{x})\right] &< (1-\epsilon_0)(2)\left(\frac{1}{2\phi(\bar{x})}\right) + \epsilon_0\left(-\frac{\phi(\bar{x})}{B^{3/2}}\right)\left(\frac{1}{2\phi(\bar{x})}\right) \\ &< \frac{1}{\phi(\bar{x})} - \frac{\epsilon_0}{2B^{3/2}} \\ &< -\frac{\epsilon_0}{4B^{3/2}} \end{aligned}$$

since $\phi(\bar{x}) > 4B^{3/2}/\epsilon_0$. Thus the Expected Decrease Hypothesis of Hajek's Lemma holds with $D = 4B^{3/2}/\epsilon_0$ and $\gamma = \epsilon_0/4B^{3/2}$, and so Hajek's Lemma applies. Thus $EW_n^{SS}(F) = O(1)$, the conclusion of Theorem 3.4. ■

3.2 Improving on SS for Bounded Waste Distributions

Unfortunately, although SS has bounded expected waste for bounded waste distributions with no multiply-occurring dead-end levels, it does not do so well for *all* bounded waste distributions. Consider the distribution F with $B = 9$, $J = 2$, $s_1 = 2$, $s_2 = 3$, and $p_1 = p_2 = 1/2$. It is easy to see that F is a bounded waste distribution, since 3's by themselves can pack perfectly, and only one 3 is needed for every three 2's in order that the 2's can go into perfectly packed bins. Note, however, that 8 is a multiply-occurring dead-end level for F, so Theorem 3.4 does not apply. In fact, $EW_n^{SS}(F) = \Omega(\log n)$, as the following informal reasoning suggests: It is likely that somewhere within a sequence of $n \log n$ items from F there will be $\Omega(\log n)$ consecutive 2's. These are in turn likely to create $\Omega(\log n)$ bins of level 8, and hence, since 8 is a dead-end level, $\Omega(\log n)$ waste.

Fortunately, this is the worst possible result for SS and a bounded waste distribution, as we shall see below in Theorem 3.11. First, however, let us show how a simple modification to SS yields a variant with the same running time that has $O(1)$ expected waste for all bounded waste distributions.

Like SS, this variant (SS') is on-line. It makes use of a parameterized variant SS_D on the packing rule of SS, where D is a set of levels. In SS_D, we place items so as to minimize $ss(P)$ subject to the constraint that no bin with level in D may be created unless this is unavoidable. In the latter case we start a new bin. SS' works as follows. Let U be the set of item sizes seen so far and let $D(U)$ denote the set of dead-end levels for U. (Initially U is empty and $D(U) = \{h : 1 \le h < B\}$.) Whenever an item arrives, we first check if its size is in U. If not, we update U and recompute $D(U)$. Then we pack the item according to $SS_{D(U)}$. A first observation about SS' is the following.

LEMMA 3.9. *If F is a perfectly packable distribution, then SS' will never create a dead-end level when packing a sequence of items with sizes in U_F.*

PROOF. By Observation 3.2(b), starting a bin with an item whose size is in U_F can never create a dead-end level for U_F. On the other hand, if SS' puts an item in a partially full bin, it must by definition be the case that the new level is not in $D(U)$. Since $D(U)$ can never gain elements as more item sizes are revealed, the new level cannot be in $D(U_F)$ either. Thus it is not a dead-end level for F. ■

THEOREM 3.10.

(i) *If F is a perfectly packable distribution, then $EW_n^{SS'}(F) = O(\sqrt{n})$.*
(ii) *If F is a bounded waste distribution, then $EW_n^{SS'}(F) = O(1)$.*

PROOF. We begin by bounding the expected number of items that can arrive before we have seen all item sizes in U_F. Assume without loss of generality that $U_F = \{s_1, s_2, \ldots, s_J\}$. The probability that the ith item size does not appear among the first h items generated is $(1 - p_i)^h$. Thus, if we again let $p_{min} = \min\{p_i : 1 \le i \le J\}$, the probability that we have not seen all item sizes after the hth item arrives is at most

$$\sum_{i=1}^{J} (1 - p_i)^h \le J\,(1 - p_{min})^h\,.$$

Let t be such that $J(1 - p_{min})^t \le 1/2$. Then for each integer $m \ge 0$, the probability that all the item sizes have not been seen after mt items have arrived is at most $1/2^m$. Thus if M is the number of items that have arrived when the last item size is first seen, we have that for each $m \ge 0$, the probability that $M \in (mt, (m+1)t]$ is at most $1/2^m$.

For (i), note that if P is the packing that exists immediately after the last item size is first seen, then $ss(P) \le M^2$ and

$$E[ss(P)] \le \sum_{m=0}^{\infty} \Big(((m+1)t)^2 \cdot p\big[M \in (mt, (m+1)t]\big] \Big) \le \sum_{m=0}^{\infty} \frac{((m+1)t)^2}{2^m} = 12t^2,$$

which is a constant bound depending only F. After all sizes have been seen, SS' reduces to $SS_{D(U_F)}$, and it follows from Observation 3.2(a) that Lemma 2.2 applies to the latter. We thus can conclude that for any n the packing P_n satisfies

$$E\left[ss(P_n)\right] < 12t^2 + 2n,$$

which by Lemma 2.4 implies that $EW_n^{SS'}(F) = O(\sqrt{n})$, so (i) is proved.

The argument for (ii) mimics the proof of Theorem 3.4. Using the same potential function ϕ we show that Hajek's Lemma applies when SS is replaced by $SS_{D(U_F)}$, F is a bounded waste distribution, and the initial state $\bar{x}$ is taken to be the profile of the packing P that exists immediately after the last item size is first seen by SS'.

To see that the Initial Bound Hypothesis is satisfied, we must show that there exists a constant $b > 1$ such that $E\left[b^{\phi(\bar{x})}\right]$ is bounded. To prove this, let M be the number of items in packing P. It is immediate that $\phi(\bar{x}) = \sqrt{\sum_{i=1}^{B-1} x_i^2} \le M$. Thus if we take $b = 2^{1/(2t)}$ and exploit the analysis used for (i) above we have

$$\begin{aligned} E\left[b^{\phi(\bar{x})}\right] \le E\left[b^M\right] \le \sum_{m=0}^{\infty} b^{(m+1)t} \cdot \frac{1}{2^m} &= \sum_{m=0}^{\infty} \frac{2^{(m+1)/2}}{2^m} \\ &= \sqrt{2} \sum_{m=0}^{\infty} \frac{1}{\sqrt{2}^m} = \frac{2}{\sqrt{2}-1} < 4.83. \end{aligned}$$

Thus the Initial Bound Hypothesis is satisfied. The Bounded Variation Hypothesis again follows immediately from Lemma 3.5. To prove the Expected Decrease Hypothesis, we need the facts that Lemmas 2.2 and 3.7 hold when SS is replaced by $SS_{D(U_F)}$. We have already observed that Lemma 2.2 holds. As to Lemma 3.7, the properties of SS were used in only two places. First, we needed the fact that SS could never create a packing where the count for a dead-end level exceeded 1, an easy observation there since we assumed there were no multiply-occurring dead-end levels. Here the situation is even better since by Lemma 3.9 SS' can never create a packing where the count for a dead-end level is nonzero.

The other property of SS used in proving Lemma 3.7 was simply that, in the terms of the proof of that lemma, it could be trusted to pack an item of size $s = \ell_{q+1} - \ell_q$ in such a way as to reduce $ss(P)$ by at least as much as it would be reduced by placing the item in a bin of level ℓ_q. $SS_{D(U_F)}$ will clearly behave as desired, since level ℓ_{q+1}, as it is constructed in the proof, is not a dead-end level, and so bins of level ℓ_q are legal placements for items of size $\ell_{q+1} - \ell_q$ under $SS_{D(U_F)}$.

We conclude that Lemma 3.7 holds when $SS_{D(U_F)}$ replaces SS, and so the Expected Decrease Hypotheses of Hajek's Lemma is satisfied. Thus the latter Lemma applies, and the proof of bounded expected waste can proceed exactly as it did for SS. ■

3.3 Worst-Case Behavior of SS for Bounded Waste Distributions

THEOREM 3.11. *If F is a bounded waste distribution that has multiply-occurring dead-end levels, then $EW_n^{SS}(F) = \Theta(\log n)$.*

We divide the proof of this theorem into separate upper and lower bound proofs. These are by a substantial margin the most complicated proofs in the paper, and readers may prefer to skip this section on a first reading of the paper. None of the later sections depend on the details of these proofs.

3.3.1 *Proof of the $O(\log n)$ Upper Bound.* For this result we need to exploit more of the power of Hajek's Lemma (which surprisingly is used in proving the lower bound as well as the upper bound). We will also need a more complicated potential function. Let $\mathcal{D}_F$ denote the set of dead-end levels for F and let $\mathcal{L}_F$ denote the set of levels that are not dead-end levels for F. We shall refer to the latter as *live* levels in what follows. For a given profile $\bar{x}$, define $\tau_D(\bar{x}) = \sum_{i \in \mathcal{D}_F} x_i^2$ and $\tau_L(\bar{x}) = \sum_{i \in \mathcal{L}_F} x_i^2$. Note that $\phi(\bar{x}) = \sqrt{\tau_D(\bar{x}) + \tau_L(\bar{x})}$. Our new potential function ψ must satisfy two key properties.

(1) Hajek's Lemma applies with the potential function ψ and, as before, X_i representing the profile after SS has packed i items generated according to F.
(2) For any live level h,

$$\psi(\bar{x}) \geq \sqrt{\tau_L(\bar{x})} \geq x_h. \tag{3.7}$$

Let us first show that the claimed upper bound will follow if we can construct a potential function ψ with these properties. Since Hajek's Lemma applies, there exist constants $c > 1$ and $T > 0$ such that for all $N > 0$,

$$E\left[c^{\psi(X_N)}\right] \leq T. \tag{3.8}$$

We can use (3.8) to separately bound the sums of the counts for live and dead levels. For each live level h, the component $X_{n,h}$ of the final packing profile X_n satisfies $X_{n,h} \le \psi(X_n) < c^{\psi(X_n)}/\log_e c$, and so we have

$$E\left[\sum_{h\in\mathcal{L}_F} X_{n,h}\right] \le E\left[B\frac{c^{\psi(X_n)}}{\log_e c}\right] \le \frac{BT}{\log_e c} = O(1). \tag{3.9}$$

In other words, the expected sum of the counts for live levels is bounded by a constant.

To handle the dead-end levels, we begin by noting that (3.8) also implies that for all N and all $\alpha > 1$,

$$P\left[c^{\psi(X_N)} > \alpha T\right] < \frac{1}{\alpha},$$

so if we take logarithms base c and set $\alpha = n^2/T$ we get

$$P\left[\psi(X_N) > 2\log_c n\right] < \frac{T}{n^2}. \tag{3.10}$$

Say that a placement is a *major uphill move* if it increases $ss(P)$ by more than $4\log_c n + 1$. By Observation 3.2(b) and (3.7), we know that whenever an item is generated according to F and packed by SS, one option will be to start a new bin with a live level and hence, no matter where the item is packed, the increase in $ss(P)$ will be bounded by $2\psi(X_N) + 1$. Using (3.10), we thus can conclude that at any point in the packing process, the probability that the next placement is a major uphill move is at most T/n^2. Thus, in the process of packing n items, the expected number of major uphill moves is at most T/n by the linearity of expectations.

Now let us consider the dead-end levels. Suppose the count for dead-end level h is $2B(\log_c n + 1)$ or greater and a bin b with level less than h receives an additional item that brings its level up to h. We claim that bin b, in the process of attaining this level from the time of its initial creation, must have at one time or another experienced an item placement that was a major uphill move.

To see this, let us first recall the tie-ing rule used by SS when it must choose between bins with a given level for packing the next item. Although the rule chosen has no effect on the amount of waste created, our definition of SS specified a particular rule, both so the algorithm would be completely defined and because the particular rule chosen facilitates the bookkeeping needed for this proof. The rule says that when choosing the bin of a given level h in which to place an item, we always pick the bin which most recently attained level h. In other words, the bins for each level will act as a stack, under the "last-in, first-out" rule. Now consider the bin b mentioned above. In the process of reaching level h, it received less than B items, so it changed levels fewer than B times. Note also that by our tie-breaking rule above, we know that every time the bin left a level, that level had the same count that it had when the bin arrived at the level. Thus at least one of the steps in packing bin b must have involved a jump from a level i to a level j such that $N_P(j) \ge N_P(i) + 2(\log_c n + 1)$. By Lemma 1.2 this means that the move caused $ss(P)$ to increase by at least $4(\log_c n + 1) + 1 > 4\log_c n + 1$ and hence was a major uphill move. We conclude that

$$E\left[\sum_{h\in\mathcal{D}_F}(X_{n,h} \;-\; 2B(\log_c n+1))\right]$$

$$\leq \sum_{h\in\mathcal{D}_F} E\left[\Big((X_{n,h}-2B(\log_c n+1)) : X_{n,h} > 2B(\log_c n+1)\Big)\right]$$

$$\leq E\left[\textit{Number of major uphill moves}\right] \;\leq\; \frac{T}{n}$$

and consequently

$$E\left[\sum_{h\in\mathcal{D}_F} X_{n,h}\right] < 2B^2(\log_c n+1)+\frac{T}{n} = O(\log n) \tag{3.11}$$

for fixed F. Combining (3.9) with (3.11), we conclude that

$$EW_n^{SS}(F) < E\left[\sum_{h\in\mathcal{D}_F} X_{n,h}\right] + E\left[\sum_{h\in\mathcal{L}_F} X_{n,h}\right] = O(\log n).$$

Thus all that remains is to exhibit a potential function ψ that obeys (3.7) and the three hypotheses of Hajek's Lemma. Our previous potential function $\phi(\bar{x}) = \sqrt{\tau_L(\bar{x})+\tau_D(\bar{x})}$ obeys (3.7) and the Initial Bound and Bounded Variation Hypotheses. Unfortunately, it does not obey the Expected Decrease Hypothesis for all bounded waste distributions F with multiply-occurring dead-end levels. There can exist realizable packings in which the count for the largest dead-end level is arbitrarily large (and hence so is $\phi(\bar{x})$), and yet any item with size in U_F will cause $\phi(\bar{x})$ to increase. One can avoid such obstacles by taking instead the potential function ψ to be $\sqrt{\tau_L(\bar{x})}$, the variant on ϕ that simply ignores the dead-end level counts. Unfortunately, this function also fails to obey the Expected Decrease Hypothesis, albeit for a different reason. There are relevant situations in which any item with a size in U_F will either cause an increase in $\tau_L(\bar{x})$ or else go in a bin with a dead-end level and thus leave $\tau_L(\bar{x})$ unchanged.

Thus our potential function must somehow deal with the effects of items going into dead-end level bins. Let us say that a profile $\bar{x}'$ is *constructible* from a profile $\bar{x}$ under F if there is a way of adding items with sizes in U_F to dead-end level bins of a packing with profile $\bar{x}$ so that a packing with profile $\bar{x}'$ results. Let

$$\tau_0(\bar{x}) = \min\{\tau_D(\bar{x}') : \bar{x}' \text{ is constructible from } \bar{x} \text{ under } F\}. \tag{3.12}$$

Note for future reference that $\tau_0(\bar{x})$ can never decrease as items are added to the packing. Now let

$$r_D(\bar{x}) = \tau_D(\bar{x}) - \tau_0(\bar{x}). \tag{3.13}$$

Thus $r_D(\bar{x})$ is the amount by which we can reduce $\tau_D(\bar{x})$ by adding items with sizes in U_F into bins with dead-end levels. Our new potential function is

$$\psi(\bar{x}) = \sqrt{\tau_L(\bar{x}) + r_D(\bar{x})}. \tag{3.14}$$

Note that since we must always have $r_D(\bar{x}) \geq 0$, we have $\psi(\bar{x}) \geq \sqrt{\tau_L(\bar{x})}$ and so (3.7) holds for ψ. It remains to be shown that Hajek's Lemma applies to ψ. This is significantly more difficult than showing it applies to ϕ when F has no dead-end levels.

First we prove a technical lemma that will help us understand the intricacies of the $r_D(\bar{x})$ part of our potential function ψ. Recall that if $r_D(\bar{x}) = t$, then there is some list L of items with sizes in U_F that we can add to the dead-end level bins of a packing with profile $\bar{x}$ to get to one with a profile $\bar{y}$ such that $\tau_D(\bar{y}) = \tau_D(\bar{x}) - t$, and no such list of items can yield a profile $\bar{y}'$ with $\tau_D(\bar{y}') < \tau_D(\bar{x}) - t$. In what follows, we will use an equivalent graph-theoretic formulation based on the following definition.

Definition 3.12. *A* reduction graph *G for F is a directed multigraph whose vertices are the dead-end levels for F and for which each arc (h, i) is such that $i - h$ can be decomposed into a sum of item sizes from U_F. Such a graph G is* applicable *to a profile $\bar{x}$ if $outdegree_G(i) \le x_i$ for all dead-end levels i. The profile $G[\bar{x}]$ derived from applying G to $\bar{x}$ is the vector $\bar{y}$ that has $y_i = x_i + indegree_G(i) - outdegree_G(i)$ for all dead-end levels and $y_i = x_i$ for all live levels. We say that G* verifies t for $\bar{x}$ *if $\tau_D(\bar{x}) - \tau_D(G[\bar{x}]) \ge t$.*

Note that $r_D(\bar{x})$ equals the maximum t verified for $\bar{x}$ by some applicable reduction graph G. The list L corresponding to G, i.e., the one that can be added to $\bar{x}$ to obtain $G[\bar{x}]$, is a union of sets of items of total size $i - h$ for each arc (h, i) in G.

Lemma 3.13. *Let G be a reduction graph with the minimum possible number of arcs that verifies $r_D(\bar{x})$ for $\bar{x}$. Then the following three properties hold:*

(i) No vertex in G has both a positive indegree and a positive outdegree.

(ii) Suppose that the arcs of G are ordered arbitrarily as $a_1, a_2, \ldots, a_m$, that we inductively define a sequence of profiles $\bar{y}[0] = \bar{x}, \bar{y}[1], \ldots \bar{y}[m]$ by saying that $\bar{y}[i]$ is derived by applying the graph consisting of the single arc a_i to $\bar{y}[i-1]$, $1 \le i \le m$, and that we define $\Delta[i] = \tau_D(\bar{y}[i-1]) - \tau_D(\bar{y}[i])$, $1 \le i \le m$. Then

$$\sum_{i=1}^{m} \Delta[i] = r_D(\bar{x}) \text{ and } \tag{3.15}$$

$$\Delta[i] > 0, \ 1 \le i \le m. \tag{3.16}$$

(iii) G contains fewer than $\psi(\bar{x})$ copies of any arc (h, i).

Proof. If (i) did not hold, there would be a pair of arcs (h, i) and (i, j) in G for some h, i, j. But note that then the graph G' with these two arcs replaced by (h, j) would also verify $r_D(\bar{x})$ for $\bar{x}$, and would have one less arc, contradicting our minimality assumption.

For (ii), equality (3.15) follows from a collapsing sum argument and the fact that $\bar{y}[m] = G[\bar{x}]$. The proof of (3.16) is a bit more involved. Suppose there were some k such that $\Delta[k] \le 0$. We shall show how this leads to a contradiction. Consider the result of deleting arc $a_k = (h, j)$ from G, thus obtaining new graph G' and new sequences $\bar{y}[i]'$ and $\Delta[i]'$, $1 \le i \le m - 1$. We will show that G' also verifies $r_D(\bar{x})$ for $\bar{x}$, contradicting our minimality assumption.

Note that $\bar{y}[i]' = \bar{y}[i]$, $1 \le i < k$, and hence $\Delta[i]' = \Delta[i]$ for $1 \le i < k$. Thereafter the only difference between $\bar{y}[i]$ and $\bar{y}[i]'$ is that $y[i]'_h = y[i+1]_h + 1$

and $y[i]'_j = y[i+1]_j - 1$. Suppose $i \geq k$ and that $a_i = (r, q)$. Note that by (i), $r \neq j$ and $q \neq h$. Thus we have $y[i]'_r \geq y[i+1]_r$ and $y[i]'_q \leq y[i+1]_q$. We can now apply Lemma 1.2 to bound $\Delta[i]'$. Note first that $\Delta[i]$ as defined is -1 times the quantity evaluated in that lemma, and the second case of the lemma applies here since neither of the dead-end levels q and r can be 0 or B. The lemma thus says that $\Delta[i]' = -(2d+2)$, where d here equals $y[i]'_q - y[i]'_r$. Hence we have

$$\Delta[i]' = 2\left(y[i]'_r - y[i]'_q - 1\right) \geq 2\left(y[i+1]_r - y[i+1]_q - 1\right) = \Delta[i+1].$$

Thus we have by (3.15)

$$\sum_{i=1}^{m-1} \Delta[i]' \geq \sum_{i=1}^{m} \Delta[i] - \Delta[k] \geq \sum_{i=1}^{m} \Delta[i] = r_D(\bar{x}),$$

and so G' verifies $r_D(\bar{x})$ for $\bar{x}$. Since G' has one less arc than G, this violates our assumption about the minimality of G and so yields our desired contradiction, thus proving (3.16).

Finally, let us consider (iii). Suppose there were $\psi(\bar{x})$ copies of some arc (h, i) in G. By (ii) we may assume that these are arcs $a_1, a_2, \ldots, a_{\psi(\bar{x})}$, and that each yields an improvement in τ_D. Thus when the last is applied, the count for level h must have been at least 2 more than the count for level i, and inductively, when arc $a_{\psi(\bar{x})+1-i}$ was applied, the difference in counts had to be at least $2i$. Now by Lemma 1.2, if the count for level h exceeds that for level j by δ, then the decrease in τ_D caused by applying the arc is $2\delta - 2$. Thus by (ii) we have

$$\psi(\bar{x})^2 \geq r_D(\bar{x}) \geq \sum_{i=1}^{\psi(\bar{x})} (4i - 2) = 2\psi(\bar{x})^2,$$

a contradiction. Thus (iii) and Lemma 3.13 have been proved. ■

Now let us turn to showing that Hajek's Lemma applies when ψ plays the role of ϕ. Since the initial state is the empty packing, for which $\psi(\bar{x}) = 0$, the Initial Bound Hypothesis is trivially satisfied. For the Bounded Variation Hypothesis we must show that there is a fixed bound Δ on $|\psi(\bar{x}') - \psi(\bar{x})|$, where $\bar{x}$ is any profile that can occur with positive probability in an SS packing under F and $\bar{x}'$ is any profile that can be obtained by adding an item with size $s \in U_F$ to a packing with profile $\bar{x}$ using SS. We will show this for $\Delta = 10B$. We may assume without loss of generality that $B \geq 2$, as otherwise $EW_n^{SS}(F) = 0$ for all n.

There are two cases, depending on whether $\psi(\bar{x}') \geq \psi(\bar{x})$. First suppose $\psi(\bar{x}') \geq \psi(\bar{x})$. By Lemma 3.6 it suffices to prove that $\psi(\bar{x}')^2 - \psi(\bar{x})^2 \leq 2\Delta\psi(\bar{x}) = 20B\psi(\bar{x})$. By Observation 3.2(b) we know that s is not a dead-end level and hence by (3.7) $x_s \leq \psi(\bar{x})$. Thus by the operation of SS and the fact that $\tau_0(\bar{x})$ cannot decrease, the increase in $\psi(\bar{x})^2$ is at most $(x_s+1)^2 - x_s^2 = 2x_s + 1$. If $x_s = 0$, this is clearly less than $10B$. Otherwise, we have $2x_s + 1 \leq 3x_s \leq 3\psi(\bar{x}) \leq 20B\psi(\bar{x})$, as desired.

Suppose on the other hand that $\psi(\bar{x}') < \psi(\bar{x})$, a significantly more difficult case. We need to show that $\psi(\bar{x}) - \psi(\bar{x}') \leq \Delta = 10B$. Lemma 3.6 again applies, but now requires that we show $\psi(\bar{x})^2 - \psi(\bar{x}')^2 \leq 2\Delta\psi(\bar{x}')$, where the bound is in terms of the resulting profile $\bar{x}'$ rather than the initial one $\bar{x}$. To simplify matters, we shall first show that the former is within a constant factor of the latter. This is not true in general, but we may restrict attention to a case where it provably *is* true. In particular we may assume without loss of generality that $\psi(\bar{x}) \geq 10B$, since otherwise it is obvious that any placement will reduce $\psi(\bar{x})$ by at most $10B$.

LEMMA 3.14. *Suppose F is a bounded waste distribution with $B \geq 2$, $\bar{x}$ is a profile with $\psi(\bar{x}) \geq 10B$, and $\bar{x}'$ is the profile resulting from using SS to place an item of size $s \in U_F$ into a packing with profile $\bar{x}$. Then $\psi(\bar{x}') \geq \psi(\bar{x})/2$.*

PROOF. By hypothesis, $\tau_L(\bar{x}) + r_D(\bar{x}) \geq 100B^2$. We break into cases depending on the relative values of $\tau_L(\bar{x})$ and $r_D(\bar{x})$.

Suppose $\tau_L(\bar{x}) \geq r_D(\bar{x})$, in which case $\tau_L(\bar{x}) \geq \psi(\bar{x})^2/2 \geq 50B^2$. If the new item goes into a dead-end level bin, then $\tau_L(\bar{x})$ remains unchanged and $\psi(\bar{x}') \geq \sqrt{\psi(\bar{x})^2/2} \geq .707\psi(\bar{x}) > \psi(\bar{x})/2$. If on the other hand the new item goes into a bin with a live level, say h, then $\tau_L(\bar{x})$ will decline by at most $2x_h - 1$.

We now break into two further subcases. If $2x_h - 1 < \tau_L(\bar{x})/2$, then we will have $\tau_L(\bar{x}') > \tau_L(\bar{x})/2 \geq \psi(\bar{x})^2/4$ and so $\psi(\bar{x}') > \sqrt{\psi(\bar{x})^2/4} = \psi(\bar{x})/2$. If $2x_h - 1 \geq \tau_L(\bar{x})/2$, then $x_h > \tau_L(\bar{x})/4 \geq 12.5B^2$. But this means that

$$\frac{\tau_L(\bar{x}')}{\tau_L(\bar{x})} \geq \frac{(x_h-1)^2}{x_h^2} \geq \frac{\left(12.5B^2-1\right)^2}{\left(12.5B^2\right)^2} \geq \left(\frac{49}{50}\right)^2 \geq .96 \ .$$

Thus $\tau_L(\bar{x}') \geq .96\tau_L(\bar{x}) \geq .48\psi(\bar{x})^2$ and $\psi(\bar{x}') \geq \sqrt{.48\psi(\bar{x})^2} \geq .69\psi(\bar{x}) > \psi(\bar{x})/2$. Thus when $\tau_L(\bar{x}) \geq r_D(\bar{x})$ we have $\psi(\bar{x}') \geq \psi(\bar{x})/2$ in all cases.

Now suppose that $\tau_L(\bar{x}) < r_D(\bar{x})$, in which case $r_D(\bar{x}) \geq \psi(\bar{x})^2/2 \geq 50B^2$. Consider the bin in which the new item is placed. If its new level is a live level, then its original level must have been live as well. Thus $r_D(\bar{x})$ is unchanged, and we have $\psi(\bar{x}') \geq \sqrt{\psi(\bar{x})^2/2} \geq .707\psi(\bar{x}) > \psi(\bar{x})/2$.

The only case remaining is when $r_D(\bar{x}) \geq \psi(\bar{x})^2/2 \geq 50B^2$ and the new item increases the level of the bin that receives it to a dead-end level. Thus the count for one dead-end level increases by 1. Let us denote this level by h^+. If the item was placed in a bin with a live level, that is the only change in the dead-end level counts. Otherwise, an additional one of those counts (the one corresponding to the original level of the bin into which the item was placed) will decrease by 1. Let h^- denote this level if it exists.

In the terms of Lemma 3.13, let G be a minimum-arc graph that verifies $r_D(\bar{x})$ for $\bar{x}$. Let G' equal G if h^- does not exist or if $outdegree_G(h^-) < x_{h^-}$. Otherwise let G' be a graph obtained by deleting one of the out-arcs leaving h^- in G. In both cases, G will be applicable to $\bar{x}'$. Order the arcs of G so that the deleted arc (if it exists) comes last, preceded by all the other arcs out of h^- (if any exist), preceded by all the arcs into h^+ that are not out of h^- (if any exist), preceded by all remaining arcs, and let the arcs of G' occur in the same order as they do in G. Let us now see what happens when we apply G' to $\bar{x}'$, and how this differs from what happens when we apply G to $\bar{x}$.

Let $\delta(a)$ be the decrease in τ_D due to the application of arc a when G is being applied to $\bar{x}$, and let $\delta'(a)$ be the decrease when G' is applied to $\bar{x}'$. By Lemma

3.13 and the definition of τ_0 we have

$$r_D(\bar{x}) = \sum_{a \in G} \delta(a), \tag{3.17}$$

$$r_D(\bar{x}') \geq \sum_{a \in G'} \delta'(a). \tag{3.18}$$

Thus to complete the proof of Lemma 3.14, it will suffice to show that

$$\sum_{a \in G'} \delta'(a) \geq c \sum_{a \in G} \delta(a) \tag{3.19}$$

for an appropriate constant c.

Consider an arc $a = (i,j)$ in G and let $n_i(a)$ and $n_j(a)$ ($n'_i(a)$ and $n'_j(a)$) be the corresponding level counts when a is applied during the course of applying G to $\bar{x}$ (G' to $\bar{x}'$). By Lemma 1.2 and the fact that since i and j are dead-end levels neither can be 0 or B, we have $\delta(a) = 2(n_i(a) - n_j(a)) - 2$ and $\delta'(a) = 2(n'_i(a) - n'_j(a)) - 2$.

Let h be one of i, j. Observe that if $h \notin \{h^+, h^-\}$, then $n'_h(a) = n_h(a)$, if $h = h^+$ then $n'_h(a) = n_h(a) + 1$, and if $h = h^-$ then $n'_h(a) = n_h(a) - 1$. Thus the only arcs $a = (i,j)$ for which $\delta'(a) < \delta(a)$ are those with $i = h^-$, $j = h^+$, or both. If only one of the two holds, then $\delta'(a) = \delta(a) - 2$. If both hold then $\delta'(a) = \delta(a) - 4$. As a notational convenience, let A^* denote the set of deleted arcs. (Note that A^* will either be empty or contain a single arc.) Then we have

$$\sum_{a \in G'} \delta'(a) \geq \sum_{a \in G} \delta(a) - 2\left(indegree_G(h^+) + outdegree_G(h^-)\right) - \sum_{a \in A^*} \delta(a), \tag{3.20}$$

where $outdegree_G(h^-)$ is taken by convention to be 0 if h^- does not exist.

Let us deal with that last term first. If there is an arc $a^* = (i,j)$ in A^* then by our ordering of arcs in G it is the last arc. Suppose $\delta(a*) = 2(n_i(a) - n_j(a)) - 2 > 4$. Then we $n_i(a) - n_j(a) > 3$. But this means that after the arc is applied we will have $N_P(i) - N_P(j) \geq 2$, and so it would be possible to apply an additional arc (i,j), and this would further decrease τ_D by at least 2. But this contradicts our choice of G as a graph whose application to $\bar{x}$ yielded the maximum possible decrease in $\tau(\bar{x})$. So we can conclude that

$$\delta(a^*) \leq 4 \leq 2B. \tag{3.21}$$

Now let us consider the rest of the right hand side of (3.20). Let $M = indegree_G(h^+) + outdegree_G(h^-)$. If $M \leq 10B$, then

$$\sum_{a \in G} \delta(a) - \sum_{a \in G'} \delta'(a) \leq 2M + 2B \leq 22B \leq .11\psi(\bar{x})^2,$$

since by assumption $\psi(\bar{x})^2 \geq 100B^2 \geq 200B$. Thus by (3.17), (3.18), and our assumption that $r_D(\bar{x}) \geq \psi(\bar{x})^2/2$,

$$r_D(\bar{x}') \geq .39\psi(\bar{x})^2,$$

and hence $\psi(\bar{x}') \geq .624\psi(\bar{x}) > \psi(\bar{x})/2$.

Thus we may assume that $M > 10B$. Let A_h denote the multiset of arcs in G with $i = h^-$ or $j = h^+$ or both, and let us say that a pair $< i, j >$ of dead-end levels is a *valid pair* if A_h contains at least one arc (i,j). Note that there can be at most $B-1$ valid pairs, since by Lemma 3.13 no vertex in G can have both positive indegree and positive outdegree.

Suppose $< i,j >$ is a valid pair and there are m copies of arc (i,j) in A_h. By Lemma 3.13 each copy must decrease τ_D when it is applied, and so by Lemma 1.2 and the fact that i and j are dead-end levels, it must decrease by at least 2. If we let the last copy of (i,j) in our defined order be a_1, the next-to-last by a_2, etc., we will thus have $\delta(a_k) \geq 2$, $1 \leq k \leq m$. Moreover, since an application of an arc (i,j) reduces $N_P(i) - N_P(j)$ by 2, and since by Lemma 3.13 applications of other arcs cannot increase $N_P(i)$ or decrease $N_P(j)$, we must in fact have $\delta(a_k) \geq \delta(a_{k-1})+4$, $1 \leq k \leq m$. Thus

$$\sum_{k=1}^{m} \delta(a_k) \geq \sum_{k=1}^{m}(4k-2) = 2m^2.$$

Since there can be at most $B-1$ valid pairs, we thus have

$$\sum_{a\in A_h} \delta(a) \geq 2(B-1)\left\lfloor \frac{M}{B-1} \right\rfloor^2 > 2(B-1)\left(\frac{M}{B-1}-1\right)^2 > \frac{2M^2}{B-1} - 4M. \quad (3.22)$$

Then by (3.20), (3.21), (3.22), and our assumption that $M > 10B$, we have

$$\begin{aligned} \frac{\sum_{a\in G}\delta(a) - \sum_{a\in G'}\delta'(a)}{\sum_{a\in G}\delta(a)} &\leq \frac{2M+2B}{\frac{2M^2}{B-1} - 4M} = \frac{1+\frac{B}{M}}{\frac{M}{B-1}-2} \qquad (3.23)\\ &\leq \frac{1.1(B-1)}{M-2B+2} \leq \frac{1.1(B-1)}{8B+2} \leq \frac{1.1}{8} \leq .1375\ . \end{aligned}$$

Thus by (3.17) and (3.18) and our assumption that $r_D(\bar{x}) \geq \psi(\bar{x})/2$, we have

$$r_D(\bar{x}') > .86 r_D(\bar{x}) > .43\psi(\bar{x})^2$$

and hence $\psi(\bar{x}') \geq \sqrt{r_D(\bar{x}')} > .65\psi(\bar{x}) > \psi(\bar{x})/2$. Thus in all cases we have $\psi(\bar{x}') \geq \psi(\bar{x})/2$ and Lemma 3.14 is proved. ■

Returning to the proof that Hajek's Lemma applies, recall that we are in the midst of proving that the Bounded Variation Hypothesis holds, and are left with the task of showing that $\psi(\bar{x}) - \psi(\bar{x}') \leq 10B$ in the case where $\psi(\bar{x}') < \psi(\bar{x})$. By Lemma 3.6 it will suffice to show that $\psi(\bar{x})^2 - \psi(\bar{x}')^2 \leq 20B\psi(\bar{x}')$ when $\psi(\bar{x}) \geq 10B$, which by Lemma 3.14 will follow if we can show that

$$\psi(\bar{x})^2 - \psi(\bar{x}')^2 \leq 10B\psi(\bar{x}). \quad (3.24)$$

We divide the difference $\psi(\bar{x})^2 - \psi(\bar{x}')^2$ into two parts that we will treat separately: $\tau_L(\bar{x}) - \tau_L(\bar{x}')$ and $r_D(\bar{x}) - r_D(\bar{x}')$.

We begin by bounding the first part. If the item being packed goes in an empty bin, then a live level gets increased and no dead-end level is changed, so $\psi(\bar{x})$ *increases*, contrary to hypothesis. If the item being packed goes into a bin with a dead-end level, then $\tau_L(\bar{x})$ remains unchanged. If the item goes into a bin with a live level h, then by (3.7) we have that $x_h \leq \psi(\bar{x})$, so by Lemma 1.2 the decrease in τ_L is at most $2x_h - 1 < 2\psi(\bar{x}) \leq B\psi(\bar{x})$. Thus to prove (3.24) it will suffice to prove that $r_D(\bar{x}) - r_D(\bar{x}') \leq 9B\psi(\bar{x})$. We will in fact show that $r_D(\bar{x}) - r_D(\bar{x}') \leq 3B\psi(\bar{x})$.

To bound this second difference, note first that the hypotheses of Lemma 3.14 hold. So as in the proof of that Lemma, let G be a graph that verifies $r_D(\bar{x})$. If the placement of the item changes no dead-end level counts, there is nothing to prove, so we again may assume that there is a dead-end level h^+ that increases by 1 and (possibly) a dead-end level h^- that decreases by 1. As in the proof of the Lemma we have

$$r_D(\bar{x}) - r_D(\bar{x}') \leq 2\big(indegree_G(h^+) + outdegree_G(h^-)\big) + 2B, \tag{3.25}$$

where by convention $outdegree_G(h^-)$ is taken to be 0 if h^- does not exist.

Also, as in the proof of Lemma 3.14, there are at most $B-1$ distinct pairs $< i, j >$ such that (i, j) is an arc of G and $i = h^-$, $j = h^+$, or both. But then by Lemma 3.13(iii) we have fewer than $\psi(\bar{x})$ copies of each. Given that arcs (h^-, h^+) will be double counted in $indegree_G(h^+) + outdegree_G(h^-)$, we thus have

$$indegree_G(h^+) + outdegree_G(h^-) < B\psi(\bar{x}).$$

Combining this with (3.25) we conclude that

$$r_D(\bar{x}) - r_D(\bar{x}') \leq 2B\psi(\bar{x}) + 2B \leq 2B\psi(\bar{x}) + 2\psi(\bar{x})/10 < 3B\psi(\bar{x}).$$

We thus conclude (3.24) holds and hence so does the Bounded Variation Hypothesis.

To complete the proof that Hajek's Lemma applies, all that remains is to show that the Expected Decrease Hypothesis holds. Essentially the same proof that was used when there were no multiply-occurring dead-end levels will work, except that Lemma 3.7 needs to be modified to account for the possibility of such levels and we need to show that both it and Lemma 2.2 hold for $\psi(\bar{x})^2$.

This is straightforward for Lemma 2.2, which essentially says that assuming F is a perfectly packable distribution, the expected increase in $\phi(\bar{x})^2$ that can result from using SS to pack an item generated according to F is less than 2. This will hold for $\psi(\bar{x})^2$ as well since by definition

$$\begin{aligned}
\psi(\bar{x})^2 &= \tau_L(\bar{x}) + r_D(\bar{x}) \\
&= \tau_L(\bar{x}) + \tau_D(\bar{x}) - \tau_0(\bar{x}) \\
&= \phi(\bar{x})^2 - \tau_0(\bar{x}),
\end{aligned}$$

and by definition $\tau_0(\bar{x})$ can never decrease.

As to Lemma 3.7, we need only modify it by increasing the two key constants involved. The precise values of these constants are not relevant to satisfying the Expected Decrease Hypothesis. In particular, we can prove the following variant on Lemma 3.7.

LEMMA 3.15. *Let F be a bounded waste distribution and let P be any packing that can be created by applying SS to a list of items all of whose sizes are in U_F. If $\bar{x}$ is the profile of P and $\psi(\bar{x}) > 8B^2$, then there is a size $s \in U_F$ such that if an item of size s is packed by SS into P, the resulting profile $\bar{x}'$ satisfies*

$$\psi(\bar{x}')^2 \leq \psi(\bar{x})^2 - \frac{\psi(\bar{x})}{4B^2}.$$

PROOF. Since $\tau_0(\bar{x})$ can never decrease and $\psi(\bar{x})^2 = \tau_L(\bar{x}) + \tau_D(\bar{x}) - \tau_0(\bar{x})$, the result will follow if we can show that there exists an item size s such that if an item of size s is packed by SS, $ss(P) = \tau_L(\bar{x}) + \tau_D(\bar{x})$ will decline by at least $\psi(\bar{x})/(4B^2)$.

Suppose $\tau_L(\bar{x}) \geq \psi(\bar{x})^2/2$. Then as in the argument used in the proof of Lemma 3.7 there has to be a live level h with $x_h \geq \sqrt{\tau_L(\bar{x})/B} \geq \psi(\bar{x})/(\sqrt{2B})$ and hence a size s that will cause $ss(P)$ to decline by at least

$$2\left(\frac{x_h}{B} - 1\right) \geq 2\frac{\psi(\bar{x})}{\sqrt{2}B^{3/2}} - 2 > \frac{\psi(\bar{x}) + 8B^2}{\sqrt{2}B^{3/2}} - 2 > \frac{\psi(\bar{x})}{\sqrt{2}B^{3/2}} > \frac{\psi(\bar{x})}{4B^2}.$$

Suppose on the other hand that $\tau_L(\bar{x}) < \psi(\bar{x})^2/2$. In this case we must have $r_D(\bar{x}) > \psi(\bar{x})^2/2$. Let G be a minimum-arc reduction graph that verifies $r_D(\bar{x}) \geq \psi(\bar{x})^2/2$, and suppose G contains m arcs, ordered as $a_1, a_2, \ldots, a_m$. By Lemma 3.13(i),(iii), we know that $m < (B-1)\psi(\bar{x})$. Thus by Lemma 3.13(ii) we know that for some i, $1 \leq i \leq m$,

$$\Delta[i] \geq \frac{r_D(\bar{x})}{m} > \frac{\psi(\bar{x})^2}{2m} > \frac{\psi(\bar{x})^2}{2B\psi(\bar{x})} = \frac{\psi(\bar{x})}{2B},$$

where $\Delta[i]$ is defined to be the reduction in τ_D when the arc a_i is applied to the intermediate profile $\bar{y}[i-1]$, created by the application of earlier arcs in sequence to $\bar{x}$. Suppose arc $a_i = (h, j)$. Now by Lemma 3.13(i), the fact that h is the source of arc a_i means that it cannot have been a sink of a previous arc, so we must have $y[i-1]_h \leq x_h$. Similarly the fact that j is the sink of arc a_i means that it cannot be the source of any previous arc, so $y[i-1]_j \geq x_j$. But then the reduction in τ_D that would be obtained if a_i were applied directly to $\bar{x}$, i.e., if an item of size $j-h$ is placed in a bin of level h, is by Lemma 1.2

$$2(x_h - x_j - 1) \geq 2(y[i-1]_h - y[i-1]_j - 1) = \Delta[i] > \frac{\psi(\bar{x})}{2B}.$$

Unfortunately, there may not be an item of size $j-h$ in U_F. Lemma 3.13 only insures that there is a collection of items with sizes in U_F whose total size is $j-h$. But note that this collection can contain at most B items and that from the above we have $x_h - x_j > \psi(\bar{x})/(4B)$. Thus, as in the argument used in the proof of Lemma 3.7 and above, there must be a size $s \in U_F$ that will cause $\tau_D(\bar{x})$ and hence $ss(P)$ to decline by at least

$$2\left(\frac{\psi(\bar{x})}{4B^2} - 1\right) \geq \frac{\psi(\bar{x}) + 8B^2}{4B^2} - 2 = \frac{\psi(\bar{x})}{4B^2}. \quad \blacksquare$$

The remainder of the proof that Expected Decrease Hypothesis is satisfied by $\psi(\bar{x})$ proceeds just as the proof for $\phi(\bar{x})$ did when there were no multiply-occurring dead-end levels. Thus Hajek's Lemma applies and the upper bound of Theorem 3.11 is proved. $\blacksquare$

3.3.2 *Proof of the $\Omega(\log n)$ Lower Bound.* We begin the proof with a sequence of lemmas.

LEMMA 3.16. *Suppose s is a divisor of the bin size B. Then if an item of size s is placed into a packing P using SS, the value of $ss(P)$ can increase by at most 1.*

PROOF. All we need show is that there is *some* way of placing an item of size s so that $ss(P)$ increases by 1 or less, which will imply that SS must choose a placement that does no worse. If $s = B$ then placing an item of size s has no effect on $ss(P)$, so assume $s < B$. If there is no bin of level s, then starting a new bin with an item of size s will increase $ss(P)$ by 1, so assume $N_P(s) > 0$ and let $h_s = \max\{h : s|h \text{ and } N_P(h) > 0\}$. If $h_s = B - s$, then placing our item in a bin with this level will decrease $ss(P)$, so assume $h_s \leq B - 2s$. By Lemma 1.2, placing an item of size s in one of the bins with level h_s increases $ss(P)$ by at most $2(N_P(h_s + s) - N_P(h_s)) + 2 \leq 0$. Thus in every case there is a way to increase $ss(P)$ by 1 or less, as required. ■

Let us say that a level h is *divisible for F* if any set of items with sizes in U_F that has total size h can contain only items whose sizes are divisors of B.

LEMMA 3.17. *If h is a multiply-occurring dead-end level for F then h is not divisible for F.*

PROOF. Let $\mathcal{H}$ be the set of all levels i, $1 \leq i \leq B-1$, that are divisible for F and assume, for the sake of contradiction, that $h \in \mathcal{H}$. Since h is a multiply-occurring dead-end level for F, there is some list L that under SS yields a packing containing at least two bins with level h. Consider the first time during the packing of L that a level $i \in \mathcal{H}$ had its count $N_P(i)$ increase from 1 to 2, and let s be the size of the item x whose placement caused this to happen. By definition of *divisible level*, s must be a divisor of B, and so by Lemma 3.16, the placement of x can have increased $ss(P)$ by at most 1. But this is impossible: If $i = s$ then the insertion of x would have increased $ss(P)$ by $2^2 - 1^2 = 3$. On the other hand, suppose $i > s$. Since i is a divisible level, so is $i - s$. Thus $N_P(i - s) = N_P(i) = 1$ just before x was packed: Neither count can exceed 1 by our choice of i, the latter must be 1 if it is to increase to 2 after the placement of x, and the former must be 1 since x can only create a bin with level i if there is a bin of level $i - s$ into which it can be placed. However, this means that $ss(P)$ increases by 2, contradicting Lemma 3.16. So $h \notin \mathcal{H}$, as desired. ■

LEMMA 3.18. *Suppose s is an item size that does not evenly divide the bin capacity B and we are asked to pack an arbitrarily long sequence of items of size s using SS. Let $d_i = is$, $0 \leq i \leq \lfloor B/s \rfloor$. For all $m > 0$, the packing in existence just before the first time $N_P(d_1) > m$ must have $N_P(d_i) = mi$ for every d_i.*

PROOF. Let us say that mi is the *target* for level d_i. We first show that it must be the case that $N_P(d_i)$ is no more than its target, $1 < i \leq \lfloor B/s \rfloor$, so long as $N_P(d_1)$ has never yet exceeded its target. Suppose not, and consider the packing just before the first one of these counts, say $N_P(d_i)$, exceeded its target. In this packing we must have $N_P(d_i) = mi$. Let $\Delta_h = N_P(d_h) - N_P(d_{h-1})$, $1 \leq h \leq \lfloor B/s \rfloor$, where by convention $N_P(0) = 0$ and so $\Delta_1 = N_P(d_1)$. Now note that if $\Delta_i \geq \Delta_1$, then, by Lemma 1.2, starting a new bin will cause $ss(P)$ to increase by less that placing an item of size s in a bin with level d_{i+1}. Since Δ_1 by hypothesis is m or less, this implies that $\Delta_i \leq m - 1$. But then we must have $N_P(d_{i-1}) \geq (i-1)m + 1$, contradicting our assumption that level d_i was the first to have its count exceed its target.

For the lower bound, note that in the packing just before $N_P(d_1)$ first exceeds m, it must be the case that $\Delta_1 = N_P(d_1) = m$. Since this was the preferred move under SS, it must be the case by Lemma 1.2 that $\Delta_i \geq m$, $2 \leq i \leq \lfloor B/s \rfloor$. The result follows. ■

LEMMA 3.19. *Suppose F is a fixed discrete distribution with at least one multiply-occurring dead-end level h and H is a positive constant. Then there is a list L_H of length $O(H)$ consisting solely of items with sizes in U_F, such that the packing resulting from using SS to pack L_H contains at least H bins with level h.*

PROOF. By Lemma 3.17 there must be a set $S = \{x_0, x_1, \ldots, x_t\}$ of items with sizes in U_F whose total size is h, and for which $s(x_0)$ is not a divisor of B. Let us also arrange the indexing so that all items of any given size appear contiguously in the sequence $s(x_0), s(x_1), \ldots, s(x_t)$. Note that we may assume that $s(x_i) \geq 2$, $0 \leq i \leq t$, since if 1 were in U_F there could be no dead-end levels. Let $h_i = \sum_{j=0}^{i} s(x_j)$, $0 \leq i \leq t$. Note that $h_t = h$. Further, let $k = \lfloor B/s(x_0) \rfloor$.

Our list L_H will consist of a sequence of $t+1$ (possibly empty) segments, the first of which (Segment 0) consists of $H3^t \sum_{i=1}^{k} i^2 = H3^t k(k+1)(2k+1)/6$ items of size $s(x_0)$. In the packing P obtained by using SS to pack these items, we will have by Lemma 3.18 that level $i \cdot s_0$ will have count $iH3^t$, $1 \leq i \leq k$, and in particular level $h_0 = s(x_0)$ will have count $H3^t$. In what follows we use "P" generically to denote the current packing. Note that after Segment 0 has been packed, P contains $H3^t \sum_{i=1}^{k} i = H3^t k(k+1)/2$ partially filled bins.

Segment 1 consists of the shortest possible sequence of items of size $s(x_1)$ that, when added to P using SS, will cause the count for level $h_1 = s(x_0) + s(x_1)$ to equal or exceed $H3^{t-1}$. A sequence of this sort must exist for the following reasons: If $N_P(h_1)$ is itself $H3^{t-1}$ or greater, as for instance it would be if $s(x_1) = s(x_0)$, then the empty segment will do. Otherwise, suppose $N_P(h_1) < H3^{t-1}$. So long as $N_P(h_0) \geq 2H3^{t-1}$ and $N_P(h_1) < H3^{t-1}$, placing an item of size $s(x_1)$ in a bin with level h_0 would cause a greater reduction in $ss(P)$ than placing it in a bin of level h_1 could, and so would be the preferred move. Since we can place $H3^{t-1}$ items in bins of level h_0 before $N_P(h_0) \leq 2H3^{t-1}$, and each such placement would increase $N_P(h_1)$ by 1, this means we will eventually have placed enough to increase $N_P(h_1)$ to the desired target. Note that we will eventually be forced to place items in bins of level h_0 rather than some level other than h_0 or h_1, since the existence of moves that decrease $ss(P)$ means that no new bins are being created.

We complete our argument by induction. In general, we start Segment j, $2 \leq j \leq t$ with a packing in which $N_P(h_{j-1}) \geq H3^{t-j+1}$ and no new bins have been created since Segment 0. The segment then consists of the shortest possible sequence of items of size $s(x_j)$ that will cause the count for level $h_j = h_{j-1} + s(x_j)$ to equal or exceed $H3^{t-j}$. An argument analogous to that for Segment 1 says that this must eventually occur without any additional bins being started. Thus at the end of Phase t we have H bins with level $h_t = h$. Given that all the $s(x_j)$ are 2 or greater and no new bins were started after Segment 0, the total number of items in our overall list L_H is at most

$$\frac{H3^t k(k+1)}{2} \cdot \frac{B}{2} < B^3 3^B H = O(H)$$

for fixed F, as required. ■

For future reference, note that since all the $s(x_j)$ are 2 or greater, the number of segments in L_H is no more than $B/2$.

LEMMA 3.20. *Suppose P and Q are two packings for which*

$$|P - Q| \equiv \sum_{h=1}^{B-1} |N_P(h) - N_Q(h)| = M$$

and L is a list consisting entirely of items of the same size $s \geq 2$. Then the packings P' and Q' resulting from using SS to pack L into P and Q satisfy $|P' - Q'| \leq BM$.

PROOF. We prove the lemma for the special case of $M = 1$. The general result then follows by introducing a sequence of packings $P = R_0, R_1, \ldots, R_M = Q$ with $|R_i - R_{i-1}| = 1$, $1 \leq i \leq M$, and applying the $M = 1$ case to each such pair. So assume $|P - Q| = 1$.

Let g denote the level that has different counts under P and Q and suppose without loss of generality that $N_P(g) = N_Q(g) + 1$. Let P_i and Q_i denote the packings that result after the first i items of L have been packed into P and Q respectively. We will say that a triple (i, j, ℓ), $0 \leq i, j \leq |L|$ and $0 \leq \ell \leq B$, is a *compatible* triple if either

(1) $P_i = Q_j$ and $\ell \in \{0, B\}$, or
(2) $|P_i - Q_j| = 1$, and ℓ is the unique bin level such that $1 \leq \ell \leq B - 1$, $N_{P_j}(\ell) = N_{Q_i}(\ell) + 1$.

Note that by this definition $(0, 0, g)$ is a compatible triple.

CLAIM 3.21. *If (i, j, ℓ) is a compatible triple with $i, j < |L|$ then one of the following three triples must also be compatible:*

$$(i+1, j+1, \ell), \quad (i+1, j, \ell+s), \quad (i, j+1, \ell-s).$$

PROOF OF CLAIM. Consider the packings P_i and Q_j. Suppose SS would place an item of size s in bins with the same level in both P_i and Q_j. Then the same bins counts would be changed in the same way for P_i and Q_j and so $(i+1, j+1, \ell)$ would be a compatible triple.

Otherwise suppose SS would place an item of size s in bin h_P for P_i and in h_Q for Q_j, with $h_P \neq h_Q$. In this case P_i and Q_j must be different, and we are in case 2 of compatibility. Let $\Delta_Q(h)$ (resp. $\Delta_P(h)$) denote the net reduction in the sum of squares if an item of size s is placed in a bin of level h in Q_j (resp. P_i), assuming such a placement is legal. Since the bin counts $N_{P_i}(h)$ and $N_{Q_j}(h)$ are equal for every h other than ℓ, it follows that $\Delta_P(h) = \Delta_Q(h)$ for all h's other than ℓ and $\ell - s$. Since SS makes different choices for P_i and for Q_j, it must be that at least one of h_Q, h_P is either ℓ or $\ell - s$. We now show that $h_P = \ell$ or $h_Q = \ell - s$ by showing that the other two options are impossible.

By hypothesis we have $N_{P_i}(\ell) = N_{Q_j}(\ell) + 1$ and all other counts are equal, so $\Delta_P(\ell - s) < \Delta_Q(\ell - s)$ (if $\ell - s \geq 0$), $\Delta_P(\ell) > \Delta_Q(\ell)$ (if $\ell + s \leq B$), and for all other values of h, $\Delta_P(h) = \Delta_Q(h)$. Suppose $h_P = \ell - s$. Then we must have $\Delta_Q(\ell - s) > \Delta_P(\ell - s) \geq \Delta_P(h)$ for all $h \neq \ell - s$. But $\Delta_Q(h) = \Delta_P(h)$ for all $h \notin \{\ell - s, \ell\}$ and $\Delta_Q(\ell) < \Delta_P(\ell)$. Thus we must have $h_Q = \ell - s = h_P$ contradicting our assumption that they are unequal. Similarly, if $h_Q = \ell$ then we must have $h_P = \ell$, again a contradiction. Thus either $h_P = \ell$ or $h_Q = \ell - s$.

In the first case, $h_P = \ell$, we must have $\ell + s \leq B$. Packing an item of size s into a bin with level ℓ in P_i reduces $N_{P_i}(\ell)$ by 1, so that $N_{P_{i+1}}(\ell) = N_{Q_j}(\ell)$. If $\ell + s = B$, i.e., we fill up a bin, then $|P_{i+1} - Q_j| = 0$, and so $(i+1, j, \ell+s = B)$ is a compatible triple. If $\ell + s < B$ then $N_{P_i}(\ell + s)$ will increase by 1 and we will have $N_{P_{i+1}}(\ell+s) = N_{P_i}(\ell+s) + 1 = N_{Q_j}(\ell+s) + 1$, while all other levels now have the same counts. Thus $(i+1, j, \ell+s)$ is again a compatible triple.

In the second case, $h_Q = \ell - s$, we must have $\ell - s \geq 0$. Packing an item of size s into a bin with level $\ell - s$ in Q_j increases $N_{Q_j}(\ell)$ by 1, so that $N_{P_i}(\ell) = N_{Q_{j+1}}(\ell)$. If $\ell - s = 0$, i.e. we pack s into a new bin, then $|P_i - Q_{j+1}| = 0$, and so $(i, j+1, \ell - s = 0)$ is a compatible triple. If $\ell - s > 0$ then $N_{Q_j}(\ell - s)$ will decrease by 1 and we will have $N_{Q_{j+1}}(\ell - s) = N_{Q_j}(\ell - s) - 1 = N_{P_i}(\ell - s) - 1$, while all other levels now have the same counts. Thus $(i, j+1, \ell - s)$ is again a compatible triple.

This completes the proof of the Claim. ∎

Given the Claim and the fact that $(0, 0, g)$ is a compatible triple, we have by induction that at least one of the three following scenarios must hold:

(1) $(|L|, |L|, g)$ is a compatible triple, or
(2) There is an integer a, $1 \leq a \leq (B - g)/s$ such that $(|L|, |L| - a, g + as)$ is a compatible triple, or
(3) There is an integer b, $1 \leq b \leq g/s$ such that $(|L| - b, |L|, g - bs)$ is a compatible triple.

In the first case we have $|P' - Q'| = 1$, which clearly satisfies the Lemma's conclusion. In the second we have $|P' - Q_{|L|-a}| = 1$, but to get Q' from $Q_{|L|-a}$ we will need to add a additional items of size s, and each addition will change one or two level counts by 1. Since $s \geq 2$ and $g \geq 1$, we must have $a \leq (B - g)/s \leq (B - 1)/2$. Thus we can conclude that $|P' - Q'| \leq 1 + B - 1 = B$ as desired. The third case follows analogously, except now we use the fact that $g \leq B - 1$, and the Lemma is proved. ∎

Lemma 3.22. *Suppose F is a fixed discrete distribution with at least one multiply-occurring dead-end level and X is a positive constant. Then for any $D > X$ there is a list $L_{X,D}$ of length $O(D)$ consisting solely of items with sizes in U_F, such that for any packing P with no live-level count exceeding X, the packing Q resulting from using SS to add $L_{X,D}$ into P contains at least D bins with dead-end levels.*

Proof. We may assume that P contains fewer than D bins with dead-end levels, because the number of bins with dead-end levels can never decrease and if we already had D such bins any list will do for $L_{X,D}$. Let h be a multiply-occurring dead-end level for F. For our list we simply let $L_{X,D}$ be the list L_H derived for h using Lemma 3.19, with $H = (XB + D)B^{B/2} + D = O(D)$ for fixed F. By Lemma 3.19 the length of L_H will be $O(H) = O(D)$.

If P_0 denotes the empty packing, we know by Lemma 3.19 that if SS is used to pack L_H into P_0 it will create a packing P_0' with at least H bins having the dead-end level h. Let P' denote the packing that would result if we used SS to add

L_H to P. Note that $|P - P_0| = \sum_{i=1}^{B-1} N_P(i) \leq X(B-1)+D-1$. Thus by applying Lemma 3.20 once for each segment of L_H and using the fact that L_H contains no more than $B/2$ segments, we have that $|P' - P_0'| \leq B^{B/2}(XB+D)$. But this means that for dead-end level h we must have $N_{P'}(h) \geq H - B^{B/2}(XB+D) = D$ and so P' contains at least the desired number of bins with dead-end levels. ■

LEMMA 3.23. *Let P_N be the packing after N items generated according to F have been packed by SS. There is a constant X, depending only on F, such that for any $N > 0$*

$$p[N_{P_N}(i) \leq X \text{ for all live levels } i] \geq \frac{1}{2}. \tag{3.26}$$

PROOF. Recall from the inequality (3.8) of the proof of the $O(\log n)$ upper bound on the expected waste of SS that for any $N > 0$, if X_N is the profile after packing N items, then there are a constants c and T, depending only on F, such that

$$E\left[c^{\psi(X_N)}\right] \leq T.$$

This means that

$$p\left[c^{\psi(X_N)} > 2T\right] \leq \frac{1}{2}$$

and hence that

$$p\left[\psi(X_N) > \log_c(2T)\right] \leq \frac{1}{2}.$$

Since as we have repeatedly observed $\psi(\bar{x}) \geq x_h$ for every live level h, this in turn means that the probability is at least $1/2$ that no live level count exceeds $\log_c(2T)$. Thus the Lemma holds with $X = \log_c(2T)$. ■

We are now in a position to prove our $\Omega(\log n)$ lower bound on $EW_n^{SS}(F)$ when F has multiply-occurring dead-end levels. Consider the lists $L_{X,D}$ specified by Lemma 3.22 for the value of X given by Lemma 3.23, and let ℓ_D denote the length of $L_{X,D}$. Since the value of X depends only on F, Lemma 3.22 implies that there is a constant d, depending only on F, such that for all $D > X$, $\ell_D < dD$.

Now suppose we have a random list L of length dD of items generated according to F. The probability that $L_{X,D}$ is a prefix of L is at least ϵ^{ℓ_D}, where $\epsilon = \min\{p_j : 1 \leq j \leq J\}$. Let $a = \log_2(1/\epsilon)$. Then the probability that $L_{X,D}$ is *not* a prefix of L is at most $(1-(1/2)^{adD})$.

Now consider a random list L^* of length $dD2^{adD}$, viewed as a sequence of 2^{adD} random segments of length dD. The probability that none of these segments has $L_{X,D}$ as a prefix is

$$\left(1 - \frac{1}{2^{adD}}\right)^{2^{adD}} < \frac{1}{e} < \frac{1}{2}.$$

In other words, the probability that at least one of these segments has $L_{X,D}$ as a prefix exceeds $1/2$. Consider the *last* segment that has $L_{X,D}$ as a prefix (should any such segments exist), and the packing P that exists just before this copy of $L_{X,D}$ is packed. Note that by choosing the last such segment, we do not condition in any way the list that precedes this copy or the packing P. Hence by Lemma 3.23, with probability at least 1/2 the packing P has no live level count exceeding X, and by Lemma 3.22, after the segment is added to the packing, the new packing (and all subsequent ones) will contain at least D bins with dead-end levels. Thus the expected number of bins with dead-end levels after all of L^* is packed is at least $(1/2)(1/2)D = D/4 = \Omega(\log |L^*|)$. The lower bound follows. ■

4. SS AND LINEAR WASTE DISTRIBUTIONS

The implication of Theorem 2.5 that $ER_\infty^{SS}(F) = 1$ for all perfectly packable distributions F unfortunately does not carry over to the case where $EW_n^{OPT} = \Theta(n)$.

THEOREM 4.1. *There exist distributions F_k, $1 \le k \le \infty$, such that*

$$\limsup_{k\to\infty} ER_\infty^{SS}(F_k) = 1.5\ .$$

PROOF. Let F_k be the distribution in which the bin size is $B = 2k+1$ and the single item size 2 occurs with probability 1. Consider an n-item list L_n generated according to F_k where n is divisible both by k and by $\sum_{i=1}^k i^2 = k(k+1)(2k+1)/6$. Then $OPT(L_n) = n/k$ and by Lemma 3.18, we have

$$SS(L_n) = \left(\frac{n}{\sum_{i=1}^k i^2}\right)\left(\sum_{i=1}^k i\right) = n \cdot \frac{\left(\frac{k(k+1)}{2}\right)}{\left(\frac{k(k+1)(2k+1)}{6}\right)} = \frac{3n}{2k+1}.$$

Thus $ER_\infty^{SS}(F_k)$, which is defined as a lim sup, equals $3k/(2k+1)$ and the Theorem follows. ■

We conjecture that 3/2 is the worst possible value for $ER_\infty^{SS}(F)$ over all discrete distributions F, although at present the best upper bound we can prove is 3, which is implied by the following worst-case result.

THEOREM 4.2. *For all lists L, $SS(L) \le 3\lceil s(L)/B\rceil \le 3OPT(L)$.*

PROOF. Let x be the last item of size less than $B/3$ that starts a new bin and let s be the size of x. (If no such x exists, then all bins are at least $B/3$ full in the final packing and we are done.) Let P be the packing just before x was packed. It is sufficient to show that the average bin content in the bins of P is at least $B/3$. If that is so, then the packing of subsequent items cannot reduce the average bin content in the bins not containing x to less than $B/3$. Consequently if m is the final number of bins in the packing, we must have $s(L)/B > (m-1)/3$ and hence $OPT(L) \ge \lceil s(L)/B\rceil \ge m/3$ and the theorem follows.

So let us show that the average bin content in the bins of P is at least $B/3$. For $1 \le j \le s$, let ℓ_j be the greatest integer such that $j + \ell_j s < B$ and let Ω_j denote the set of bins with contents $j, j+s, \ldots, j+\ell_j s$. Note that $\Omega_1, \ldots, \Omega_s$ is a partition of the bins of P into s sets, and if we can show that the average contents of the bins in each nonempty Ω_j is at least $B/3$, we will be done. Fix j and suppose k is the least integer such that either $N_P(j+ks) > 0$ or $j+ks \ge B/3$. If $j+ks \ge B/3$ then *every* bin in Ω_j has contents at least $B/3$ and so Ω_j behaves as desired. So suppose $j+ks < B/3$, in which case we must have $k \le \ell_j - 2$. Since SS places x

in a new bin, we must by Lemma 1.2 have

$$0 \le N_P(s) \le N_P(j+hs+s) - N_P(j+hs), \quad h = k, \ldots, \ell_j - 1$$

and hence $N_P(j+ks) \le N_P(j+ks+s) \le \cdots \le N_P(j+\ell_j s)$. This means that if we let $t = j+ks$ the average contents of the bins in Ω_j is at least

$$\begin{aligned}
\frac{t+(t+s)+\cdots+(t+(\ell_j-k)s)}{\ell_j-k+1} &= \frac{t(\ell_j-k+1)+s(\ell_j-k+1)(\ell_j-k)/2}{\ell_j-k+1} \\
&= t + \frac{(\ell_j-k)s}{2} \\
&= j+ks+\frac{(\ell_j-k)s}{2} \\
&> \frac{j+\ell_j s}{2} \ge \frac{B-s}{2} > \frac{B}{3}. \quad \blacksquare
\end{aligned}$$

Theorem 4.2 raises the question what the actual worst-case behavior of SS is. If we define the asymptotic worst-case ratio R_∞^A for an algorithm A to be

$$\limsup_{n\to\infty} \max\left\{\frac{SS(L)}{OPT(L)} : L \text{ is a list with } OPT(L) = n\right\}$$

the theorem implies that $R_\infty^{SS} \le 3$. This bound has recently been lowered to 2.7777... [Csirik et al. 2005], but that is still a significantly worse asymptotic bound than we have for the classic online heuristics as First Fit and Best Fit, both of which have $R_\infty^A = 1.7$ [Johnson et al. 1974]. Moreover, although the upper bound on R_∞^{SS} is not tight, SS is definitely worse than First and Best Fit in the above asymptotic worst-case sense:

THEOREM 4.3. $R_\infty^{SS} \ge 2$.

PROOF. This lower bound is proved by exhibiting lists L_m for each odd $m > 0$ satisfying $OPT(L_m) = (m+1)/2$ and $SS(L_m) = m$. List L_m has bin size $B = 2m+1$ and consists of m items with the sizes $2m$, $2m-2$, $2m-4$, ..., 6, 4, 2, in that order. SS will place each in a separate bin, whereas an optimal packing would combine $2(m-i)$ with $2i$, $1 \le i < m/2$. $\blacksquare$

Note that the above examples yield only one optimum value for each value of B and so are not asymptotic for fixed B. If one defines

$$R_\infty^A(B) = \limsup_{n\to\infty} \max\left\{\frac{SS(L)}{OPT(L)} : L \text{ is a list for bin size } B \text{ with } OPT(L) = n\right\}$$

then the we know of no examples for any fixed B that show that $R_\infty^{SS}(B)$ exceeds the lower bound of roughly 1.54 that van Vliet [1992] showed holds for *all* online algorithms A.

5. IDENTIFYING PERFECTLY PACKABLE DISTRIBUTIONS

Given the observations of the previous section, it would be valuable to be able to identify those distributions F that satisfy the hypotheses of Theorem 2.5, i.e., those for which $EW_n^{OPT}(F) = O(\sqrt{n})$ and hence $ER_\infty^{SS}(F) = 1$ is guaranteed. This task is unfortunately NP-complete, as it would require us to solve the PARTITION problem [Garey and Johnson 1979]. Fortunately, however, the problem is not NP-complete in the strong sense, and as we shall now see, can be solved in time pseudo-polynomial in B via linear programming, as was claimed but not proved in Csirik et al. [1999].

Suppose our discrete distribution is as described above, with a bin capacity B, integer item sizes $s_1, s_2, \ldots, s_J$, and positive rational probabilities $p_1, p_2, \ldots, p_J$. Our linear program, which for future reference we shall call the "Waste LP for F," will have JB variables $v(j,h)$, $1 \le j \le J$ and $0 \le h \le B-1$, where $v(j,h)$ represents the rate at which items of size s_j go into bins whose current level is h. The constraints are:

$$v(j,h) \ge 0, \quad 1 \le j \le J, \quad 0 \le h \le B-1, \tag{5.1}$$

$$v(j,h) = 0, \quad 1 \le j \le J, \quad s_j > B-h, \tag{5.2}$$

$$\sum_{h=0}^{B-1} v(j,h) = p_j, \quad 1 \le j \le J, \tag{5.3}$$

$$\sum_{j=1}^{J} v(j,h) \le \sum_{j=1}^{J} v(j,h-s_j), \quad 1 \le h \le B-1. \tag{5.4}$$

where by convention the value of $v(j,h-s_j)$ when $h - s_j < 0$ is taken to be 0 for all j. Constraints (5.2) say that no item can go into a bin that is too full to have room for it. Constraints (5.3) say that all items must be packed. Constraints (5.4) say that bins with a given level are created at least as fast as they disappear. The goal is to minimize

$$c(F) \equiv \sum_{h=1}^{B-1} \left((B-h) \cdot \left(\sum_{j=1}^{J} v(j,h-s_j) - \sum_{j=1}^{J} v(j,h) \right) \right). \tag{5.5}$$

Let $ES(F)$ denote the expected item size $\sum_{j=1}^{J} s_j p_j$.

Lemma 5.1. *Suppose F is a discrete distribution and let $L_n(F)$ be a random n-item list generated according to F.*

(1) *For all $n > 0$,* $\left| EW_n^{OPT}(F) - \dfrac{nc(F)}{B} \right| \le O(\sqrt{n})$.

(2) *There exist constants b and N^* such that for all $n \ge N^*$*

$$P\left[\left| OPT(L_n(F)) - \frac{n}{B}\Big(ES(F) + c(F)\Big) \right| > bn^{2/3} \right] \le \frac{1}{n^{1/6}} \ .$$

This lemma, which we shall prove shortly, implies the following three results.

Theorem 5.2. *Suppose F is a discrete distribution. Then*

$$\lim_{n\to\infty} \left(\frac{EW_n^{OPT}(F)}{n} \right) = \frac{c(F)}{B} \ .$$

Theorem 5.3. *Suppose F is a discrete distribution. Then $EW_n^{OPT}(F) = O(\sqrt{n})$ if and only if $c(F) = 0$.*

LEMMA 5.4. *Suppose F is a discrete distribution and A is a (possibly randomized) bin packing algorithm for which $A(L)/OPT(L) \le d$ for some fixed constant d and all lists L. Then*

$$ER_\infty^A(F) = \frac{ES(F) + B \cdot \limsup_{n\to\infty} EW_n^A(F)/n}{ES(F) + c(F)} .$$

Theorems 5.3 and 5.2 are immediate consequences of Claim 1 of Lemma 5.1. Lemma 5.4 follows from Claim 2. Basically, it says that $ER_\infty^A(F)$, which is defined in terms of expected ratios, can actually be computed in terms of ratios of expectations. It follows because Claim 2 implies that we can divide the set of lists L of length n generable according to F into two sets. For the first set, which has cumulative probability $1 - 1/n^{1/6}$, we have

$$E\left[\frac{A(L)}{OPT(L)}\right] = \left(\frac{nES(F) + B \cdot EW_n^A(F)}{nES(F) + nc(F)}\right)\left(1 + O\left(\frac{1}{n^{1/3}}\right)\right). \tag{5.6}$$

For the second set, which has cumulative probability $1/n^{1/6}$, $E[A(L)/OPT(L)] \le d$. Thus this set contributes at most $d/n^{1/6}$ to the overall expected ratio for $L_n(F)$, meaning that (5.6) holds with L replaced by $L_n(F)$ and $1/n^{1/3}$ replaced by $1/n^{1/6}$. Lemma 5.4 follows. We now turn to the proof of Lemma 5.1.

PROOF OF LEMMA 5.1. Consider the values $v(j,h)$ of the variables in an optimal basic solution to the LP. Since all the coefficients and right-hand sides of the LP are rational, all these variable values must be rational as well, and there exists a positive integer N such that $Nv(j,h)$ is an integer, $1 \le j \le J$ and $0 \le h \le B-1$. For each positive integer k, let L_k be a list consisting of $k\sum_{h=0}^{B-1} Nv(j,h)$ items of size s_j, $1 \le j \le J$. By (5.3) L_k will contain kNp_j items of size s_j for each j, for a total of kN items. We will thus have $s(L_k) = kN \cdot ES(F)$.

Note that we can construct a packing of L_k simply by following the instructions provided by the variable values in the solution to the LP. That is, for each j, start $kNv(j,0)$ bins by placing an item of size s_j into an empty bin. By (5.4), the number of bins of level 1 will now be at least $\sum_{j=1}^{J} kNv(j,1)$. Thus we can take a set consisting of $kNv(j,1)$ items of size s_j, $1 \le j \le J$, and place each of these items in a distinct bin with level 1. We can now proceed to pack bins of level 2, and so on. Let P_k denote the resulting packing.

How many bins does this packing contain? A bin in P_k that has level h contains items of total size h by definition, and in addition has a gap of size $B-h$. Thus the total number of bins is simply the $1/B$ times the sum of the item sizes and the sum of the gap sizes, that is

$$\frac{1}{B}\left(kN \cdot ES(F) + kN\sum_{h=1}^{B-1}\left((B-h)\cdot\left(\sum_{j=1}^{J} v(j,h-s_j) - \sum_{j=1}^{J} v(j,h)\right)\right)\right)$$

and hence

$$|P_k| \quad = \quad \left(\frac{kN}{B}\right)\Big(ES(F) + c(F)\Big). \tag{5.7}$$

Note for future reference that since P_k has at least one item per bin, we have $|P_k| \le kN$ and hence (5.7) implies $ES(F) + c(F) \le B$ and $c(F) \le B - 1$.

Now, since L_k is in essence the "expected value" of the random list $L_{kN}(F)$, we can use the packings P_k as models for packing the random lists $L_n(F)$, $n > 0$. We proceed as follows: Let L be a specific list of n items generated according to F. Let $k \ge 0$ be such that $kN \le n < (k+1)N$. Now note that the packing P_k has kNp_j "slots" for items of size s_j, $1 \le j \le J$, and L is expected to have between kNp_j and $(k+1)Np_j$ such items. Let n_j denote the number of items of size s_j among the first kN items of L and define

$$\Delta_j^+ = \max\{0, n_j - kNp_j\},\ 1 \le j \le J,$$
$$\Delta_j^- = \max\{0, kNp_j - n_j\},\ 1 \le j \le J.$$

Thus Δ_j^+ is the *oversupply* of items of size s_j among the first kN items, Δ_j^- is the *shortfall*, and $\sum_{j=1}^J \Delta_j^+ = \sum_{j=1}^J \Delta_j^-$. Let X_n denote $n - kN + \sum_{j=1}^J \Delta_j^+$. We claim that

$$\left| OPT(L) - \frac{n}{B}\big(ES(F) + c(F)\big) \right| \quad \le \quad X_n. \tag{5.8}$$

We first show that $OPT(L) \le (n/B)(ES(F) + c(F)) + X_n$: Starting with our model packing P_k, place as many items of L into the appropriate slots as possible, and then place the leftover items in additional bins, one per bin. The total number of bins used will then be at most $|P_k|$ plus the number of additional bins. This number is at most the oversupply of items among the first kN of L, i.e., $\sum_{j=1}^J \Delta_j^+$, plus the number of items after the first kN in L that do not go into slots, which can clearly be no more than $n - kN$. We thus have

$$OPT(L) \le |P_k| + X_n = \frac{kN}{B}\big(ES(F) + c(F)\big) + X_n \le \frac{n}{B}\big(ES(F) + c(F)\big) + X_n.$$

A slightly more complicated argument implies $(n/B)(ES(F)+c(F)) \le OPT(L)+X_n$: First observe that the packing P_k defined above for L_k must be an optimal packing for L_k. If not, i.e., if $OPT(L_k) \le (kN/B)(ES(F) + c(F))$, then we could use an optimal packing for L_k to define a better solution to our LP, contradicting our assumption that $c(F)$ was the optimal solution value for the LP. Next observe that if we are given a packing P for L, we can construct a closely related one for L_k by a process of addition: For each of the at most $\sum_{j=1}^J \Delta_j^- = \sum_{j=1}^J \Delta_j^+$ items in L_k that do not have counterparts of the same size in L_n, we add a new bin to P containing just that item. This new packing contains at least as many items of each size as does L_k and so must contain at least $OPT(L_k)$ bins. Thus we have

$$|P_k| = OPT(L_k) \le OPT(L) + \sum_{j=1}^J \Delta_j^+.$$

Now observe that by (5.7) we have

$$\frac{n}{B}\bigg(ES(F) + c(F)\bigg) \le |P_k| + \frac{n - kN}{B}\bigg(ES(F) + c(F)\bigg) \le |P_k| + n - kN$$

since as previously observed $ES(F) + c(F) \le B$. Combining these last two inequalities we obtain $(n/B)(ES(F) + c(F)) \le OPT(L) + X_n$ as claimed.

Thus (5.8) holds, and our remaining task is to estimate $X_n = n - kN + \sum_{j=1}^{J} \Delta_j^+$. Since each n_j is a sum of independent Bernoulli variables when considered by itself, we have $E[\Delta_j^+] \le \sqrt{kNp_j(1-p_j)} < \sqrt{kNp_j}$. Given that $\sum_{j=1}^{J} \sqrt{kNp_j}$ is maximized when all the probabilities are equal, we have that $E[\sum_{j=1}^{J} \Delta_j^+] \le J\sqrt{kN/J} \le \sqrt{nJ}$. Since J and N are constants, we thus have

$$E[X_n] = n - kN + E\left[\sum_{j=1}^{J} \Delta_j^+\right] \le N + \sqrt{nJ} = O(\sqrt{n}). \tag{5.9}$$

Claim 2 of the lemma now follows in straightforward fashion from (5.8) and (5.9). That Claim 1 holds follows from these and the fact that $EW_N^{OPT}(F) = E[OPT(L_n(F))] - E[s(L_n(F))/B]$ and $E[s(L_n(F))] = nES(F)$. ■

Thus one can determine whether $EW_n^{OPT}(F)$ is sublinear and, if it is not, compute the constant of proportionality on the expected linear waste, all in the time it takes to construct and solve the Waste LP for F. The worst-case time for this process obeys the following time bound.

THEOREM 5.5. *Given a description of a discrete distribution F in which all probabilities are presented as rational numbers with a common denominator $D \ge B$, the Waste LP for F can be constructed and solved in time*

$$O\left((JB)^{4.5} \log^2 D\right) = O\left(B^9 \log^2 D\right).$$

PROOF. Given its straightforward description, the LP can clearly be constructed in time proportional to its size, so construction time will be dominated by the time to solve the LP. For that, the best algorithm currently available is that of Vaidya [1989], whose running, expressed in bit operations, is $O((M+N)^{1.5}NL^2)$, where M is the larger of the number of variables and the number of constraints (the latter including the "≥ 0" constraints), N is the smaller, and L is a measure of the number of bits needed in the computation if all operations are to be performed in exact arithmetic.

Our LP has JB variables and the number of constraints is $\Theta(JB)$. Thus for our LP the running time is $O((JB)^{2.5}L^2) = O(B^5L^2)$. To obtain a bound on L, note that all coefficients in the constraints of the LP are 1, 0, or -1 and the coefficients in the objective function are all $O(B)$. The leaves the probabilities p_j to worry about. Note that we can determine $c(F)$ by solving the LP with each p_j replaced by its numerator (the integer Dp_j), and then dividing the answer by D. If we proceed in this way, then all the "probabilities" are integers bounded by D. Following the precise definition of L given in Vaidya [1989] we can then conclude that $L = O(JB \log D)$, giving us the overall running time bound claimed. ■

Although this running time bound is pseudopolynomial in B, it will be polynomial if B is polynomially bounded in terms of J, which is true for many of the distributions of interest in practice. Moreover, much better running times are obtainable in practice by using commercial primal simplex codes rather than interior point techniques to solve the LP's. See Applegate et al. [2003] which details simplex-based methods that can be used to compute $c(F)$ in reasonable time for discrete distributions with J and B as large as 1,000 and 10,000, respectively.

In the remainder of this section, we will show how we can further distinguish between the cases in which $EW_n^{OPT}(F) = \Theta(\sqrt{n})$ and those in which $EW_n^{OPT}(F) = O(1)$. Our goal is to distinguish cases (a) and (b) in the Courcoubetis-Weber theorem, as described in Section 2. Thus we need to determine, given that $\bar{p}_F$ is in Λ_F, whether it is also in the interior of Λ_F. Our approach is based on solving J additional, related LP's. The total running time will simply be $J+1$ times that for solving the original LP, and so we will be able to determine whether $EW_n^{OPT}(F) = O(\sqrt{n})$ and if so, which of the two cases hold, in total time $O(J^{5.5}B^{4.5}\log^2 D) = O(B^{10}\log^2 D)$.

For each i, $1 \le i \le J$, let $x_i \ge 0$ be a new variable and let LP_i denote the linear program obtained from the Waste LP for F by (a) changing the inequalities in (5.4) to equalities, (b) replacing (5.3) by

$$\begin{aligned} \sum_{h=0}^{B-1} v(i,h) &= p_i + x_i, \\ \sum_{h=0}^{B-1} v(j,h) &= p_j, \quad 1 \le j \le J \text{ and } j \ne i, \end{aligned} \tag{5.10}$$

and (c) changing the optimization criterion to "maximize x_i." Let $c_i(F)$ denote the optimal objective function value for LP_i. Note that LP_i is feasible for $x_i = 0$ whenever $c(F) = 0$, so that $c_i(F)$ is always well-defined and non-negative in this case.

Theorem 5.6. *If F is a discrete distribution, then $EW_n^{OPT}(F) = O(1)$ if and only if $c(F) = 0$ and $c_i(F) > 0$, $1 \le i \le J$.*

Proof. Combining the Courcoubetis-Weber Theorem with Theorem 5.3 we know that for all discrete distributions F,

$$\bar{p}_F \in \Lambda_F \text{ if and only if } c(F) = 0. \tag{5.11}$$

Let $\bar{q}(i,\beta)$ denote the vector obtained from $\bar{p}_F$ by setting $q_i = p_i + \beta$ and $q_j = p_j, j \ne i$. By (5.11) and the construction of the linear programs LP_i, it is easy to see that $\bar{q}(i,\beta)$ is in Λ_F if and only if LP_i is feasible when $x_i = \beta$. Thus by convexity, $\bar{q}(i,\beta)$ is in Λ_F if and only if $0 \le \beta \le c_i(F)$.

Let us first suppose that the stated properties of $c(F)$ and the $c_i(F)$ do not hold. If $c(F) \ne 0$, then $\bar{p}_F$ is not even in Λ_F, much less in its interior. So suppose $c(F) = 0$ but $c_i(F) = 0$ for some i, $1 \le i \le J$. Then for any $\epsilon > 0$ there is a vector $\bar{q}$ with $|\bar{q} - \bar{p}_F| \le \epsilon$ that is not in Λ_F, namely $\bar{q}(i,\epsilon)$. Thus by definition $\bar{p}_F$ is not in the interior of Λ_F.

On the other hand, suppose $c(F) = 0$ and $c_i(F) > 0$, $1 \le i \le J$. To show that $\bar{p}_F$ is in the interior of Λ_F, we make use of two elementary properties of such cones:

C1. If the vector $\bar{a} = \langle a_1, \ldots, a_d \rangle$ is in a cone Λ, then so is the vector $r\bar{a} = \langle ra_1, \ldots, ra_d \rangle$ for any $r > 0$.

C2. If vectors $\bar{a} = \langle a_1, \ldots, a_d \rangle$ and $\bar{b} = \langle b_1, \ldots, b_d \rangle$ are in Λ, then so is the vector sum $\bar{a} + \bar{b} = \langle a_1 + b_1, \ldots, a_d + b_d \rangle$.

In other words, any positive linear combination of elements of the cone is itself in the cone. Our proof works by showing that there is an ϵ such that any $\bar{q}$ with $|\bar{p}_F - \bar{q}| \le \epsilon$ can be constructed out of a positive linear combination of vectors $\bar{q}(i, \beta_i)$ with $0 \le \beta_i \le c_i(F)$, $1 \le i \le J$. We begin by defining a set of key quantities.

$$\begin{aligned}
c_{min} &= \min\{c_i(F) : 1 \le i \le J\}, \\
p_{max} &= \max\{p_i : 1 \le i \le J\}, \\
p_{min} &= \min\{p_i : 1 \le i \le J\}, \\
\delta &= \min\left\{\frac{1}{2}, \frac{c_{min}}{4Jp_{max}}\right\}, \\
\epsilon &= \min\left\{\frac{p_{min}}{4}, \left(\frac{c_{min}}{8J}\right)\left(\frac{p_{min}}{p_{max}}\right)\right\}.
\end{aligned}$$

Note that by hypothesis $c_{min} > 0$ and by definition all the p_i are positive, so $p_{min} > 0$. Hence δ and ϵ are also positive. Suppose $\bar{q} = \langle q_1, \ldots, q_J \rangle$ is any vector with $|\bar{p}_F - \bar{q}| \le \epsilon$. We will show that $\bar{q}$ can be constructed out of a positive linear combination of vectors $\bar{q}(i, \beta_i)$ as specified above.

Let $\epsilon_i = q_i - (1-\delta)p_i$, $1 \le i \le J$. We first observe that all the ϵ_i are positive. This is clearly true for all i such that $q_i \ge p_i$. Suppose $q_i < p_i$. If $\delta = 1/2$ we have

$$\epsilon_i = q_i - p_i + \delta p_i \ge \delta p_i - \epsilon \ge \frac{p_{min}}{2} - \frac{p_{min}}{4} = \frac{p_{min}}{4} > 0. \tag{5.12}$$

If on the other hand $\delta = c_{min}/(4Jp_{max})$, then

$$\epsilon_i \ge \delta p_i - \epsilon \ge \left(\frac{c_{min}}{4Jp_{max}}\right) p_{min} - \left(\frac{c_{min}}{8J}\right)\left(\frac{p_{min}}{p_{max}}\right) = \left(\frac{c_{min}}{8J}\right)\left(\frac{p_{min}}{p_{max}}\right) > 0. \tag{5.13}$$

We next observe that for each i, $1 \le i \le J$,

$$\epsilon_i = q_i - p_i + \delta p_i \le \epsilon + \delta p_i \le \frac{c_{min}}{8J} + \frac{c_{min}}{4Jp_{max}} p_{max} < \frac{c_{min}}{2J}. \tag{5.14}$$

Now consider the vectors $\bar{q}(i, \beta_i)$, where $\beta_i = J\epsilon_i/(1-\delta)$, $1 \le i \le J$. By (5.12) through (5.14) and the definition of δ, we have

$$0 < \beta_i = \frac{J\epsilon_i}{1-\delta} \le 2J\left(\frac{c_{min}}{2J}\right) = c_{min},$$

and so all these vectors are in Λ_F. Now consider the vector

$$\bar{r} = \langle r_1, \ldots, r_J \rangle = \frac{1-\delta}{J} \sum_{i=1}^{J} \bar{q}(i, \beta_i).$$

Since $\bar{r}$ is a positive linear combination of vectors in Λ_F, it is itself in Λ_F by (C1) and (C2). But now note that for $1 \le i \le J$, we have

$$r_i = \left(\frac{1-\delta}{J}\right)(Jp_i) + \left(\frac{1-\delta}{J}\right)\left(\frac{J\epsilon_i}{1-\delta}\right) = (1-\delta)p_i + \epsilon_i = q_i.$$

Thus $\bar{q} = \bar{r}$ and the latter is in Λ_F, as claimed. This implies that $\bar{p}_F$ is in the interior of Λ and the theorem is proved. ■

6. HANDLING NON-PERFECTLY PACKABLE DISTRIBUTIONS

In this section we consider the case when $EW_n^{OPT}(F) = \Theta(n)$. As we saw in Section 4, we can have $ER_\infty^{SS}(F) > 1$ for such F. Fortunately, for each such F one can design a distribution-specific variant on SS that performs much better. Note that we may assume $B > 1$ since otherwise the packing problem is trivial.

THEOREM 6.1. *For any discrete distribution F with $EW_n^{OPT}(F) = \Theta(n)$, there exists a randomized variant SS_F of SS such that $EW_n^{SS_F}(F) = EW_n^{OPT}(F) + O(\sqrt{n})$ and hence $ER_\infty^{SS_F}(F) = 1$ by Lemmas 5.1 and 5.4. This algorithm has expected running time $O(nB)$ and can itself be constructed in time polynomial in B and the size of the description of F.*

PROOF. Algorithm SS_F is based on the solution to the Waste LP for F, and in particular on the optimal solution value $c(F)$, which by Theorem 5.5 can be computed in time polynomial in B and the size of the description of F. The algorithm works by performing a series of steps, with new steps being taken so long as an item in L remains to be packed. At each step we flip a biased coin and according to the outcome proceed as follows.

(1) With probability $1/(1 + c(F))$ we take the next item from L and pack it according to SS.

(2) With probability $c(F)/(1 + c(F))$ we generate a new "imaginary" item of size 1 and pack it according to SS.

Let G_n denote the total size of the gaps in the packing of $L_n(F)$ by this algorithm, and let I_n denote the total size of the imaginary items in the packing. Then

$$EW_n^{SS_F}(F) = \frac{E[I_n] + E[G_n]}{B}. \tag{6.1}$$

It is straightforward to determine $E[I_n]$. Divide the packing process into n phases, each phase ending on a step in which a real rather than imaginary item is packed. The expected number of imaginary items packed in each phase is

$$\sum_{i=1}^{\infty} \left(\frac{c(F)}{1 + c(F)} \right)^i = c(F).$$

We thus can conclude the expected total number of imaginary items is $nc(F)$, and since each is of size 1 we have $E[I_n] = nc(F)$.

Let us now turn to $E[G_n]$. Note that if we consider both real and imaginary items, we are essentially packing a list generated by the distribution F^+ that has a higher proportion of items of size 1 than does F (if indeed F has any such items). If $s_1 = 1$ (in which case F does have items of size 1), we have $p_1^+ = (p_1 + c(F))/(1 + c(F))$ and $p_i^+ = p_i/(1 + c(F))$ for all $i > 1$. Otherwise $s_1^+ = 1$ and $s_{j+1}^+ = s_j$, $1 \le j \le J$, while $p_1^+ = c(F)/(1 + c(F))$ and $p_{j+1}^+ = p_j/(1 + c(F))$, $1 \le j \le J$.

CLAIM 6.2. $EW_n^{OPT}(F^+) = O(\sqrt{n})$.

Proof of Claim. By Theorem 5.3 all we need show is that the solution to the Waste LP for F^+ has $c(F^+) = 0$. Denote this LP by LP_{F^+} and denote the Waste LP for F by LP_F. We concentrate on the case where $s_1 = 1$ and $p_1^+ = (p_1 + c(F)/(1 + c(F))$. The case where $s_1 > 1$ can be handled by the same basic argument, but its description would be messier because we no longer have $s_j^+ = s_j$, $1 \le j \le J$. Let $v_0(j,h)$ be the variable values in an optimal solution for LP_F, and for $1 \le h \le B-1$ define

$$\Delta_h = \sum_j v_0(j, h - s_j) - \sum_j v_0(j,h).$$

Note that by (5.5), $c(F) = \sum_{h=1}^{B-1}(B-h)\Delta_h$. Define a new assignment v by

$$\begin{aligned} v(j,h) &= \frac{v_0(j,h)}{1+c(F)}, \quad j \neq 1, \\ v(1,h) &= \frac{v_0(1,h) + \sum_{h'=1}^{h} \Delta_{h'}}{1+c(F)}. \end{aligned}$$

for $0 \le h \le B-1$.

We claim that v satisfies the constraints of LP_{F^+} and achieves 0 for the objective function, thus implying that $c(F^+) = 0$. It is easy to see that v satisfies constraints (5.1) and (5.2) and satisfies (5.3) for $j \neq 1$. For (5.3) and $j = 1$, we have

$$\begin{aligned} \sum_{h=0}^{B-1} v(1,h) &= \frac{1}{1+c(F)}\left(p_1 + \sum_{h=1}^{B-1}\sum_{h'=1}^{h} \Delta_{h'}\right) \\ &= \frac{1}{1+c(F)}\left(p_1 + \sum_{h'=1}^{B-1}(B-h')\Delta_{h'}\right) \\ &= \frac{1}{1+c(F)}\Big(p_1 + c(F)\Big), \end{aligned}$$

as required. As for the constraints (5.4), we have for each h, $1 \le h \le B-1$, that

$$\begin{aligned} \sum_j v(j, h-s_j) \;-\; \sum_j v(j,h) & \\ &= \frac{1}{1+c(F)}\left(\sum_j v_0(j,h-s_j) - \sum_j v_0(j,h) + \sum_{h'=1}^{h-1}\Delta_{h'} - \sum_{h'=1}^{h}\Delta_{h'}\right) \\ &= \frac{1}{1+c(F)}\left(\Delta_h + \sum_{h'=1}^{h-1}\Delta_{h'} - \sum_{h'=1}^{h}\Delta_{h'}\right) \;=\; 0. \end{aligned}$$

Thus v is a feasible solution for LP_{F^+}. Finally, the value of the objective function is

$$\sum_{h=1}^{B-1}\Big(B-h\Big)\left(\sum_j v(j,h-s_j) - \sum_j v(j,h)\right) \;=\; 0. \;\blacksquare$$

Thus F^+ is a perfectly packable distribution. By Lemma 2.2, this means that the expected increase in $ss(P)$ during each step of algorithm SS_F is less than 2, no matter what the current packing looks like. For all $i > 0$ the expected increase during step i is thus less than 2 times the probability SS_F takes i or more steps. Since the expected number of steps by the above argument about $E[I_n]$ is $n(1 + c(F))$, the expected value of $ss(P)$ when the algorithm terminates is thus no more than $2n(1 + c(F))$. By Lemma 2.4 this implies that $E[G_n] \le B\sqrt{(B-1)n(1+c(F))} = O(B^2\sqrt{n})$ since $c(F) \le B$ by (5.7). Thus by (6.1) we have

$$EW_n^{SS_F}(F) = \frac{nc(F) + O(B^2\sqrt{n})}{B},$$

which by Lemma 5.1 is $EW_n^{OPT}(F) + O(\sqrt{n})$, as desired.

All that remains is to show that algorithm SS_F can be implemented to run in time $O(nB)$. This is not immediate, since there are distributions F for which $c(F)$ is as large as $\lceil B/2 \rceil - 1$. Thus the total number of items packed (including imaginary ones) can be $\Theta(nB)$, and the standard implementation of SS will take $\Theta(nB^2)$. We avoid this problem by using a more sophisticated implementation, that adds an additional data structure to aid with the packing of the imaginary items.

This data structure is a doubly-linked list of doubly-linked lists D_d. If P is the current packing, define $\delta_h = N_p(h+1) - N_P(h)$, $0 \le h \le B-1$, with $N_P(0)$ and $N_P(B)$ taken by convention to be 1/2 and $-1/2$ respectively. Then we know by Lemma 1.2 and the discussion that follows it that placing an item of size 1 into a bin of level h will yield a smaller increase (or bigger decrease) in $ss(P)$ than placing it in a bin of level h' if and only if $\delta_h < \delta_{h'}$. At any given time in the packing process, there is a sublist D_d for each value d taken on by some δ_h, with that sublist containing representatives for all those h such that $\delta_h = d$ and annotated by the value of d. The sublists are ordered in the main list by increasing value of d. For each value of h, $0 \le h \le B-1$, there is a pointer to the list for δ_h and to the representative for h in that list.

Given this data structure, we can pack an item of size 1 in constant time: find the first h in the first list D_d and place the item into a bin of level h. Note that this choice of h may violate the official tie-breaking rule for SS which requires that in case of ties, we should choose the *largest* h with $\delta_h = d_1$. However, as observed when we originally specified the official tie-breaking rules, none of the performance bounds proved in this paper depend on the precise tie-breaking rule used. Thus, we will still have $ER_\infty^{SS_F}(F) = 1$ if SS_F is implemented this way.

To complete the proof that this implementation takes $O(nB)$ time overall, we must show how to keep the data structure current with a constant amount of effort whenever an item of size 1 is packed. (We can afford $O(B)$ time for other, non-"imaginary" items, and it is easy to see how to perform the update if this much time is allowed.) For items of size 1 we exploit the fact that only two counts get changed, and no count changes by more than 1. Thus at most four δ_h's will change, and no δ_h can change by more than 2. Thus all we need show is that if δ_h changes by 2 or less, only a constant amount of work is required to update the data structure. But this follows from the fact if h is in D_d, then its new sublist can be at most two sublists away in the overall doubly-linked list, either in an already-existing sublist to which h can be prepended, or in a new sublist containing only h that can be created in constant time. ■

An obvious drawback of the algorithms SS_F is that we must know the distribution F in advance. Fortunately, we can adapt the approach taken in these algorithms to obtain a distribution-independent algorithm, simply by learning the distribution as we go along. If we engineer this properly, we can get a randomized algorithm that matches the best expected behavior we have seen in all situations:

THEOREM 6.3. *There is a randomized online algorithm SS^* that for any discrete distribution F with bin capacity B has the following properties:*

(a) *SS^* runs in expected time $O(nB)$.*
(b) *$EW_n^{SS^*}(F) = EW_n^{OPT}(F) + O(\sqrt{n})$.*
(c) *$ER_\infty^{SS^*}(F) = 1$.*
(d) *If $EW_n^{OPT}(F) = \Theta(\sqrt{n})$, then $EW_n^{SS^*}(F) = \Theta(\sqrt{n})$.*
(e) *If $EW_n^{OPT}(F) = O(1)$, then $EW_n^{SS^*}(F) = O(1)$.*

PROOF. Note that (d) will follow immediately from (b) and that (c) will follow from (b) via Lemmas 5.1 and 5.4. Thus we only need to prove (a), (b), and (e), which we will do in that order.

As the basic building blocks of SS^*, we will use a class of algorithms SS_D^r, $0 \le r < 1$ and $D \subset \{1, 2, \ldots, B-1\}$, that capture the essence of the algorithms SS_F of Theorem 6.1, modified slightly so that we can guarantee (e) above. Recall from Section 3.2 the algorithm SS' that guaranteed $EW_n^{SS'}(F) = O(1)$ for all bounded waste distributions. This algorithm made use of a parameterized packing rule SS_D, which packed so as to minimize $ss(P)$ subject to the constraint that no bin with a level in D should be created unless this is unavoidable, in which case we start a new bin. Algorithm SS' maintained a set U of all the item sizes seen so far, and used $SS_{D(U)}$ to pack items, where $D(U)$ is the set of dead-end levels for U, and SS^* will do likewise.

Algorithm SS_D^r works in steps, where in each step we flip a biased coin and proceed as follows:

(1) With probability $1 - r$ we take the next item from L and pack it according to packing rule SS_D.
(2) With probability r we generate a new "imaginary" item of size 1 and pack it according to SS_D.

Note that if $r = c(F)/(1 + c(F))$, this is the same as SS_F except for the modified packing rule. If $r > 0$ then it reduces to SS_F as soon as the first item of size 1 arrives, as there can be no dead-end levels if items of size 1 are present.

In algorithm SS^* we maintain an auxiliary data structure of counts X_i, $1 \le i \le B-1$, where X_i is the number of items of size i so far encountered in the list. From this we can derive the set U of the item sizes actually seen so far, as well as the current *empirical distribution* F', whose probability vector $\bar{p}$ is $\langle X_1/N, X_2/N, \ldots, X_{B-1}/N \rangle$, where N is the number of items seen so far. The packing process consists of a sequence of *phases*, during each of which we apply the packing rule $SS^r_{D(U)}$, where U is the set of item sizes seen up to and including the first item to be packed in the phase and $r = c(F')/(1 + c(F'))$ for the empirical distribution F' at the beginning of the phase.

We start the algorithm with a 0-phase in which U consists of the size of the first item in the list and F' is the empirical distribution based on just that first item. An i-phase terminates when either (a) we see a new item size and have to update U and recompute $D(U)$ or (b) we have packed a prespecified number of real items during the phase, where the number is $10B$ for a 0-phase and $30B4^{i-1}$ for an i-phase, $i > 0$. If an i-phase is terminated by the arrival of an item with a previously unseen size, the next phase is once again a 0-phase. Otherwise, it is an $(i + 1)$-phase. If the new phase has a different value for U or r, we begin it by closing all open bins. (A partially filled bin is considered *open* until it is closed. A closed bin can receive no further items and does not contribute to the count for its level.) We shall refer to phases that occur before all item sizes have been seen as *false* phases, and ones that occur after as *true* phases. Note that once the true phases begin, each phase (except possibly the last) packs 3 times as many items as the total number of items packed in all previous true phases.

Note that this algorithm will have the claimed running time. The list-of-lists data structure developed to enable the algorithms SS_F to run in time $O(nB)$ can be adapted to handle the SS^r_D packing rules, so the cumulative time spent running SS^r_D for the various values of D and r is $O(nB)$. In SS^* we have the added cost of re-initializing this data structure from time to time when we close all open bins, which can take $\Theta(B)$ time, but this can happen no more than $J(\lceil \log_4(n/10B) \rceil)$ times. Thus the overall time for reinitialization is $O(B^2 \log n) = o(nB)$ for fixed B. The only other computation time we need to worry about is that needed to solve the LP's used to compute the values of $c(F')$. By Theorem 5.5, the time for the LP computed at the beginning of an i-phase is $O(B^9 \log^2 D)$ where $D \le n$. Since there are no more than $J(\lceil \log_4(n/10B) \rceil)$ phases, the total time spent in solving the LP's is thus $O(B^{10} \log^3 n)$ and for fixed B is again asymptotically dominated by the time to pack the items.

The proof that SS^* satisfies (b) will proceed via a series of lemmas. To simplify things, we will modify our notation for discrete distributions. Given a discrete distribution F for bin size B, let p_i now represent the probability associated with items of size i (and hence $p_i > 0$ iff $i \in U_F$). Thus each discrete distribution for bin size B is represented by a length-$(B-1)$ vector $\bar{p}$ with non-negative entries. Under this revised formulation, the LP for F changes only minimally: J becomes $B - 1$ and s_j becomes j. It is easy to see that with these changes, the optimal solution value $c(F)$ is unchanged.

In what follows, if $\bar{p}$ and $\bar{p}'$ are two length-$(B - 1)$ vectors, we will use $||\bar{p} - \bar{p}'||$ to denote the L^1 distance between them, that is,

$$\|\bar{p} - \bar{p}'\| \equiv \sum_{i=1}^{B-1} |p_i - p'_i|.$$

LEMMA 6.4. *Suppose F and F' are discrete distributions for bin size B with probability vectors $\bar{p}$ and $\bar{p}'$. Then*

$$|c(F) - c(F')| \le B\|\bar{p} - \bar{p}'\|. \tag{6.2}$$

PROOF. We show how to convert an optimal solution to the LP for F to a solution to the LP for F' for which the objective function c satisfies

$$c \le c(F) + B\|\bar{p} - \bar{p}'\|. \tag{6.3}$$

A symmetric argument holds for the situation where the roles of F and F' are interchanged, and so (6.2) will follow.

For the purposes of this proof we can view our LP's as determined simply by the probability vectors for the distributions, $\bar{p}$ and $\bar{p}'$, and write $c(\bar{p})$ and $c(\bar{p}')$ for $c(F)$ and $c(F')$ respectively. We will convert an optimal solution to the LP for $\bar{p}$ to a feasible one for $\bar{p}'$ via a series of steps.

For $0 \le j \le B-1$, let $\bar{p}^j = (p_1^j, \ldots, p_{B-1}^j)$ be the vector with $p_i^j = p_i'$, $1 \le i \le j$ and $p_i^j = p_i$, $j+1 \le i \le B-1$. Note that $\bar{p}^0 = \bar{p}$ and $\bar{p}^{B-1} = \bar{p}'$. Let LP_j denote the LP for $\bar{p}^j$. Note that these are legitimate LP's even though the intermediate vectors $\bar{p}^j$, $0 < j < B-1$, may not have $\sum_{i=1}^{B-1} p_i^j = 1$ and hence need not correspond to probability distributions. We will show how to convert an optimal solution to LP_{j-1} to a feasible one for LP_j, $1 \le j \le B-1$, for which the objective function c satisfies

$$c \le c(\bar{p}^{j-1}) + B|p_j - p_j'|. \tag{6.4}$$

Inequality (6.3) will then follow by induction.

So consider an optimal solution to LP_{j-1}. Note that the only constraint of LP_j that is violated is the constraint of type (5.3) for j, i.e., the constraint that says that $\sum_{h=0}^{B-1} v(j,h) = p_j'$. If $p_j' \ge p_j$, our task is simple. We simply add $p_j' - p_j$ to $v(j,0)$ and leave all other variables unchanged. This will now satisfy the above constraint for j while not causing any of the others to be violated. The increase in the objective function will be $(B-j)|p_j' - p_j| \le B|p_j' - p_j|$, so (6.4) holds, as desired.

For the remaining case, suppose $p_j' < p_j$ and consider an optimal solution to LP_{j-1} that has the largest possible value for the potential function $\sum_{h=0}^{B-1} h \cdot v(j,h)$. We claim that this solution must be such that

$$\text{for all levels } h, \text{ if } v(j,h) > 0, \text{ then } v(i,h+j) = 0 \text{ for all } i \ne j. \tag{6.5}$$

Suppose not, and hence there is a level h and an integer $i \ne j$ such that $v(j,h) > 0$ and $v(i,h+j) > 0$. This means that a positive amount of size j was placed in bins with level h and then a positive amount of size i was placed in bins with the resulting level $h+j$. Let $\Delta = \min\{v(j,h), v(i,h+j)\}$, and modify the solution so that instead of first placing an amount Δ of j in bins of level h and then adding Δ of size i, we do these in reverse order. To be specific, revise $v(j,h)$ to $v(j,h) - \Delta$, $v(i,h)$ to $v(i,h) + \Delta$, $v(i,h+j)$ to $v(i,h+j) - \Delta$ and $v(j,h+i)$ to $v(j,h+i) + \Delta$. It is not difficult to see that this will not affect the objective function or any of the constraints, and so the new set of variable values will continue to represent an optimal solution to LP_{j-1}. Moreover, the potential function will have increased by $i\Delta$, a contradiction.

To convert the above optimal solution to one that is feasible for LP_j, we proceed as follows. Let $H^* = \min\{H \le B : \sum_{h=H}^{B-1} v(j,h) \le p_j - p_j'$. Set $v(j,h) = 0$, $H^* \le h \le B-1$, and reduce $v(j, H^*-1)$ by $p_j - p_j' - \sum_{h=H^*}^{B-1} v(j,h)$. The resulting solution will now satisfy the constraint of type (5.3) for j in LP$_j$. It will continue to satisfy the constraints of type (5.4) because of (6.5). Finally, the increase in the objective function will be at most $j|p_j - p'_j| \le B|p_j - p'_j|$ and so (6.4) again holds, as desired. ■

DEFINITION 6.5. *If $\bar{p}$ is a probability vector and $r \ge 0$, then $aug(\bar{p}, r)$ is the probability vector $\bar{q}$ with*

$$q_j = \begin{cases} \dfrac{p_1 + r}{1+r} & \text{if } j = 1, \\ \\ \dfrac{p_j}{1+r} & \text{otherwise.} \end{cases}$$

LEMMA 6.6. *Suppose F is a discrete distribution for bin size B with probability vector $\bar{p}$ and suppose $r, r' \ge 0$. If $q = aug(\bar{p}, r)$ and $q' = aug(\bar{p}, r')$, then*

$$||q - q'|| < 2|r - r'|.$$

PROOF. By definition,

$$\begin{aligned} \|q - q'\| &= \sum_{j=2}^{B-1} \left| \frac{p_j}{1+r} - \frac{p_j}{1+r'} \right| + \left| \frac{p_1 + r}{1+r} - \frac{p_1 + r'}{1+r'} \right| \\ &\le \left| \frac{1}{1+r} - \frac{1}{1+r'} \right| + \left| \frac{r}{1+r} - \frac{r'}{1+r'} \right| \\ &= \left| \frac{(1+r') - (1+r)}{(1+r')(1+r)} \right| + \left| \frac{r(1+r') - r'(1+r)}{(1+r')(1+r)} \right| < 2\,|r - r'|\,. \quad ■ \end{aligned}$$

LEMMA 6.7. *Suppose F is a discrete distribution for bin size B and F' is the empirical distribution measured after sampling n items with sizes chosen according to F for some $n > 0$. Let $\bar{p}$ and $\bar{p}'$ be the associated probability vectors and let $\bar{q} = aug(\bar{p}, c(F))$ and $\bar{q}' = aug(\bar{p}, c(F'))$. Then for all $\beta > 0$,*

$$\begin{aligned} &\text{(a)} \quad P\left(\|\bar{p} - \bar{p}'\| \ge \frac{J\beta}{\sqrt{2n}} \right) \le 2Je^{-\beta^2}, \\ &\text{(b)} \quad P\left(|c(F) - c(F')| \ge \frac{JB\beta}{\sqrt{2n}} \right) \le 2Je^{-\beta^2}, \\ &\text{(c)} \quad P\left(\|\bar{q} - \bar{q}'\| \ge \frac{\sqrt{2}JB\beta}{\sqrt{n}} \right) \le 2Je^{-\beta^2}. \end{aligned}$$

PROOF. By a variation on Chernoff bounds due to Hoeffding [Hoeffding 1963], and described in [Coffman, Jr. and Lueker 1991, p. 19], we have that for all j, $1 \le j \le B-1$, and $\beta > 0$,

$$P\left(\left| np_j - np'_j \right| \ge \beta\sqrt{n} \right) \le 2e^{-\beta^2/2},$$

which implies that

$$P\left(\left| p_j - p'_j \right| \ge \frac{\beta}{\sqrt{2n}} \right) \le 2e^{-\beta^2}.$$

Thus the probability that the bound is exceeded for at least one j is no more than $2(B-1)e^{-\beta^2}$. However, if $\|\bar{p}-\bar{p}'\| \geq (B-1)\beta/\sqrt{2n}$ then the bound must be exceeded for some j. Hence conclusion (a) holds. Conclusions (b) and (c) follow by Lemmas 6.4 and 6.6. ■

LEMMA 6.8. *Suppose F and F' are discrete distributions for bin size B, $\bar{q} = aug(\bar{p}, c(F))$, $\bar{q}' = aug(\bar{p}, c(F'))$, and $r' = c(F')/(1+c(F'))$. Suppose q_{min} is the smallest nonzero entry in $\bar{q}$ and $\|\bar{q}-\bar{q}'\| < q_{min}$. If the algorithm $SS^{r'}_{D(U_F)}$ is applied to a list L of n items generated according to F, then the resulting packing P of L plus the imaginary items created by $SS^{r'}_{D(U_F)}$ satisfies*

$$E[W(P)] = O(\max\{\sqrt{n}, \|\bar{q}-\bar{q}'\|n\}).$$

PROOF. Let $\delta = \|\bar{q}-\bar{q}'\|/q_{min}$. Since $\|\bar{q}-\bar{q}'\| < q_{min}$, we have $\delta < 1$ and for all j with $q_j > 0$, $q'_j \geq q - \|\bar{q}-\bar{q}'\| \geq (1-\delta)q_j > 0$.

Suppose items are generated according to F and we use $SS^{r'}_{D(U_F)}$ to pack them. At each step, we will thus be using $SS_{D(U_F)}$ to pack an item that looks as if it were generated according to the probability vector $\bar{q}'$. Let us view the packing process as follows: When an item of size j arrives, randomly classify it as a *good* item with probability $(1-\delta)q_j/q'_j$ and as a *bad* item with probability $1-(1-\delta)q_j/q'_j$. Note that if one restricts attention to the good items, they now arrive as if generated according to $\bar{q}$. Further note that by Claim 6.2 of Theorem 6.1, the distribution determined by $\bar{q}$ is a perfectly packable distribution. Thus for these arrivals we can apply Lemma 2.2, which we have already shown applies to $SS_{D(U_F)}$ as well as SS. Thus we can conclude that the expected increase in $ss(P)$ each time a good item is packed is less than 2.

Let D denote the constant $(1+q_{min})/q_{min}$. The probability that a random item is a bad item is

$$\sum_{i=1}^{J} q'_j \left(1 - \frac{(1-\delta)q_j}{q'_j}\right) = \sum_{i=1}^{J} \left(q'_j - q_j + \delta q_j\right) \leq \|\bar{q}-\bar{q}'\| + \delta = D\|\bar{q}-\bar{q}'\|.$$

For bad items, the worst-case increase in $ss(P)$ is less than $2\max_j\{N_P(j)\}+2$, an upper bound by Lemma 1.2 on the increase that would occur if our placement caused the maximum count to increase. Thus the expected increase in $ss(P)$ is less than

$$2\left(1 + D\|\bar{q}-\bar{q}'\| \max_j\{N_P(j)\}\right). \tag{6.6}$$

Let P_i be the packing after i items have been packed and let $i(t)$, $1 \leq t \leq n$, be the index of the packing that results when the tth real item is packed, with $i(0) = 0$ by convention. Note that our final packing $P = P_{i(n)}$. Define

$$\begin{aligned} Max_t &\equiv \max\{1, N_{P_{i(t)}}(j) :\ 1 \leq j \leq J\},\ 1 \leq t \leq n, \\ MaxE &\equiv \max\{E[Max_t] : 0 \leq t \leq n\}. \end{aligned}$$

CLAIM 6.9. *For all t, $0 \leq t \leq n$, and all i, $i(t) \leq i < i(t+1)$, the maximum level count in P_i is at most Max_t.*

Proof of Claim. The claim holds for $P_{i(t)}$ by definition. Suppose it holds for packing P_i, $i \geq i(t)$ and $i+1 < i(t+1)$, i.e., the next item to be packed is imaginary. In this case $SS^{r'}$ now must know that items of size 1 exist and hence that there are no dead-end levels. Hence it must make an improving move whenever one exists. Suppose the current packing has a count greater than 0 and j is the level with the biggest count, ties broken in favor of larger levels. Then there is at least one bin with level j and placing an item of size 1 into such a bin will decrease $ss(P)$. Thus $SS^{r'}$ must choose a placement that decreases $ss(P)$. This cannot increase the largest level count. Suppose on the other hand that the current packing has no level count exceeding 0. Then placing an imaginary item will only increase the maximum level count from 0 to 1, which is still no more than $Max_{i(t)}$. In both cases, we are left with a packing in which no count exceeds $Max_{i(t)}$. The claim follows by induction. ■

Claim 6.10. *For* $0 \leq t < n$,

$$E\left[ss(P_{i(t+1)}) - ss(P_{i(t)})|P_{i(t)}\right] \leq 2B\bigl(1 + D\|\bar{q} - \bar{q}'\|MaxE\bigr).$$

Proof of Claim. For each $k \geq 0$, the probability that there are more than k items packed in going from $P_{i(t)}$ to $P_{i(t+1)}$ is $\bigl(c(F')/(1+c(F'))\bigr)^k$. Given that there are more than k items packed, the expected increase in $ss(P)$ due to the packing of the $(k+1)$st item is by (6.6), Claim 6.9, and the definitions of Max_t and $MaxE$ at most

$$2(1 + D\|\bar{q} - \bar{q}'\|E[Max_t]) \leq 2(1 + D\|\bar{q} - \bar{q}'\|MaxE).$$

The total expected increase in going from $P_{i(t)}$ to $P_{i(t+1)}$ is thus at most

$$\sum_{k=0}^{\infty} \left(\frac{c(F')}{1+c(F')}\right)^k 2(1 + D\|\bar{q} - \bar{q}'\|MaxE) = 2(1 + c(F'))(1 + D\|\bar{q} - \bar{q}'\|MaxE).$$

The claim follows since $c(F) \leq B - 1$ for all distributions F by (5.7). ■

Thus by the linearity of expectations we can conclude that for $1 \leq t \leq n$

$$E[ss(P_{i(t)})] \leq 2Bt\bigl(1 + D\|\bar{q} - \bar{q}'\|MaxE\bigr) \tag{6.7}$$

and, by inequality (2.4) in the proof of Lemma 2.4, that

$$\begin{aligned} E[Max_t] \leq E\left[1 + \sum_{j=1}^{B-1} N_{P_{i(t)}}(j)\right] &\leq 1 + \sqrt{B \cdot E[ss(P_{i(t)})]} \\ &\leq 1 + \sqrt{2B^2 t\,(1 + D\|\bar{q} - \bar{q}'\|MaxE)} \\ &\leq 2B\sqrt{n\,(1 + D\|\bar{q} - \bar{q}'\|MaxE)} \end{aligned}$$

and hence

$$MaxE \leq 2B\sqrt{n\,(1 + D\|\bar{q} - \bar{q}'\|MaxE)}. \tag{6.8}$$

If $D\|\bar{q} - \bar{q}'\|MaxE \leq 1$, we have $E[ss(P_{i(n)})] \leq 4Bn$ by (6.7). So by Lemma 2.4 we have

$$E[W(P)] = E[W(P_{i(n)})] \leq \sqrt{BE[ss(P_{i(n)})]} \leq 2B\sqrt{n} = O(\sqrt{n})$$

for fixed F. Otherwise we have by (6.8) that $MaxE \leq 2B\sqrt{2Dn\|\bar{q}-\bar{q}'\|}\sqrt{MaxE}$. But this implies $MaxE \leq 8B^2Dn\|\bar{q}-\bar{q}'\|$, and consequently by (6.7)

$$E[ss(P_{i(n)})] \leq 2Bn + 16B^3D^2(\|\bar{q}-\bar{q}'\|n)^2$$

and hence by Lemma 2.4 that

$$E[W(P)] = E[W(P_{i(n)})] \leq \sqrt{BE[ss(P_{i(n)})]} = O(\max\{\sqrt{n}, \|\bar{q}-\bar{q}'\|n\})$$

for fixed F. Thus in either case $E[W(P)] = O(\max\{\sqrt{n}, \|\bar{q}-\bar{q}'\|n\})$ and Lemma 6.8 is proved. ∎

We can now address part (b) of Theorem 6.3. Let us divide the waste created by SS^* into three components. Let n_A denote the number of items seen before all sizes in U_F have appeared.

—Waste in bins created during the packing of the first n_A items (during what we called *false* phases).

—Waste in bins created after the first n_A items have been packed, either during the 0-phase or during an i-phase, $i > 0$, for which $\|\bar{q}-\bar{q}'\| > q_{min}$ in the terminology of Lemma 6.8 (*Type 1* true phases).

—Waste in the remaining bins (*Type 2* true phases).

For waste in bins created during false phases, we first determine a bound on $E[n_A]$. The analysis is similar to that used in the proof of Theorem 3.10. The probability that we have not seen all item sizes after the hth item arrives is no more than $J\,(1-p_{min})^h$. If we choose the smallest t such that $J(1-p_{min})^t \leq 1/2$, then for each integer $m > 0$, the probability that all the item sizes have not been seen after mt items have arrived is at most $1/2^m$. Thus for each $i \geq 0$, the probability that $n_A \in (mt, (m+1)t]$ is at most $1/2^m$. Hence

$$E[n_A] \leq \sum_{m=0}^{\infty}\Big(m+1)t \cdot p\big[n_A \in (mt, (m+1)t]\big]\Big) \leq t \cdot \sum_{m=0}^{\infty}\frac{(m+1)}{2^m} = 4t.$$

Thus the expected false phase waste resulting from bins that contain at least one real item is bounded by $4t(B-1)/B$.

The only other possible waste during false phases consists of 1 unit of waste for each bin containing only imaginary items. The expected number of imaginary items that arrive before all item sizes have been seen is bounded by $(n_A+1)c(F_{max})$, where F_{max} is the empirical distribution F' that has the largest value of $c(F')$ among all those computed before all item sizes have been seen. Since $c(F') \leq B-1$ for all distributions F' this is at most $(4t+1)(B-1)$. Moreover, all but one of the bins containing only imaginary items that are started during a given phase must be completely full: as already remarked, if there are any partially filled bins when an imaginary item (of size 1) arrives, then placing it in a bin whose level has the largest count (ties broken in favor of higher levels) will cause a decrease in $ss(P)$ and hence is to be preferred to starting a new bin. Thus the expected number of bins containing only imaginary items is at most $(4t+1)(B-1)/B$ plus the expected number of false phases. Since the number of false phases is clearly less than $n_A/(10B) + J$, the total expected waste during false phases is at most

$$\frac{4t(B-1)}{B} + \frac{(4t+1)(B-1)}{B} + \frac{E[n_A]}{10B} + J \le 8t+1+J - \frac{8t+1}{B} + \frac{4t}{10B} < 8t+J+1,$$

which is $O(1)$ for fixed F.

We now turn to the Type 1 true phases. The first of these is the true 0-phase, which is Type 1 by definition. In this phase the expected number of real items packed is at most $10B$ and the expected waste is at most $20B+1$ by an argument like that in the previous paragraph with $4t$ replaced by $10B$ and only one phase to worry about.

By a similar argument, if there is a true i-phase, $i > 0$, the number of real items packed in it is at most $30B \cdot 4^{i-1}$ and the expected waste during the phase is at most $60B \cdot 4^{i-1} + 1 < 16B \cdot 4^i$. Whether this phase contributes to the Type 1 waste depends on the empirical distribution F' measured at the beginning of the phase. In particular, we must have $\|\bar{q} - \bar{q}'\| > q_{min}$.

Now the distribution F' is based on at least $10B \cdot 4^{i-1}$ samples from F. Thus by Lemma 6.7(c), the probability that $\|\bar{q} - \bar{q}'\| \ge \sqrt{2}JB\beta/\sqrt{2.5B4^i}$ is bounded by $2Je^{-\beta^2}$. Thus the probability that $\|\bar{q} - \bar{q}'\| \ge q_{min}$ is at most $2Je^{-(1.25q_{min}^2/J^2B)4^i} = 2Jd^{-4^i}$ where $d = e^{1.25q_{min}^2/J^2B} > 1$ is a constant independent of i. The expected waste that this phase can produce by being a Type 1 phase is thus at most $(32BJ)(4^i/d^{4^i})$. Summing over all true phases we conclude that the total expected waste for Type 1 phases is at most

$$20B + 1 \ + \ 32BJ\sum_{i=1}^{\infty}\frac{4^i}{d^{4^i}} = O(1).$$

Finally, let us turn to the waste during Type 2 true phases. Suppose the true i-phase, $i > 0$, is of Type 2, and let F' be the empirical distribution at the beginning of the phase, with $\bar{p}'$ being its probability vector. F' must have been based on the observation of at least $10B4^{i-1}$ items generated according to F. Thus by Lemma 6.7(b) there are constants α and γ depending on F but independent of i such that $E\big[|c(F) - c(F')|\big] < \gamma/\sqrt{5B4^i} = \alpha 2^{-i}$.

Let N_i be the number of real items packed during the true i-phase, and recall that $N_i \le 30B4^{i-1}$. This means that the expected waste due to imaginary items created during the phase is at most

$$\frac{N_i c(F')}{B} \le \frac{N_i\big(c(F) + \alpha 2^{-i}\big)}{B} \le \frac{N_i c(F)}{B} + 7.5\alpha 2^i.$$

Note that the total number of true phases is at most $\lceil \log_4(n/10B)\rceil < \lfloor \log_4 n\rfloor = \lfloor (1/2)\log_2 n\rfloor$. Thus even if all such phases are of Type 2, we have that the expected total waste during the Type 2 phases due to imaginary items is bounded by

$$\frac{nc(F)}{B} + 7.5\alpha\sum_{i=1}^{\lfloor \log_4 n\rfloor} 2^i < \frac{nc(F)}{B} + 15\alpha\sqrt{n} = \frac{nc(F)}{B} + O(\sqrt{n}).$$

Now let us consider the waste caused by empty space in the bins packed during true phases of Type 2. First note that the set of items contained in open bins at the end of the i-phase consists of all items packed during this phase plus possibly items from immediately preceding true phases that operated with the same value of r. Even if all preceding true phases operated with the same value of r, this could be no more than $10B4^i$ items. Moreover, as argued above we know that the empirical distribution F' computed at the beginning of the i-phase has $E\big[|c(F) - c(F')|\big] < \alpha 2^{-i}$ for some fixed α, so that by Lemma 6.6, $E\big[\|\bar{q} - \bar{q}'\|\big] < 2\alpha 2^{-i}$. Since this is a Type 2 phase, we have by definition that $\|\bar{q} - \bar{q}'\| \le q_{min}$ and so Lemma 6.8 applies and we can conclude that there is a constant γ such that the expected empty space in the packing is bounded by

$$\gamma \max\left\{(10B4^i)(2\alpha 2^{-i}), \sqrt{10B4^i}\right\} = O(2^i).$$

Thus the expected total empty space of this kind over all true phases of Type 2 is once again $O(\sqrt{n})$, and so the expected total waste in bins started in Type 2 true phases (empty space plus imaginary items) is $nc(F)/B + O(\sqrt{n})$. Given that the expected waste in false levels and in true levels of Type 1 was bounded, this means that

$$EW_n^{SS^*}(F) = \frac{nc(F)}{B} + O(\sqrt{n}),$$

which by Lemma 5.1 means that Claim (b) of Theorem 6.3 has been proved.

It remains to prove Claim (e), that $EW_n^{SS^*}(F) = O(1)$ whenever $EW_n^{OPT}(F) = O(1)$, i.e., whenever F is a bounded waste distribution. Suppose F is a bounded waste distribution with size vector $\bar{s}$ and probability vector $\bar{p}$. From the Courcoubetis-Weber Theorem, we know that there is an $\epsilon > 0$ such that any distribution F' over the same set of item sizes that has a probability vector $\bar{p}'$ satisfying $\|\bar{p} - \bar{p}'\| \le \epsilon$ is a perfectly packable distribution and hence has $c(F') = 0$ by Theorem 5.6.

Once again, we can divide the waste produced in an SS^* packing of a list generated according to F into three components, although this division is somewhat different.

—Waste in bins created during false phases.

—Waste in bins created in true phases through the last such phase in which the starting empirical distribution F' had $c(F') > 0$.

—Waste created in all subsequent phases.

As in the analysis of Claim (b), we can conclude that the total expected waste for the false phases is bounded.

Consider now the waste created in true phases through the last phase that started with $c(F') > 0$. If this was the true 0-phase, the expected waste is bounded by $20B+1$, again as argued in Claim (b). If it was the true i-phase, $i > 0$, then at most $10B4^i$ items can have been packed in true phases through this point, and so the expected waste would be at most $20B4^i + 1 < 21B4^i$ by an analogous argument. Now the probability that the i-phase is the last phase with $c(F') > 0$ is clearly no more than the probability that it simply had $c(F') > 0$. As remarked above, this can only have happened if $\|\bar{p} - \bar{p}'\| > \epsilon$. Since the empirical distribution at the start of the i-phase, $i > 0$, is based on at least $10B4^{i-1}$ samples from F, by Lemma 6.7(a), the probability that $\|\bar{p} - \bar{p}'\| > \epsilon$ is at most $2Je^{-(5B\epsilon^2/J^2)4^i} = 2Jd^{-4^i}$

for some $d > 1$. Thus the total expected waste through the last true phase with $c(F') > 0$ is at most

$$20B + 1 \ + \ 42JB \sum_{i=1}^{\infty} \frac{4^i}{d^{4^i}} = O(1).$$

Finally, if there are any phases after the last one that had $c(F') > 0$ and hence $r > 0$, let the first such phase be the i_0-phase. This phase begins by closing all previously open bins because r has just changed from a positive value to 0. From now on, however, no more bin closures will take place since $r = 0$ for all remaining phases and hence never changes. Thus the packing beginning with the i_0-phase is simply an $SS^0_{D(U_F)} = SS_{D(U_F)}$ packing of items generated according to F, and by Theorem 3.10 has $O(1)$ expected waste.

Thus the total expected waste under SS^* is $O(1)$, Claim (e) holds, and Theorem 6.3 is proved. ■

7. SS AND ADVERSARIAL ITEM GENERATION

The results for SS^* in the previous section are quite general with respect to the context traditionally studied by papers on the average case analysis of bin packing algorithms: the standard situation in which item sizes are chosen as independent samples from the same fixed distribution F. However, that context itself is somewhat limited, in that one can conceive of applications in which some dependence exists between item sizes. Perhaps surprisingly, the arguments used to prove Theorems 2.5 and 3.4 imply that SS itself can do quite well in some situations where there is dependence and that dependence is controlled by an adversary.

Suppose that our item generation process works as follows: Let B be a fixed bin size. For each item x_i, $i = 1, 2, \ldots$, the size of item x_i is chosen according to a discrete distribution F_i with bin size B. The choice of F_i, however, is allowed to be made by an adversary, given full knowledge of all item sizes chosen so far, the current packing, and the packing algorithm we are using. It would be difficult to do well against such an adversary unless it were somehow restricted, so to introduce a plausible restriction, let us say that such an adversary is *restricted to* $\mathcal{F}$, where $\mathcal{F}$ is a set of discrete distributions, if all the F_i used must come from $\mathcal{F}$. As a simple corollary of the proof of Theorem 2.5 we have the following.

THEOREM 7.1. *Let B be a given bin size and suppose items are generated by an adversary restricted to the set of all perfectly packable distributions for bin size B. Then the expected waste under SS is $O(\sqrt{n})$.*

PROOF. By Lemma 2.2 we know that $E[ss(P)]$ increases by less than 2 whenever SS packs an item whose size is generated by a perfectly packable distribution. Thus we can conclude that if we pack n items generated by our adversary, we still must have $E[ss(P)] < 2n$. The rest follows by Lemma 2.4, as in the proof of Theorem 2.5. ■

Note that without the restriction to perfectly packable distributions, the adversary could force the *optimal* expected waste to be linear, so Theorem 7.1 is in a sense the strongest possible result of this sort. With even more severe restrictions on $\mathcal{F}$, one can guarantee *bounded* expected waste against an adversary.

THEOREM 7.2. *Suppose $\mathcal{F}$ is a set of bounded waste distributions for a given bin size B, none of which has multiply-occurring dead-end levels, and there is an $\epsilon > 0$ such that every distribution that is within distance ϵ of a member of $\mathcal{F}$ is a perfectly packable distribution. Then if items are generated by an adversary restricted to $\mathcal{F}$, the expected waste under SS is $O(1)$.*

PROOF. This follows from the proof of Theorem 3.4, since the general hypothesis of Hajek's Lemma allows for adversarial item generation. Essentially the same proof as was used to show Theorem 3.4 applies. ■

Note that for any bounded waste distribution F there is an $\epsilon_F > 0$ such that every distribution that is within distance ϵ_F of F is perfectly packable, as follows from the Courcoubetis-Weber Theorem. Hence the universal ϵ required by Theorem 7.2 must exist for any *finite* set of bounded waste distributions. Thus we have the following interesting corollary.

COROLLARY 7.3. *Suppose $\mathcal{F} = \{U\{j,k\}, 1 \le j \le k-2\}$ for some fixed $k > 0$. Then if items are generated by an adversary restricted to $\mathcal{F}$, the expected waste under SS is $O(1)$.*

PROOF. As shown in Coffman, Jr. et al. [2000a] abd Coffman, Jr. et al. [2002b], all $k-2$ such distributions are bounded waste distributions. ■

If we omit from Theorem 7.2 the requirement that the distributions in $\mathcal{F}$ have no multiply-occurring dead-end levels, then the best upper bound on the expected waste for SS grows to $O(\log n)$, as follows from the proof of Theorem 3.11. Note that we cannot improve this to $O(1)$ by using SS' instead of SS as we did in the non-adversarial case. For example, the adversary could generate its first item using the distribution that yields items of size 1 with probability 1, and then switch to a bounded waste distribution with multiply-occurring dead-end levels. SS', having seen an item of size 1, would conclude that $1 \in U_F$ and hence that there are no dead-end levels. So from then on it would pack exactly as SS would and hence would produce $\Omega(\log n)$ waste as implied by the lower bound in Theorem 3.11.

8. THE EFFECTIVENESS OF VARIANTS ON SS

In this section we return to the standard model for item generation, and ask how much of the good behavior of SS depends on the precise details of the algorithm. It turns out that SS is not unique in its effectiveness, and we shall identify a variety of related algorithms A that share one or more of the following *sublinearity* properties with SS (where (a) is a weaker form of (b)):

(a) [Sublinearity Property]. If $EW_n^{OPT}(F) = O(\sqrt{n})$, then $EW_n^A(F) = o(n)$.
(b) [Square Root Property]. If $EW_n^{OPT}(F) = O(\sqrt{n})$, then $EW_n^A(F) = O(\sqrt{n})$.
(c) [Bounded Waste Property]. If $EW_n^{OPT}(F) = O(1)$ and F has no multiply-occurring dead-end levels, then $EW_n^A(F) = O(1)$.

8.1 Objective Functions That Take Level into Account

One set of variants on SS are those that replace the objective function $ss(P)$ by a variant that multiplies the squared counts by some function depending only on

B and the corresponding level, and then packs items so as to minimize this new objective function. Examples include

$$\sum_{h=1}^{B-1} N_P(h)^2(B-h), \quad \sum_{h=1}^{B-1} \left[N_P(h)(B-h)\right]^2, \text{ and } \sum_{h=1}^{B-1} \frac{N_P(h)^2}{h}.$$

The first of the above three variants was proposed in 1996 by David Wilson (personal communication, 2000), before we had invented the algorithm SS itself. Wilson's unpublished experiments with this algorithm already suggested that it satisfied the Square Root and Bounded Waste Properties for the $U\{j,k\}$ distributions, a claim we can now confirm as a consequence of the following more general result.

THEOREM 8.1. *Suppose $f(h,B)$ is any function of the level and bin capacity, and A is the algorithm that packs items so as to minimize $\sum_{h=1}^{B-1} N_P(h)^2 f(h,B)$. Then A satisfies the Square Root and Bounded Waste Properties.*

PROOF. Such algorithms satisfy the Square Root Property, since by Lemma 2.2 the expected increase in the objective function at each step is still bounded by a constant $(2\max\{f(h,B) : 1 \le h \le B-1\})$. They satisfy the Bounded Waste Property, since the proof of Theorem 3.4 need only be modified to change some of the constants used in the arguments. Details are left to the reader. ∎

We conjecture that the $EW_n^{SS}(F) = \Theta(\log n)$ result of Theorem 3.11 for distributions F with multiply-occurring dead-end levels also carries over to these variants, but the length and complexity of the proof of the original result makes verification a much less straightforward task.

As to which of these variants performs best in practice, we performed preliminary experimental studies using the distributions studied in Csirik et al. [1999], i.e., $U\{h,100\}$, $1 \le h < 100$ (as defined in the Introduction), and $U\{18,j,100\}$, $18 \le j < 100$, where $U\{h,j,k\}$ is the distribution in which the bin size is k, the set of possible item sizes is $S = \{h, h+1, \ldots, j\}$, and all sizes in S are equally likely. The distributions in the first class are all bounded waste distributions except for $U\{99,100\}$, for which $EW_n^{OPT}(F) = \Theta(\sqrt{n})$. The distributions in the second class include ones with all three possibilities for $EW_n^{OPT}(F)$: $O(1)$, $\Theta(\sqrt{n})$, and $\Theta(n)$. We also tested a few additional more idiosyncratic distributions. The values of n tested typically ranged from 100,000 to 100,000,000. Our general conclusion was that there is no clear winner among SS and the variants describe above; the best variant depends on the distribution F.

8.2 Objective Functions with Different Exponents

A second class of variants that at least satisfy the Sublinearity Property is obtained by changing the exponent in the objective function.

THEOREM 8.2. *Suppose SrS denotes that algorithm that at each step attempts to minimize the function $\sum_{h=1}^{B-1}(N_P(h))^r$. Then for all perfectly packable distributions*

F,

$$EW_n^{SrS}(F) \;=\; \begin{cases} O\left(n^{\frac{1}{r}}\right), & 1 < r \le 2, \\ \\ O\left(n^{\frac{r-1}{r}}\right), & 2 \le r < \infty. \end{cases}$$

(Note that when $r = 2$ both bounds equal $O(\sqrt{n})$, the known bound for $SS = S2S$.)

Proof. Suppose P is any packing and a random item i is generated according to F. By the argument used in the proof of Lemma 2.2, we know that there is an algorithm A_F such that if i is packed by A_F, then for each h, $1 \le h \le B-1$, the expected increase in $N_P(h)^r$ given that $N_P(h)$ changes and that the current value $N_P(h) > 0$, is bounded by

$$\frac{1}{2}\Big((N_P(h)+1)^r - N_P(h)^r\Big) \;+\; \frac{1}{2}\Big((N_P(h)-1)^r - N_P(h)^r\Big)$$

$$= \frac{(N_P(h)+1)^r + (N_P(h)-1)^r}{2} - N_P(h)^r.$$

Let $x = \max\{N_P(h) : 1 \le h \le B-1\}$. Given that at most two counts change when an item is packed and that the expected increase for a zero-count is at most $1^r = 1$, the expected increase in $\sum_{h=1}^{B-1}(N_{P_n}(h))^r$ when i is packed is thus at most

$$\max\Big\{2,\; (x+1)^r + (x-1)^r - 2x^r\Big\}. \tag{8.9}$$

Since SrS packs items so as to minimize $\sum_{h=1}^{B-1}(N_{P_n}(h))^r$, the expected increase in this quantity when we pack i using SrS instead of A_F can be no greater.

We thus need to bound (8.9) when r is fixed. For $x \le 2$, it is clearly bounded by a constant depending only on r, so let us assume that $x > 2$. To bound (8.9) in this case, we know by Taylor's Theorem that there exist θ_1 and θ_2, $0 < \theta_1, \theta_2 < 1$, such that

$$(x+1)^r \;=\; x^r + rx^{r-1} + \frac{r(r-1)}{2!}x^{r-2} + \frac{r(r-1)(r-2)}{3!}(x+\theta_1)^{r-3}, \tag{8.10}$$

$$(x-1)^r \;=\; x^r - rx^{r-1} + \frac{r(r-1)}{2!}x^{r-2} - \frac{r(r-1)(r-2)}{3!}(x-\theta_2)^{r-3}. \tag{8.11}$$

Substituting, we conclude that (8.9) is bounded by the maximum of 2 and

$$r(r-1)x^{r-2} + \frac{r(r-1)(r-2)}{6}\Big[(x+\theta_1)^{r-3} - (x-\theta_2)^{r-3}\Big]. \tag{8.12}$$

If $1 < r < 2$, then (8.12) has a fixed bound depending only on r when $x > 2$. Thus if P_n is the packing that exists after all n items have been packed by SrS, the expected value of $\sum_{h=1}^{B-1}(N_{P_n}(h))^r$ is $O(n)$. If $r > 2$, then (8.12) grows as $\Theta(x^{r-2}) = O(n^{r-2})$. Thus in this case the expected value of $\sum_{h=1}^{B-1}(N_{P_n}(h))^r$ is $O(n^{r-1})$.

Let $C_i = \sum_{h=1}^{B-1} P[N_P(h) = i]$, $0 \le i < n$. Note that $\sum_{i=1}^{n} C_i = B-1$ and $\sum_{i=1}^{n} iC_i$ is the expected number of partially filled bins in the packing and hence an upper bound on the expected waste. We can bound this using Holder's Inequality:

$$\sum a_i b_i \le \left(\sum a_i^p\right)^{\frac{1}{p}} \left(\sum b_i^q\right)^{\frac{1}{q}} \text{ when } \frac{1}{p} + \frac{1}{q} = 1. \tag{8.13}$$

Set $a_i = i(C_i)^{\frac{1}{r}}$, $b_i = (C_i)^{\frac{r-1}{r}}$, $p = r$, and $q = \dfrac{r}{r-1}$. In the case where $1 < r < 2$, we have concluded that there is a d such that $\sum_{i=1}^n C_i i^r \le dn$. Thus Holder's Inequality yields

$$E[W(P_n)] \;<\; \sum iC_i \le \left(\sum C_i i^r\right)^{\frac{1}{r}} \left(\sum C_i\right)^{\frac{r-1}{r}} < (dn)^{\frac{1}{r}} B^{\frac{r-1}{r}} = O(n^{\frac{1}{r}})$$

as claimed. On the other hand, if $r > 2$ we have $\sum_{i=1}^n C_i i^r \le dn^{r-1}$ for some constant d and so Holder's Inequality yields

$$E[W(P_n)] \;<\; \sum iC_i \le \left(\sum C_i i^r\right)^{\frac{1}{r}} \left(\sum C_i\right)^{\frac{r-1}{r}} < d^{\frac{1}{r}} n^{\frac{r-1}{r}} B^{\frac{r-1}{r}} = O(n^{\frac{r-1}{r}})$$

as claimed. ■

Despite the differing qualities of the bounds in Theorem 8.2, limited experiments with the SrS for $r = 1.5$, 3, and 4 revealed no consistent winner among these variants and SS. Indeed, they suggest that these algorithms, and perhaps all the algorithms SrS with $r > 1$, might satisfy the Square Root and Bounded Waste Properties as well as the Sublinearity Property. Although we currently do not see how to prove these conjectures in general, we can show that the algorithms SrS satisfy the Bounded Waste Property when $r \ge 2$.

THEOREM 8.3. *If $r \ge 2$ and F is a bounded waste distribution with no multiply-occurring dead-end levels, then $EW_n^{SrS}(F) = O(1)$.*

PROOF. As in the proof of Theorem 3.4, we apply Hajek's Lemma. By an argument analogous to the one used in that proof, it is straightforward to show that the desired conclusion will follow if Hajek's Lemma can be shown to apply to the potential function

$$\phi(\bar{x}) = \left(\sum_{h=1}^{B-1} x_i^r\right)^{1/r}.$$

For this potential function, the Initial Bound Hypothesis applies since we begin with the empty packing. The Bounded Variation Hypothesis applies since for a given value y of $\phi(\bar{x})$, the maximum possible change in ϕ occurs when a single entry in $\bar{x}$ equals y and all the rest are 0, in which case ϕ can increase to at most $y + 1$ and decrease to no less than $y - 1$.

The main challenge in the proof is proving that the Expected Decrease Hypothesis applies. For this we need the following results, analogues of Lemmas 2.2, 3.6, and 3.7, used in the proof of Theorem 3.4.

LEMMA 8.4. *Let F be a perfectly packable distribution and $r \ge 2$. Then there is a constant d, depending only on r, such that if P is an arbitrary packing into bins of size B whose profile is given by the vector $\bar{x}$ with $\phi(\bar{x}) > 0$, i is an item randomly* *generated according to F, and $\bar{x}'$ is the profile of the packing resulting if i is packed into P according to SrS,*

$$E\left[\phi(\bar{x}')^r : \bar{x}\right] < \phi(\bar{x})^r + d\phi(\bar{x})^{r-2}.$$

PROOF. Note that $x_h \le \phi(\bar{x})$, $1 \le h \le B-1$ by definition. The result thus follows by (8.12) in the proof of Theorem 8.2. ■

LEMMA 8.5. *Let y and a be positive and $r \ge 2$. Then*

$$y - a \le \frac{y^r - a^r}{ra^{r-1}}. \tag{8.14}$$

PROOF. Consider the functions $f_a(y) = (y-a) - (y^r - a^r)/(ra^{r-1})$, $a > 0$. We need to show that for all $a > 0$, $f_a(y) \le 0$ whenever $y > 0$. But observe that the derivative

$$f_a'(y) = 1 - \frac{ry^{r-1}}{ra^{r-1}}$$

is greater than 0 if $y < a$, equals 0 if $y = a$, and is less than 0 if $y > a$. Thus $f_a(y)$ takes on its maximum value when $y = a$, in which case it is 0, as desired. ■

LEMMA 8.6. *Suppose F is a distribution with no multiply-occurring dead-end levels and $r \ge 2$. Let P be any packing that can be created by applying SrS to a list of items all of whose sizes are in U_F. If $\bar{x}$ is the profile of P and $\phi(\bar{x}) > r^2 B^{r+1/r}$ where B is the bin size, then there is a size $s \in U_F$ such that if an item of size s is packed by SrS into P, the resulting profile $\bar{x}'$ satisfies*

$$\phi(\bar{x}')^r \le \phi(\bar{x})^r - \frac{\phi(\bar{x})^{r-1}}{B^{(r^2-1)/r}}.$$

PROOF. Let x_h be the largest level count. By the definition of ϕ we have $\phi(\bar{x})^r \le Bx_h^r$ and hence $x_h \ge \phi(\bar{x})/B^{1/r} \ge r^2 B^r$. Thus h cannot be a multiply-occurring dead-end level and as in the proof of Lemma 3.7, there must be some $h' \ge h$ and size $s \in U_F$ such that $h' + s \le B$ and

$$\Delta \equiv x_{h'} - x_{h'+s} \ge x_h/B \ge \frac{\phi(\bar{x})}{B^{1+1/r}} \ge r^2 B^{r-1}.$$

Let y denote $x_{h'+s}$. Then if an item of size s were to be packed, we could reduce $\sum_{h=1}^{B-1} x_h^r$ by at least

$$(y+\Delta)^r - (y+\Delta-1)^r + y^r - (y+1)^r.$$

Using Taylor's Theorem as in the proof of Theorem 8.2 but with one fewer term in the expansions than in (8.10) and (8.11), we conclude the reduction is at least

$$\left[r(y+\Delta)^{r-1} - \frac{r(r-1)(y+\Delta-\theta_1)^{r-2}}{2}\right] - \left[ry^{r-1} + \frac{r(r-1)(y+\theta_2)^{r-2}}{2}\right],$$

where $0 < \theta_1, \theta_2 < 1$. But note that the amount we must subtract due to the two lower order terms is less than

$$\begin{aligned} r(r-1)(y+\Delta)^{r-2} = \frac{r(y+\Delta)^{r-1}}{(y+\Delta)/(r-1)} &\le \frac{r(y+\Delta)^{r-1}}{\Delta/r} \\ &\le \frac{r(x_h)^{r-1}}{rB^{r-1}} = \left(\frac{x_h}{B}\right)^{r-1} \le \Delta^{r-1}. \end{aligned}$$

Since the higher order terms are $r(y+\Delta)^{r-1} - ry^{r-1} \geq r\Delta^{r-1}$, we can conclude that ϕ must decrease by at least

$$(r-1)\Delta^{r-1} \geq (r-1)\left(\frac{\phi(\bar{x})}{B^{1+1/r}}\right)^{r-1} \geq \frac{\phi(\bar{x})^{r-1}}{B^{(r^2-1)/r}}$$

as claimed. ■

To prove that ϕ satisfies the Expected Decrease Hypothesis of Hajek's Lemma, we argue much as in the proof of Theorem 3.4. Since F is a bounded waste distribution, there is an $\epsilon > 0$ such that the process of generating items according to F is equivalent to generating items of the size s specified in Lemma 8.6 with probability ϵ and otherwise generating items according to a slightly modified perfectly packable distribution F'. By Lemmas 8.4 and 8.6, the expected increase in $\phi(\bar{x})^r$ is then at most

$$(1-\epsilon)d\phi(\bar{x})^{r-2} - \frac{\epsilon\phi(\bar{x})^{r-1}}{B^{(r^2-1)/r}},$$

which, assuming $\phi(\bar{x})$ is sufficiently large, is less than $-b\phi(\bar{x})^{r-1}$ for some constant $b > 0$ depending only on F and r. By Lemma 8.5 we thus have

$$E\left[\phi(\bar{x}') - \phi(\bar{x})\right] \leq -\frac{b\phi(\bar{x})^{r-1}}{r\phi(\bar{x})^{r-1}} = -\frac{b}{r}$$

and so the Bounded Decrease Hypothesis holds for ϕ, Hajek's Lemma applies, and we can conclude as in Theorem 3.4 that $EW_n^{SrS}(F) = O(1)$. ■

8.3 Combinatorial Variants

In this section we consider satisfying the Sublinearity Property with algorithms that do not depend on powers of counts. As our first two candidates, consider the algorithms that are in a sense the limits of the SrS algorithms as $r \to 1$ and $r \to \infty$, a promising approach since the SrS algorithms all satisfy the Sublinearity Property and may even satisfy the Square Root Property.

An obvious candidate for a limiting algorithm when $r \to 1$ is $S1S$, the algorithm that always tries to minimize $\sum_{h=1}^{B-1} N_P(h)$, i.e., the number of partially filled bins. To do this, we simply must never start a new bin if that can be avoided and must always perfectly pack a bin when possible (i.e., if the size of the item to be packed is s and there is a partially full bin with level $B-s$, we must place the item in such a bin). By itself this is not a completely defined algorithm, since one needs to provide a tie-breaking rule. If we use our standard tie-breaking rule (always chooses a bin with the highest acceptable level), note that $S1S$ reduces to the classic Best Fit algorithm. As already observed in the Introduction, Best Fit provably has linear expected waste for the bounded waste distributions $U\{8,11\}$ and $U\{9,12\}$, and empirically seems to behave just as poorly for many other such distributions [Coffman, Jr. et al. 1993]. We doubt that any other tie-breaking rule will do better. For instance, if we always choose the lowest available level when the item will not pack perfectly, we typically do much worse than Best Fit. Thus no $S1S$ algorithm is likely to satisfy the Sublinearity Property.

Taking the limit of SrS as $r \to \infty$ seems more promising. The algorithm $S\infty S$ proceeds as follows. Given an item a of size s, we choose the level h^* of the bin into which we place a as follows. Let $\Delta(h) = N_P(h) - N_P(h+s)$, $0 \le h \le B-s$, where we take $N_P(0) = 0$ and $N_P(B) = -1$ by convention. We first consider the set of bin levels $S_1 = \{h : 0 \le h \le B-s, \text{ and } \Delta(h) \ge 2\}$. If $S_1 \ne \phi$, we let h^* be that $h \in S_1$ that has the maximum value of $N_P(h)$, ties broken in favor of the h with the largest value of $\Delta(h)$, and ties for this value broken according to largest value of h. If S_1 is empty, we next consider the set $S_2 = \{h : 0 \le h \le B-s, \text{ and } \Delta(h) = 1\}$. If $S_2 \ne \phi$, then we let h^* be the largest $h \in S_2$. If S_2 is also empty, we know that for all h, $0 \le h \le B-s$, $\Delta(h) \le 0$. Let $h*$ be that h with the minimum value for $N_P(h+s)$, ties broken in favor of the largest value for $\Delta(h)$, and ties for this value broken according the the largest value of h. It is not difficult to see that for any fixed packing these are the choices that will be made by SrS for all sufficiently large values of r.

Experiments suggest that $S\infty S$ has bounded expected waste for $U\{8,11\}$ and $U\{9,12\}$ as well as all the bounded waste distributions $U\{h,100\}$, $1 \le h \le 98$. It still violates the Sublinearity Property, however. For example, $EW_n^{OPT}(U\{18 : 27, 100\}) = \Theta(\sqrt{n})$ but experiments clearly indicate that $S\infty S$ has linear waste for this distribution. A simpler distribution exhibiting the behavior is F with $B = 51$, $U_F = \{11, 12, 13, 15, 16, 17, 18\}$, and all sizes equally likely. Experiments convincingly suggest that $EW_n^{S\infty S}(F) = \Theta(n)$, but it is easy to see that this is a perfectly packable distribution, since both the first four and the last three item sizes sum to $B = 51$. Moreover, if one modifies F to obtain a distribution F' in which items of size 1 are added, but with only 1/10 the probability of the other items, one obtains a bounded waste distribution for which $S\infty S$ continues to have linear waste. Using other tie-breaking rules, such as preferring the lower level bin, appears only to make things worse. So no $S\infty S$ algorithm is likely to satisfy the Sublinearity Property.

Not surprisingly, the simpler combinatorial variants on $S\infty S$ that restrict attention solely to one of the two values $N_P(h)$ and $N_P(h+s)$ also fail. In the first of these, $S \max h$, we always place an item of size s in a bin whose level has maximum count among all levels no greater than $B-s$, assuming that the count for empty bins is by definition 0. In the second, $S \min h$, we place the item so as to minimize the count of the resulting level, assuming that the count for full bins is by definition -1. $S \max h$ has linear waste for $U\{8,11\}$ and $U\{9,12\}$, perhaps not surprising since even if the item to be packed would perfectly fill a bin, $S \max h$ may well choose not to do this. $S \min h$ is better, seeming to handle the $U\{j,k\}$ appropriately. However, it has linear waste on the same three perfectly packable/bounded waste distributions mentioned above on which $S\infty S$ also failed. Perhaps surprisingly, its constants of proportionality appear to be better than those for $S\infty S$ on the first two of these distributions. This may be because, unlike the latter algorithm, it will choose a placement that perfectly packs a bin when this is possible.

Indeed, perfectly packing a bin when that is possible would seem like an inherently good idea. We know that it is not *necessary* to do this, since SS does not always do it, but how could it hurt? Let *perfectSS* be the algorithm that places the current item so as to perfectly pack a bin if this is possible, but otherwise places it so as to minimize $ss(P)$. Surely this algorithm should do just as well as SS. Surprisingly, there are cases where this variant too violates the Sublinearity Property.

Consider the distribution F with bin size $B = 10$, $U_F = \{1, 3, 4, 5, 8\}$, $p(1) = p(3) = p(5) = 1/4$, and $p(4) = p(8) = 1/8$. This is a perfectly packable distribution, as the probability vector can be viewed as a convex combination of the perfect packing configurations $(8, 1, 1)$, $(4, 3, 3)$, and $(5, 5)$. However, experiments show that *perfectSS* has linear waste for this distribution (as does $S \min h$ but not $S\infty S$). Why does this happen? Note that essentially all the items of size 1 must be used to fill the bins that contain items of size 8. Thus whenever a 1 arrives and there is a bin of level 8, we need to place the 1 in such a bin. Unfortunately, *perfectSS* will prefer to put that 1 in a bin with level 9 if such a bin exists, and bins with level 9 can be created in other ways than simply with an 8 and a 1. Three 3's or a 5 and a 4 will do. On average this happens enough times to ruin the packing. (The count for level 9 never builds up to inhibit the nonstandard creation of such bins because level 9 bins keep getting filled by 1's.) Standard SS avoids this problem and has $\Theta(\sqrt{n})$ expected waste because it allows the counts for levels 8 and 9 to grow roughly as $\sqrt{n}$, with the latter being roughly half the former. This means that placing a 1 in a bin with level 8 is a downhill move, but creating a level 9 bin by any other means is an uphill move.

8.4 Variants Designed for Speed

Our final class of alternatives to SS are designed to improve the running time, possibly at the cost of packing quality. Recall that J denotes the number of item sizes under F. The $\Theta(nB)$ running time for the naive implementation of SS can be improved to $\Theta(nJ)$ by maintaining for each item size $s \in U_F$ the list-of-lists data structure we introduced to handle items of size 1 in the implementation of algorithm SS^* described in Section 6. This approach unfortunately will not be much of an improvement over the naive algorithm for distributions F with large numbers of item sizes, and it remains an open problem as to whether SS (or any of the variants described above that satisfy the Sublinearity Property) can be implemented to run in $o(nB)$ time in general. However, if one is willing to alter the algorithm itself, rather than just its implementation, one can obtain more significant speedups. Indeed, we can devise algorithms that satisfy both the Square Root and Bounded Waste Properties and yet run in time $O(n \log B)$ or even $O(n)$ (although there will of course be a tradeoff between running time and the constants of proportionality on the expected waste).

We shall first describe the general algorithmic approach and prove that algorithms that follow it will satisfy the two properties. We will then show how algorithms of this type can be implemented in the claimed running times. The key idea is to use data structures for each item size, as in the $O(nJ)$ implementation mentioned above, but only require that they be approximately correct (so that we need not spend so much time updating them). In particular, we maintain for each item size s a set of local values $N_{P,s}(h)$ for the counts $N_P(h)$, and only require these local counts satisfy

$$\left| N_P(h) - N_{P,s}(h) \right| \le \delta \tag{8.15}$$

for some constant δ. When an item of size s arrives, we place it so as to minimize $ss_s(P) = \sum_{i=1}^{B-1} N_{P,s}(h)^2$, subject only to the additional constraint that we cannot place the item in a bin with local count δ or less, since there is no guarantee that such bins exist. Let $ApproxSS_\delta$ be an algorithm that operates in this way.

LEMMA 8.7. *Suppose F is a perfectly packable distribution with bin size B, P is a packing into bins of size B, $\delta \geq 0$, and x is an item randomly generated according to F. Then if x is packed according to $ApproxSS_\delta$, the expected increase in $ss(P)$ is at most* $10\delta + 3$.

PROOF. We first need a generalization of Claim 2.3 from the proof of Lemma 2.2:

CLAIM 8.8. *Suppose F is a perfectly packable distribution with bin size B and $\delta \geq 0$. Then there is an algorithm A_F such for any packing P into bins of size B, if an item x is randomly generated according to F, A_F will pack x in such a way that x does not go in a partially filled bin with a level h for which $N_P(h) \leq \delta$ and yet for each level h with $N_P(h) > \delta$, $1 \leq h \leq B-1$, the probability that $N_P(h)$ increases is no more than the probability that it decreases.*

This is proved by a simple modification of the proof of Claim 2.3 to require that for each optimal bin the items are ordered so that all the levels S_1 through $S_{last(Y)}$ have counts greater than δ and none of the levels $S_{last(y)} + s(y_i)$ do for $i > last(y)$.

Claim 8.8 implies that the expected increase in $ss(P)$ under A_F is at most $2\delta+2$: If a count greater than δ changes, the proof of Lemma 2.2 implies that the expected increase in $ss(P)$ is at most 1. Counts of δ or less can only increase, but in this case $ss(P)$ can increase by no more than $2\delta + 1$. At most two counts can change during any item placement, and at most one of them can be a count of δ or less. Thus the expected change in $ss(P)$ obeys the claimed bound, and if SS_δ is the algorithm that places items so as to minimize $ss(P)$ subject to the constraint that no item can be placed in a partially filled bin whose level's count is δ or less, we can conclude that the expected increase in $ss(P)$ when SS_δ places an item generated according to F is also at most $2\delta + 2$.

So consider what happens when $ApproxSS_\delta$ packs an item with size $s \in U_F$. Suppose that placement is into a bin of level h, and that $N_P(h+s) - N_P(h) = d$. Note that by Lemma 1.2 the smallest increase in $ss(P)$ this can represent is $2d+1$. Now by (8.15) we must have $N_{P,s}(h+s) - N_{P,s}(h) \leq d+2\delta$ and so the move chosen by $ApproxSS_\delta$ must place the item in a bin of level h' satisfying $N_{P,s}(h'+s) - N_{P,s}(h') \leq d+2\delta$. But then, again by (8.15), we must have $N_P(h'+s) - N_P(h') \leq d+4\delta$ and hence, again by Lemma 1.2, $ss(P)$ can increase by at most $2d + 8\delta + 2$, or at most $8\delta + 1$ more than the increase under SS_δ. Since the expected value for the latter was at most $2\delta + 2$, the Lemma follows. ■

THEOREM 8.9. *For any $\delta \geq 0$,*

(a) *If F is a perfectly packable distribution, then $EW_n^{ApproxSS_\delta}(F) = O(\sqrt{n})$.*

(b) *If F is a bounded waste distribution with no multiply-occurring dead-end levels, then $EW_n^{ApproxSS_\delta}(F) = O(1)$.*

(c) *Suppose $ApproxSS'_\delta$ is the algorithm that mimics $ApproxSS_\delta$ except that it never creates a bin that, based on the item sizes seen so far, has a dead-end level, unless this is unavoidable, in which case it starts a new bin. Then this algo-*

rithm has $EW_n^{ApproxSS'_\delta}(F) = O(1)$ *for* all *bounded waste distributions, as well as* $EW_n^{ApproxSS'_\delta}(F) = O(\sqrt{n})$ *for all perfectly packable distributions.*

PROOF. Note that for any fixed δ, $10\delta + 3$ is a constant, and having a constant bound on the expected increase in $ss(P)$ was really all we needed to prove the above results for SS and SS'. Thus the above three claims all follow by essentially the same arguments we used for SS and SS', with constants increased appropriately to compensate for property (8.15). ■

Let us now turn to questions of running time.

LEMMA 8.10. *Suppose* $t \geq 1$ *is an integer. There are implementations of* $ApproxSS_{tB}$ *and* $ApproxSS'_{tB}$ *that work for all instances with bin size* B *and run in time* $O(B^2 + n(1 + (\log B)/t))$*, where the hidden constants do not depend on* t*,* B*, or* n*.*

PROOF. We shall describe an implementation for $ApproxSS_{tB}$. The implementation for $ApproxSS'_{tB}$ is almost identical except for the requirement that we keep track of the dead-end levels and avoid creating bins with those levels when possible, which we already discussed in Section 3.2 and which cumulatively takes at most $O(B^2)$ time.

Our implementations maintain a data structure for each item size s encountered, the data structure being initialized when the size is first encountered. We are unfortunately unable to use the list-of-list data structure involved in the implementation of SS^*, since the efficiency of that data structure relied on the fact that counts could only change by 1 when they were updated. Now they may change by as much as tB. Therefore we use a standard priority queue for the up to B possible levels h of bins into which an item of size s might be placed. Here the "possible levels" for s are 0 together with all those h such that $h + s \leq B$ and $N_{P,s}(h) > tB$. The levels are ranked by the increase in $ss_s(P)$ that would result if an item of size s were packed in a bin of level h. We can use any standard priority queue implementation that takes $O(1)$ time to identify an element with minimum rank and $O(\log B)$ to delete or insert an element. The priority queue for size s_j is created when the first item of this size is seen. This will take $O(B)$ time per queue, for a total of at most $O(B^2)$.

When we pack an item of size s, we first identify the "best" level h for it as specified by the priority queue for s. We then place x in a bin of level h and update the global counts $N_P(h)$ and $N_P(h+s)$. This all takes $O(1)$ time. Local counts are not immediately changed when an item is packed. Local count updates are performed more sporadically, and initiated as follows. We maintain a counter $c(h)$ for each level h, initially set to 0. This counter is incremented by 1 every time $N_P(h)$ changes and reset to 1 whenever it reaches the value $tB + 1$. Suppose the item sizes seen so far are $s_1, s_2, \ldots, s_j$. The local count $N_{P,s_i}(h)$ is updated only when the new value of $c(h)$ satisfies $c(h) \equiv 0 (\text{mod } t)$ and $i = c(h)/t$. Note that this means that $N_P(h)$ changes only tB times between any two updatings of $N_{P,s_i}(h)$ and so (8.15) is satisfied for $\delta = tB$.

Whenever $N_{P,s}(h)$ is updated, we make up to two changes in the priority queue for s, each of which involves one or two insertions/deletions and hence takes $O(\log B)$

time: First, if $h + s \leq B$ we may need to update the priority queue entry for h. If h is in the queue but now $N_{P,s}(h) \leq tB$, then we must delete it from the queue. If it is not in the queue but now $N_{P,s}(h) > tB$ we must insert it. Finally, if it is in the queue and $N_{P,s}(h) > tB$, but its rank is not the correct value (with respect to $N_{P,s}(h)$ and $N_{P,s}(h+s)$), then it must be deleted and reinserted with the correct value. Similarly, if $h - s \geq 0$, then we may have to update the entry for $h - s$.

It is easy to verify that the above correctly implements $ApproxSS_{tB}$. The overall running time is $O(B^2)$ for initializing the priority queues, $O(n)$ for packing and updating the true counts $N_P(h)$ and $O((n/t)\log B)$ for updating local counts and priority queues, as required. ■

THEOREM 8.11. *There exist algorithms A1SS, A2SS, A1SS' and A2SS' such that*

(a) *All four satisfy the Square Root and Bounded Waste Properties.*

(b) *A1SS' and A2SS' have bounded expected waste for* all *bounded waste distributions.*

(c) *A1SS and A1SS' run in time $O(B^2 + n\log B)$.*

(d) *A2SS and A2SS' run in time $O(B^2 + n)$.*

PROOF. Given Theorem 8.9, it is easy to get algorithms with the above properties from Lemma 8.10: If we take $t = 1$ we get running time $O(n \log B)$ and if we take $t = \log B$ we get running time $O(n)$. (Of course the bigger the value of t, the worse the constants of proportionality for the expected waste.) ■

We can also devise fast analogues of Section 6's distribution-specific algorithms SS^F that always have $ER_\infty^A(F) = 1$, even for distributions whose optimal expected waste is linear. This however involves more than just applying the approximate data structures described above. The $O(nB)$ running times for the SS^F algorithms derive from two sources, only one of which (the need for $\Theta(B)$ time to pack an item) is eliminated by using the approximate data structures. The second source of $\Theta(nB)$ time is the need to possibly pack $\Theta(nB)$ imaginary items of size 1.

To avoid this obstacle, we need an additional idea. Recall that SS^F attains $ER_\infty^{SS^F}(F) = 1$ by simulating the application of SS to a perfectly packable distribution F' derived from F. The modified distribution F' was constructed using the optimal value $c(F)$ for the linear program of Section 5. Distribution F' was equivalent to generating items according to F with probability $1/(1 + c(F))$ and otherwise generating an (imaginary) item of size 1.

Our new approach uses more information from the solution to the LP. Let $v(j, h)$, $1 \leq j \leq J$ and $0 \leq h \leq B - 1$, be the variable values in an optimal solution for the LP for F. For $1 \leq h \leq B - 1$ define

$$\Delta_h \equiv \sum_{j=1}^{J} v(j, h - s_j) - \sum_{j=1}^{J} v(j, h).$$

Note that Δ_h is essentially the percentage of partially filled bins in an optimal packing whose gap is of size $B - h$. Let $T = \sum_{h=1}^{B-1} \Delta_h$ and note that we must have $T \leq 1$. Our new algorithm uses SS to pack the modified distribution F'' obtained as follows. With probability $1/(1+T)$ we generate items according to the original distribution F. Otherwise (with probability $T/(1+T)$) we generate "imaginary" items according to the distribution in which items of size s have probability Δ_{B-s}. It is not difficult to show that this is a perfectly packable distribution and that the expected total size of the imaginary items is $c(F)$, as in SS^F. Now, however, the number of imaginary items is bounded by n, so the time for packing them is no more than that for packing the real items, and hence can be $O(n \log B)$ or $O(n)$ as needed.

One can construct a learning algorithm SS^{**} based on these variants just as we constructed the learning algorithm SS^* based on the original SS^F algorithms. We conjecture that SS^{**} will satisfy the same general conclusions as listed for SS^* in Theorem 6.3. The proof will be somewhat more complicated, however, and so we leave the details to interested readers.

We should note before concluding the discussion of fast variants of SS that our results on this topic are probably of theoretical interest only. A complicated $O(B^2 + n \log B)$ algorithm like $ApproxSS_B$ would be preferable to an $O(nB)$ or $O(nJ)$ implementation of SS only when B is fairly large, presumably well over 100. However, the constants involved in the expected waste produced by $ApproxSS_B$ are substantial in this case. For example, consider the bounded waste distribution $U\{400, 1000\}$. For this distribution we simulated $ApproxSS_{400}$, which should produce less waste than $ApproxSS_B$ when $B = 1000$ and should only be 2.5 times slower. For $n = 100,000$, $ApproxSS_{400}$ typically uses 100,000 bins, i.e., one per item and roughly 5 times the optimal number, even though Theorem 8.11 says that the expected waste is asymptotically $O(1)$. On the other hand, Best Fit, which itself runs in time $O(n \log B)$ but is conjectured to have linear expected waste for this distribution, empirically uses roughly 0.3% more bins than necessary. (SS uses roughly 0.25%.) Things have improved by the time $n = 10,000,000$, but not enough to change the ordering of algorithms. Now $ApproxSS_{400}$ uses only roughly 9.8% more bins than necessary, while Best Fit uses roughly 0.28%. SS is down to an average excess of 0.0025%. This consists of roughly 50 excess bins for SS versus 5,600 for BF versus 200,000 for $ApproxSS_{400}$ It does not appear that $ApproxSS_{400}$ is likely to catch BF until the value of n becomes very much larger.

9. CONCLUSIONS AND OPEN QUESTIONS

In this paper we have discussed a collection of new, nonstandard, and surprisingly effective algorithms for the classical one-dimensional bin packing problem. We have done our best to leave as few major open problems as possible, but several interesting ones do remain:

—Can SS itself be implemented to run in time $o(nB)$, so that we aren't forced to use the approximate versions described in the previous section?

—What is $\max\{ER_\infty^{SS}(F) : F \text{ is a discrete distribution}\}$? The results of Section 4 only show that this maximum is at least 1.5 and no more than 3.0, an upper bound that has subsequently been improved to 2.777... by Csirik et al. [2005]. A related question concerns the asymptotic worst-case performance ratio for SS, which currently is only known to lie between 2.0 and 2.777... We conjecture that in both cases the true answer lies nearer the lower bound.

—Is our conjecture correct that SrS satisfies both the Square Root and Bounded Waste Properties for all $r > 1$? Is there any polynomial-time algorithm that satisfies the Sublinearity Property and does not involve at least implicitly computing the powers of counts?

—Can one obtain a meaningful theoretical analysis of the constants of proportionality involved in the expected waste rates for particular distributions and the various bin packing algorithms we have discussed? Empirically we have observed wide differences in these constants for algorithms that, for example, both have bounded expected waste for a given distribution F, so theoretical insights here may well be of practical value.

—Is there an effective way to extend the Sum-of-Squares approach to continuous distributions while preserving its ability to get sublinear waste when the optimal waste is sublinear?

Finally, there is the question of the extent to which approaches like that embodied in the Sum-of-Squares algorithm can be applied to other problems. A first step in this direction is the adaptation of SS to the bin covering problem in Csirik et al. [2001]. In bin covering we are given a set of items and a bin capacity B, and must assign the items to bins so that each bin receives items whose total size is *at least* B and the number of bins packed is maximized. Here "waste" is the total *excess* over B in the bins and the class of "perfectly packable distributions" is the same as for ordinary bin packing. The interesting challenge here becomes to construct algorithms that have good worst- and average-case behavior for distributions that aren't perfectly packable, while still having $O(\sqrt{n})$ expected waste for perfectly packable distributions. For details, see Csirik et al. [2001].

The results for bin covering suggest that the Sum-of-Squares approach may be more widely applicable, but bin covering is still quite close to the original bin packing problem. Can the Sum-of-Squares approach (or something like it) be extended to problems a bit further away?

ACKNOWLEDGMENTS

The authors thank the referees for their thorough reading of this very long paper, which led to many useful suggestions for improving the presentation and helped us avoid several significant technical imprecisions.

REFERENCES

Albers, S. and Mitzenmacher, M. 1998. Average-case analyses of first fit and random fit bin packing. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms.* SIAM, Philadelphia, Pa., 290–299.

Applegate, D. L., Buriol, L., Dillard, B., Johnson, D. S., and Shor, P. W. 2003. The cutting-stock approach to bin packing: Theory and experiments. In *Proceedings of the 5th Workshop on Algorithm Engineering and Experimentation.* SIAM, Philadelphia, Pa., 1–15.

Coffman, Jr., E. G., Courcoubetis, C., Garey, M. R., Johnson, D. S., McGeoch, L. A., Shor, P. W., Weber, R. R., and Yannakakis, M. 1991. Fundamental discrepancies between average-case analyses under discrete and continuous distributions. In *Proceedings 23rd Annual ACM Symposium on Theory of Computing.* ACM, New York, 230–240.

Coffman, Jr., E. G., Courcoubetis, C., Garey, M. R., Johnson, D. S., Shor, P. W., Weber, R. R., and Yannakakis, M. 2000a. Bin packing with discrete item sizes, Part I: Perfect packing theorems and the average case behavior of optimal packings. *SIAM J. Disc. Math. 13*, 384–402.

Coffman, Jr., E. G., Courcoubetis, C., Garey, M. R., Johnson, D. S., Shor, P. W., Weber, R. R., and Yannakakis, M. 2002b. Perfect packing theorems and the average-case behavior of optimal and online bin packing. *SIAM Review 44*, 95–108. Updated version of Coffman, Jr. et al. [2000a].

Coffman, Jr., E. G., Johnson, D. S., McGeoch, L. A., Shor, P. W., and Weber, R. R. Bin packing with discrete item sizes, Part III: Average case behavior of FFD and BFD. In preparation.

Coffman, Jr., E. G., Johnson, D. S., Shor, P. W., and Weber, R. R. 1993. Markov chains, computer proofs, and average-case analysis of Best Fit bin packing. In *Proceedings 25th Annual ACM Symposium on Theory of Computing*. ACM, New York, 412–421.

Coffman, Jr., E. G., Johnson, D. S., Shor, P. W., and Weber, R. R. 1997. Bin packing with discrete item sizes, part II: Tight bounds on first fit. *Random Structures and Algorithms 10*, 69–101.

Coffman, Jr., E. G. and Lueker, G. S. 1991. *An Introduction to the Probabilistic Analysis of Packing and Partitioning Algorithms*. Wiley & Sons, New York.

Courcoubetis, C. and Weber, R. R. 1990. Stability of on-line bin packing with random arrivals and long-run average constraints. *Probability in the Engineering and Informational Sciences 4*, 447–460.

Csirik, J., Johnson, D. S., and Kenyon:, C. 2001. Better approximation algorithms for bin covering. In *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, Philadelphia, Pa., 557–566.

Csirik, J., Johnson, D. S., and Kenyon, C. 2005. On the worst-case performance of the sum-of-squares algorithm for bin packing. E-Print `arXiv:cs.DS/0509031`, arXiv.org e-Print archive (`http://arxiv.org/archive/cs`).

Csirik, J., Johnson, D. S., Kenyon, C., Shor, P. W., and Weber, R. R. 1999. A self organizing bin packing heuristic. In *Proceedings 1999 Workshop on Algorithm Engineering and Experimentation*, M. Goodrich and C. C. McGeoch, Eds. Lecture Notes in Computer Science 1619, Springer-Verlag, Berlin, Germany, 246–265.

Garey, M. R. and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman, New York, New York.

Gilmore, P. C. and Gomory, R. E. 1961. A linear programming approach to the cutting stock problem. *Oper. Res. 9*, 849–859.

Gilmore, P. C. and Gomory, R. E. 1963. A linear programming approach to the cutting stock program — Part II. *Oper. Res. 11*, 863–888.

Grötschel, M., Lovasz, L., and Schrijver, A. 1981. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica 1*, 169–197.

Hajek, B. 1982. Hitting-time and occupation-time bounds implied by drift analysis with applications. *Adv. Appl. Prob. 14*, 502–525.

Hoeffding, W. 1963. Probability inequalities for sums of bounded random variables. *J. American Statistical Association 58*, 13–30.

Johnson, D. S., Demers, A., Ullman, J. D., Garey, M. R., and Graham, R. L. 1974. Worst-case performance bounds for simple one-dimensional packing algorithms. *SIAM J. Comput. 3*, 299–325.

Karmarkar, N. and Karp, R. M. 1982. An efficient approximation scheme for the one-dimensional bin packing problem. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*. IEEE Computer Society, Los Alamitos, Calif., 312–320.

Karp, R. M. and Papadimitriou, C. H. 1982. On the linear characterization of combinatorial optimization problems. *SIAM J. Comput. 11*, 620–632.

Kenyon, C. and Mitzenmacher, M. 2002. Linear waste of best fit bin packing on skewed distributions. *Random Structures and Algorithms 20*, 441–464. Preliminary version in *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, IEEE Computer Society, Las Alamitos, Calif., 2000, 582–589.

Kenyon, C., Rabani, Y., and Sinclair, A. 1998. Biased random walks, Lyapunov functions, and stochastic analysis of best fit bin packing. *J. Algorithms 27*, 218–235. Preliminary version in *Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, SIAM, Philadelphia, Pa., 1996, 351–358.

Rhee, W. T. 1988. Optimal bin packing with items of random sizes. *Math. Oper. Res. 13*, 140–151.

Rhee, W. T. and Talagrand, M. 1993a. On line bin packing with items of random size. *Math. Oper. Res. 18*, 438–445.

Rhee, W. T. and Talagrand, M. 1993b. On line bin packing with items of random sizes – II. *SIAM J. Comput. 22*, 1251–1256.

Vaidya, P. M. 1989. Speeding-up linear programming using fast matrix multiplication. In *Proceedings of the 30th Annual IEEE Symposium on Foundations of Computer Science*. IEEE Computer Society, Los Alamitos, CA, 332–337.

Valério de Carvalho, J. M. 1999. Exact solutions of bin-packing problems using column generation and branch and bound. *Annals of Operations Research 86*, 629–659.

van Vliet, A. 1992. An improved lower bound for on-line bin packing algorithms. *Inf. Process. Lett. 43*, 277–284.

Received October 2002; revised September 2005; accepted September 2005