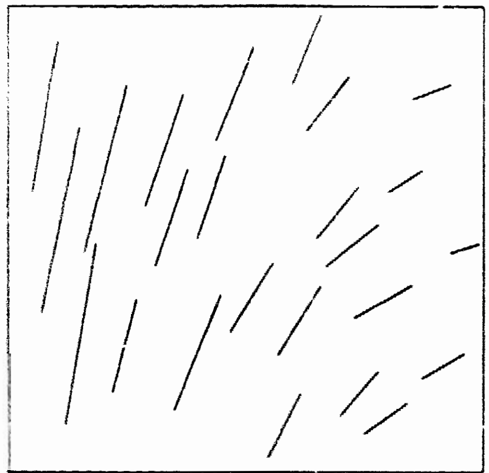

Figure 6.2.6: A matching satisfying Condition 2.

grid points in a row. This matches all but $O(n^{1/3}\log^{1/2} n)$ of the + points to − points. (See Figure 6.2.6.) This matching is up-right since we matched each point horizontally $7l$ grid points to the right, and the edges used in matching the + points and − points to the grid points have length at most l. We have thus matched each point at least $5l$ grid points to the right. Similarly, we have matched all the − points up (except those within $7l$ of the right border, which are unmatched anyway). This matching also satisfies Condition 2. If there are two edges (a, b) and (c, d) with b to the left of c, then the grid points matched to a and c are separated by at least $5l$ grid points, since the grid point matched to c is at worst $2l$ left of that matched to b. There is a possible difference of $2l$ between the vertical length of an edge and the corresponding grid point edge. Since a is at least $5l$ grid points to the left of c, we have that the grid point edge corresponding to (a, b) is at least $5l$ longer than the one for (c, d), so that (a, b) is at least l longer than (c, d). This matching therefore satisfies Condition 2. Thus MFF, and also FF, has wasted space at most $O(n^{2/3}\log^{1/2} n)$. ∎

6.3. The Lower Bound

We prove that the algorithm First Fit gives an expected value of $\Omega(n^{2/3})$ wasted space when packing items which are uniformly distributed on $[0, 1]$.

Proof: In this proof we will be considering two different sequences; the sequence of items which are to be packed (this does not change) and the sequence of bins which are partially full (this changes as items are packed). To avoid confusion, we will call the sequence of items the *list* of items and the sequence of bins the *queue* of bins. A ***subsequence*** of one of these sequences will be a subset of its elements, in the same order as they were in the original sequence. If all these elements are adjacent, we call it a *contiguous subsequence.*

Consider an item that is being packed by First Fit. Into what bins can this item go? It can only go into a bin if this bin is less full than all previous bins in the queue (otherwise it would have been packed in one of the previous bins). We will call this subsequence of bins that have more empty space than all preceding bins the ***greedy decreasing subsequence*** of bins. This subsequence is the decreasing subsequence formed greedily, i.e., by starting with the first bin and always adding the next bin in the queue which will make the subsequence decreasing. This subsequence ends with the first empty bin. First Fit can only add items to bins in the greedy decreasing subsequence.

For this proof, we will look at the tail of the greedy decreasing subsequence starting with the first bin less than $\frac{2}{3}$ full. We call this tail Q. No item less than $\frac{1}{3}$ can be in any bin later in the queue than the first bin of Q, since any items less than $\frac{1}{3}$ would be put in the first bin of Q or an earlier bin. Thus, any 2-item bin after the first bin of Q is more than $\frac{2}{3}$ full, and thus not in Q. This shows that all the bins in Q, except possibly for the first one, contain only one item.

We consider the items which fill bins. Since we almost certainly use at least

$n/2$ bins (the optimal packing almost certainly contains more bins than this [Lu]), at least half of all items will be the top item of a bin in the final packing. Thus, if we can show that $\frac{1}{2}+\delta$ of the time an item leaves at least ϵ empty space in a bin, we can conclude that $\delta n/2$ bins in the final packing have ϵ empty space.

Assume there are k bins in Q. Any item larger than $\frac{1}{3}$ must be placed in a bin in Q. An item which will fill a bin to higher than $1-\frac{1}{12k}$ must have size within $1/12k$ of the amount of empty space in a bin in Q, so the probability of an item filling a specific bin to within $1/12k$ of full is at most $1/12k$. Since there are only k bins in Q, the probability of an item filling one of these to within $1/12k$ of full is less than $k \cdot \frac{1}{12k} = \frac{1}{12}$. Since an item goes into a bin in Q with probability $\frac{2}{3}$, there is at least a $\frac{2}{3}-\frac{1}{12}=\frac{7}{12}$ probability that an item will leave a space of $\frac{1}{12k}$ or greater in a bin. Thus, if there are never more than k bins in Q, with high probability the final packing will have $\frac{1}{13}n$ bins with at least $\frac{1}{12k}$ empty space.

We now use Hammersley's result [Ha] that with probability at least $1-e^{-\alpha n}$, a random sequence of length n has no decreasing subsequence longer than $cn^{1/2}$, for some $c>2$ and some α. After the first bin of Q, the bins in Q each have only one item in them. These bins were filled in their order in the queue of bins, Thus, these items form a decreasing subsequence of the list of items. Let P be the contiguous subsequence of the list of items starting with the item packed in the first bin of Q and ending with the item packed in the last non-empty bin of Q. All the items in P having size greater than $\frac{2}{3}$ are packed in bins by themselves. Assume there are $n^{1/3}$ bins in Q. Applying Hammersley's result to each of the $n-n^{2/3}/\sqrt{c}$ contiguous subsequences of the list of items of length $n^{2/3}/\sqrt{c}$, we get that with probability at least $1-ne^{-\alpha n^{2/3}/\sqrt{c}}$ none of these contains a decreasing subsequence of length $n^{1/3}$. Thus, with high probability,

P is not contained in a consecutive subsequence of length $n^{2/3}/\sqrt{c}$, and thus has length at least $n^{2/3}/\sqrt{c}$.

Suppose the greedy decreasing subsequence Q is never longer than $n^{1/3}$. Then, by the preceding arguments, every item had a $\frac{7}{12}$ probability of leaving $\frac{1}{12}n^{-1/3}$ or more empty space in the bin it entered, so we have $\Omega(n^{2/3})$ wasted space in the final packing. On the other hand, if the subsequence Q were longer than $n^{1/3}$ at some time t, we have that with high probability, at time t there were $\Omega(n^{2/3})$ bins in P. With high probability $\frac{1}{7}$ of the items in P have size between $\frac{2}{3}$ and $\frac{5}{6}$. At time t these are all the only items in their bins. Thus, at time t there were $\Omega(n^{2/3})$ bins with at least $\frac{1}{6}$ empty space. If there ever were $\Omega(n^{2/3})$ of these bins, we will show that with high probability there still are, so we still have $\Omega(n^{2/3})$ wasted space.

Suppose that at some point we have $\Omega(n^{2/3})$ bins filled to $\frac{5}{6}$ or less. After this point, say we will be receiving $m < n$ more items. Any items larger than $\frac{1}{2}$ will start new bins, so we need enough small items to fill all these new bins and the $\Omega(n^{2/3})$ bins that were filled to less than $\frac{5}{6}$. With probability at least $1 - 1/n$, the empty space we need to fill in these new bins (created by large items) totals to $\frac{1}{4}m \pm O(\sqrt{n \log n})$ and the total size of the small items we can use to fill these totals to $\frac{1}{4}m \pm O(\sqrt{n \log n})$. Thus, we will with probability at least $1 - 1/n$ still have $\Omega(n^{2/3})$ wasted space. ∎

Chapter 7. Arbitrary On-line Algorithms

7.1. A Lower Bound

In this chapter, we prove a general lower bound for any on-line algorithm that does not know the number of items that will be input. We show that any algorithm packing items uniformly distributed on $[0,1]$ must waste $\Omega(\sqrt{n \log n})$ space. We do this by relating the performance of the algorithm to the rightward matching problem. We then use the lower bound of Ajtai, Komlós and Tusnády for this matching problem. This lower bound contrasts with off-line algorithms or on-line algorithms that are given the number of items in advance. These can achieve $O(\sqrt{n})$ average wasted space, which is within a constant factor of optimal.

Definition 7.1.1: An on-line algorithm for bin packing is one that packs each item as soon as it is received.

The model we will use is that the algorithm has no information about the number of items until it receives the last one. More specifically, we set up a distribution as follows:

1. Choose k from 1 to n at random.
2. Choose k items uniformly on $[0,1]$.
3. Input these items to the algorithm, ending with a "stop" signal after the kth item.

We now show the expected wasted space is $\Omega(\sqrt{n \log n})$.

Theorem 7.1.2: Let A be an on-line algorithm that receives k items uniformly distributed on $[0, 1]$ and is not given any information about the value of k until it has received the last item. Let k be chosen with equal probability from the integers between 1 and n. Then A must waste an expected value of $\Omega(\sqrt{n \log n})$ space.

Proof: For a random list L of n items, consider the wasted space that algorithm A produces when packing the first k items of the list. We will show the average of this wasted space over k must be with high probability $\Omega(\sqrt{n \log n})$. This proves the theorem, since choosing k first and then choosing a random sequence of length k is equivalent to first choosing a random sequence L of length n and then only looking at the first k items. We can choose k second because our conditions on the algorithm A require it to pack the initial items of a sequence in a way that does not depend on the number of items in the sequence or on the items which will be received later.

We give a lower bound on average wasted space by analyzing the rightward matching problem. To convert a list of items to a planar matching problem, we represent the items received by the algorithm by points in a unit square. The x-coordinate will be the size of the item. The y-coordinate will depend on the time the item was received. To fit the n items into the square, we put the jth item at distance j/n from the top. Next, we label the items larger than $\frac{1}{2}$ with '+' and those smaller than $\frac{1}{2}$ with '−'. We then fold the plane about the line $x = \frac{1}{2}$ (See Figure 5.2.2), so a + point with x-coordinate s will be moved so it has x-coordinate $1 - s$. For the time being consider only items with size between $\frac{1}{3}$ and $\frac{2}{3}$. We join every + point in this range to a − point in this range representing an item packed in the same bin, if there is such an item. This gives a bipartite matching M between + and − points.

We examine this bipartite matching M more closely. It is a matching on a bipartite graph G between the $-$ points in $(\frac{1}{3}, \frac{1}{2})$ and the $+$ points in $(\frac{1}{2}, \frac{2}{3})$. A $-$ point can only be matched to a $+$ point to the right of it, since otherwise the sizes of the corresponding items would sum to more than 1. Only one item larger than $\frac{1}{2}$ can fit in a bin, only two items in the range $(\frac{1}{3}, \frac{1}{2})$ can fit in a bin, and an item in $(\frac{1}{3}, \frac{1}{2})$ cannot fit in a bin with an item in $(\frac{2}{3}, 1)$. Thus, the total number of $+$ points plus half the number of unmatched $-$ points in G is a lower bound on the number of bins used. We prove the lower bound on the expected number of bins by giving a lower bound on the average number of unmatched $-$ points.

The matching M is a rightward matching in a unit square. By the results on rightward matching proved in Part I of this thesis, we have that the sum of the vertical lengths of the edges in a rightward matching is $\Omega(\sqrt{n \log n})$, where unmatched points are considered matched to the top or bottom of the square.

The sum of the vertical lengths of the edges in the matching is equal to the integral over t of the number of edges crossing the horizontal line $y = t$. The y-axis is time, and the number of edges crossing the line $y = t$ is the number of unmatched points at time t. Thus, this integral is the average number of unmatched points over time. The lower bound given in the lemma thus bounds the expected number of unmatched points. By applying this lemma to the points with x-coordinate greater than $\frac{1}{3}$, we get a $\Omega(\sqrt{n \log n})$ lower bound on the expected number of unmatched points in G. This proves Theorem 7.1.2. ∎

7.2. A Counter-example

In this section, we show that if we know in advance how many items there are to be packed, there is an on-line algorithm HBF (for Half Best Fit) with expected wasted space $\Theta(\sqrt{n})$. This contrasts with theorem 7.2. in the previous section, which gave a lower bound for on-line algorithms that did not known in advance how many items there were to be packed. The optimal packing wastes $\Theta(\sqrt{n})$ space on the average, so this is within a constant factor of optimal.

The algorithm HBF is very simple: pack the first $n/2$ items one item per bin, and then pack the remaining items using the algorithm Best Fit. We will show that HBF gives a better packing than an algorithm we will call MHBF, and then show that MHBF produce $\Theta(\sqrt{n})$ wasted space.

The algorithm MHBF packs the first $n/2$ items one per bin, and then uses a modification of BF to pack the remaining items. We will require in MHBF, however, that there are only two items per bin, and in a two-item bin, the first item is from the first half of the items and the second item is from the second half.

The algorithm MHBF treats the first half and the second half of the items differently. We will call an item from the first half a $+$ item, and an item from the second half a $-$ item. Let the list of items that MHBF packs be L.

We will be able to use the techniques we developed for MBF on the algorithm MHBF by modifying the items to obtain L'. The way the algorithm MHBF packs L will correspond to the way the algorithm MBF packs L'. If a $+$ item in L has size s, we replace it in the list L' with in item with size $1+s$. The $-$ items in L we leave alone. Now, the way MBF packs items in L' into bins of size 2 is the same as the way that MHBF packs items in L. The algorithm MBF will pack all the $+$ items of L' one per bin, since two of them do not fit into a bin. It will then pack the remaining items into the same bins that MHBF packs them, since the amount of space in bins containing $+$ items is the same in $MBF(L')$ and $MHBF(L)$. Now, by Lemma 5.2.5, $MBF(L')$ is an optimal up-right matching between $+$ items and $-$ items in L. However, all the $+$ items were put in before all the $-$ items, so any $-$ item can be matched to a $+$ item to its right.

We next show that $\#HBF(L) \leq \#MHBF(L)$ for any list L. Again, we use

.9		.75	.6		.3		.15
+	−	+	+	−	+	−	+
	.15			.5		.8	

Figure 7.2.1:

Lemma 5.2.1. This lemma says that if you delete an item from a list L to obtain L', then $\#MBF(L) \geq \#MBF(L')$. The argument of the preceding paragraph shows that this lemma also applies to MHBF. Let L' be the list obtained from L by deleting all $-$ items which are not the first $-$ item to be put in a bin by the algorithm HBF. Then,

$$\#HBF(L) = \#MHBF(L') \leq \#MHBF(L).$$

We now show that $MHBF$ uses on the average $\Theta(\sqrt{n})$ bins when packing items uniformly distributed on $[0,1]$. This translates into a matching problem. We are given $n/2$ $-$ points and $n/2$ $+$ points with values chosen uniformly between 0 and 1. A $-$ point can only be matched to a $+$ point if their values sum to less than 1. We will show that the expected number of unmatched points in an optimal matching is $\Theta(\sqrt{n})$.

By Hall's Theorem, the number of unmatched points is

$$\max_{0 \leq x \leq 1} |\{Q \in \mathcal{P}^+ : w(Q) \geq 1 - x\}| - |\{P \in \mathcal{P}^- : w(P) \geq x\}|.$$

We put the points on a line, with a $-$ point P with value $w(P)$ put down at the point $w(P)$, and a $+$ point Q with value $w(Q)$ put down at the point $1 - w(Q)$. (See Figure 7.2.1.) Now, the number of unmatched points is just the maximum number of excess $+$ points in an initial segment of the line. This has expected value $\Theta(\sqrt{n})$. ∎

Conclusions and Open Problems

In this thesis we placed several planar matching problems that had previously investigated in various contexts into one framework. We then found new tight bounds for two of these problems: up-right matching and maximum edge length matching. We then applied these problems to give bounds on the performance of two on-line algorithms for bin packing: First Fit and Best Fit. We also used the matching problems to prove a lower bound on the performance of any on-line algorithm.

Of the four bin packing problems we defined, tight bounds on three are known. The fourth, rightward planar matching, is still open. If a lower bound of $\Omega(\sqrt{n}\log^{3/4} n)$ were shown for this rightward matching, this would prove that there is no on-line algorithm for bin packing that produces $o(\sqrt{n}\log^{1/2} n)$ wasted space, which would show that Best Fit is an optimal on-line algorithm. Even if the answer to rightward matching were lower, Best Fit could still be an optimal on-line algorithm, since finding an optimal rightward matching could require information not available to on-line algorithms. This is a very interesting topic for further study.

Another bound presented in this paper that could be improved is the result on the wasted space of the First Fit algorithm. Neither the performance of the algorithm nor the number of unmatched points in the corresponding matching problem is known exactly. Furthermore, it is not clear that these two results are the same.

Planar matching problems have been useful for analyzing the average case behavior of algorithms in at least three cases. They were used to analyze multi-dimensional bin packing in [KLM]. In this thesis, we used them to analyze on-line algorithms for bin packing. Finally, Coffman and Leighton have used them to analyze a dynamic allocation algorithm [CL]. It is quite possible that planar matching problems can be used to analyze the average-case behavior of still other algorithms. This could be a fruitful topic of investigation.

Dudley [Du1,Du2] has proven a result on the discrepancies of multi-dimensional sets which, in two dimensions, becomes the up-right matching problem. His bounds on these discrepancies are not tight, but differ by a $\log n$ factor. It is possible that the techniques used in this paper could generalize to prove tight bounds on these problems. This is worth investigating.

Bibliography

[AKT] M. Ajtai, J. Komlós, and G. Tusnády, "On optimal matchings," Combinatorica 4, pp. 259-264, 1983.

[BJLM] J.L. Bentley, D.S. Johnson, F.T. Leighton, and C. C. McGeoch, "An experimental study of bin packing," *Proc. 12st Ann. Allerton Conf. on Communication, Control, and Computing*, University of Illinois, Urbana IL, pp. 51-60, 1983.

[BJLMM] J.L. Bentley, D.S. Johnson, F.T. Leighton, C.C. McGeoch, and L. McGeoch, "Some unexpected expected behavior results for bin packing," *Proc. 16th ACM Symp. on Theory of Computing*, pp. 279-288, 1984.

[BJLMM2] J.L. Bentley, D.S. Johnson, F.T. Leighton, C.C. McGeoch, and L. McGeoch, "Open problems in bin packing, working paper," Technical report, Bell Laboratories, Murray Hill, NJ, 1984.

[Br] D.J. Brown, "A lower bound for on-line one-dimensional bin packing algorithms," Technical Report R86, Coordinated Science Lab., U. of Illinois, Urbana, IL, 1979.

[CGJ] E.G. Coffman, Jr., M.R. Garey, and D.S. Johnson, "Approximation algorithms for bin packing — an updated survey," to appear.

[CHSY] E.G. Coffman, Jr., M. Hofri, K. So, and A.C. Yao, "A stochastic model of bin packing", Information and Control 44, pp. 105-115, 1980.

[CL] E.G. Coffman, Jr and F.T. Leighton, "A provably efficient algorithm for dynamic storage allocation," work in progress.

[Du1] R.M. Dudley, "A course on empirical processes", *École d'été de Probabilités de Saint-Flour XII, 1982*, Lecture Notes in Math. No. 1097, pp. 1-142, Springer Verlag, NY, 1984. (The relevant part is section 8.4.)

[Du2] R.M. Dudley, "Empirical and Poisson processes on classes of sets or functions too large for central limit theorems," Z. Wahrsheinlichkeitstheorie verw. Gebiete 61, pp. 355-368, 1982.

[FL] W. Fernandez de la Vega and G.S. Lueker, "Bin packing can be solved within $1 + \epsilon$ in linear time," Combinatorica 1, pp. 349-355, 1981.

[Fr] G.N. Frederickson, "Probabilistic analysis for simple one- and two-dimensional bin packing algorithms," Inf. Proc. Letters 11, pp. 156-161.

[Ha] J.M. Hammersley, "A few seedlings of research," *Proc. 6th Berkeley Symp. on Mathematical Statistics and Probability*, U. of California, Berkeley, CA, Vol. 1, pp. 345-394, 1970

[Jo] D.S. Johnson, Near-optimal bin packing algorithms, Ph.D. Thesis, Massachusetts Institute of Technology, June 1973, Project MAC TR-109.

[JDUGG] D.S. Johnson, A. Demers, J.D. Ullman, M.R. Garey, and R.L. Graham, "Worst-case performance bounds for simple one-dimensional packing algorithms," SIAM J. of Computing 3, pp. 299-325, 1974.

[Ka] N. Karmarkar, "Probabilistic analysis of some bin-packing algorithms," *Proc. 23rd Annual FOCS*, pp. 107-111, 1982.

[KK] N. Karmarkar and R.M. Karp, "An efficient approximation scheme for the one-dimensional bin packing problem," *Proc. 23rd Annual FOCS*, pp. 312-320, 1982.

[KKLM] N. Karmarkar, R.M. Karp, G.S. Lueker, and F. Murgolo, "Bin packing with symmetric or decreasing distributions," paper in preparation.

[KLM] R.M. Karp, M. Luby, and A. Marchetti-Spaccamela, "Probabilistic analysis of multi-dimensional bin packing problems," *Proc. 16th ACM Symp. on Theory of Computing*, pp. 289-298, 1984.

[LL] F.T. Leighton and C.E. Leiserson, "Wafer-scale integration of systolic arrays," *Proc. 23rd Annual FOCS*, pp. 297-311, 1982.

[Li] F.M. Liang, "A lower bound for on-line bin packing," Inf. Proc. Letters 10, pp. 76-79, 1980.

[Lu] G.S. Lueker, "An average-case analysis of bin packing with uniformly distributed item sizes," Report No. 181, Department of Information and Computer Science, U. of California, Irvine, CA, 1982.

[PS] C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization, Algorithms and Complexity*, Prentice Hall, NJ, 1982.

[Sh] P.W. Shor, "The average-case analysis of some on-line algorithms for bin packing," *Proc. 25th Annual FOCS*, pp. 193-200, 1984.