J. Rissanen

Bounds for Weight Balanced Trees

Abstract: It has been shown that the cost W of a weight balanced binary tree satisfies the inequalities, $H \leq W \leq H + 3$, where H is the entropy of the set of the leaves. For a class of "smooth" distributions the inequalities, $H \leq W \leq H + 2$, are derived.

These results imply that for sets with large entropy the search times provided by such trees cannot be substantially shortened when binary decisions are being used.

1. Introduction

A basic technique for repeatedly searching for the elements of a given ordered finite set is to store the elements in the given order as the leaves in a binary tree. By a suitable marking of the nodes of the tree any given element can be found by making a series of binary comparisons, which, in effect, determine the unique path from the root to the leaf corresponding to the element being searched for. The tree is constructed by taking into account the relative frequencies with which the elements are being searched.

In [1–4] algorithms were given for constructing trees with the shortest mean path length or "cost." In general, such an optimal search tree does not provide an optimal way to retrieve the elements of the given set, optimality being measured by the mean path length. The reason is that the order condition imposed on the leaves is a restriction. However, as shown in [1] the optimal cost is not greater than $H + 2$, where H is the entropy of the set of leaves. This means that as the set to be stored gets bigger and bigger—not in terms of its cardinality, but in terms of its entropy—then the ratio of the optimal mean path length to the ultimate, the entropy, approaches 1.

We shall study the question of how good is the classical weight balanced tree constructed by the rule that each internal node in the tree is chosen so as to equalize the probabilities of the sets of the leaves in the left and the right subtrees of that node. This question in the more general case where even the internal nodes have weights was raised by Knuth in [3]. We shall show that the cost of such a leaf-weighted tree is not greater than $H + 3$, so that for sets with large entropy this cost in relation to the optimal cost is close to one.

This should be of practical interest, since in data storing, in contrast with communication and encoding, the sets of messages considered are large indeed, and since the search probabilities are estimates anyway, an easily constructed asymptotically optimal tree could be fully adequate. The savings in construction of such an asymptotically optimal tree would be particularly important when the search probabilities and the set to be stored undergo a gradual change so that an updating of the tree is needed.

In the last section we sharpen the general bound derived in Section 2 by restricting the class of the allowable distributions.

We should mention that in [5] a different kind of asymptotic optimality of the weight balanced tree was shown in statistical terms. The statistics involve an assumed distribution of the probability distributions $(p_1, \cdots, p_n)$ of the leaves; i.e., the trees themselves are samples of an assumed population. While such statistical estimates are of independent interest they do not provide an answer to the questions considered here.

2. General case

To establish the notations for trees and subtrees, let the ordered set of the leaves be the interval $[1,n]$ of the first n natural numbers, and let the nonnegative weight of i be p_i. There will be no loss in generality in assuming that $\sum_{i=1}^{n} p_i = 1$, so that we may speak of p_i as the probability of i being searched for. Figure 1 shows schematically a tree T_{ij} with the subinterval $[i,j]$ as the ordered set of the leaves.

The root of this tree is denoted by the pair (i,j). The

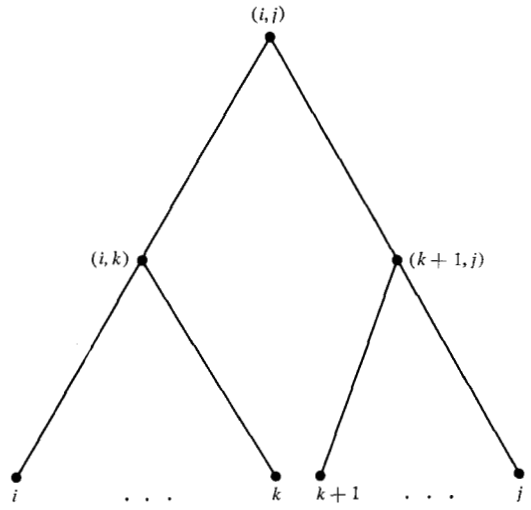

Figure 1 Tree T_{ij} having subinterval $[i,j]$ as its ordered set of leaves.

figure also indicates the roots of the left and the right subtrees, T_{ik} and $T_{k+1,j}$, respectively. The leaf k is called the *index* of the root (i,j).

If ℓ_r denotes the length of the path from the root (i,j) to the leaf r in $[i,j]$, i.e., the number of edges in the unique path from (i,j) to r, then the *weighted* path length of T_{ij} is defined as

$$V_{ij}(p_i, \cdots, p_j) = \sum_{r=i}^{j} p_r \ell_r . \tag{1}$$

The *mean* path length of T_{ij}, or its *cost*, is then defined as

$$W_{ij} = V_{ij}\left(\frac{p_i}{P_{ij}}, \cdots, \frac{p_j}{P_{ij}}\right)$$

$$= \frac{1}{P_{ij}} V_{ij}(p_i, \cdots, p_j), \; P_{ij} = \sum_{r=i}^{j} p_r , \tag{2}$$

where the second equality follows from the linearity of (1). The definition (1) immediately leads to the recursion,

$$V_{ij}(p_i, \cdots, p_j) = P_{ij} + V_{ik}(p_i, \cdots, p_k) + V_{k+1,j}(p_{k+1}, \cdots, p_j)$$

$$\text{for } i \leq k < j, \; V_{ii}(p_i) = 0 . \tag{3}$$

From (2) and (3) we get the formula,

$$W_{ij} = 1 + \frac{P_{ik}}{P_{ij}} W_{ik} + \frac{P_{k+1,i}}{P_{ij}} W_{k+1,j} ,$$

$$i \leq k < j, \; W_{ii} = 0, \tag{4}$$

which is valid for any binary tree with $[i,j]$ as the ordered set of leaves.

Next, let $H_{ij}(p_i, \cdots, p_j) = -\sum_{r=i}^{j} p_r \log p_r$ denote the entropy of the set $[i,j]$ in bits. In analogy to (2) write

$$H_{ij} = H_{ij}\left(\frac{p_i}{P_{ij}}, \cdots, \frac{p_j}{P_{ij}}\right).$$

Then by a direct verification we have the basic identity (in k), [6]:

$$H_{ij} = H\left(\frac{P_{ik}}{P_{ij}}, 1 - \frac{P_{ik}}{P_{ij}}\right) + \frac{P_{ik}}{P_{ij}} H_{ik} + \frac{P_{k+1,j}}{P_{ij}} H_{k+1,j}, \; H_{ii} = 0, \tag{5}$$

where $H(p, 1-p) = -p \log p - (1-p) \log (1-p)$.

The formula (5) looks like (4) except for the first terms in the right hand sides. In the context of a given tree T_{1n} both (4) and (5) express a cost of the subtree T_{ij} in terms of the costs of the left and the right subtrees T_{ik} and $T_{k+1,j}$, respectively. In (4) the first term, 1, is then a constant incremental cost reflecting the fact that the path lengths grow by steps of one unit no matter with what probabilities, P_{ik}/P_{ij} and $P_{k+1,j}/P_{ij}$, the left and the right subtrees are picked. In the case of (5), in contrast, the incremental change in the normalized entropy depends clearly on these two probabilities.

There is also another difference between (4) and (5): W_{ij} depends on k in that if we consider trees T_{ij} with the fixed interval $[i,j]$ as the set of the leaves for various values for k then W_{ij} will not remain constant; in (5), in contrast, H_{ij} is not dependent on k at all.

Even if W_{ij} depends on k it cannot be less than H_{ij}, which immediately follows from (4) and (5) with the fact that

$$H\left(\frac{P_{ik}}{P_{ij}}, 1 - \frac{P_{ik}}{P_{ij}}\right) \leq 1.$$

We shall now give a useful formula for the difference $W_{1n} - H_{1n}$ for any binary tree T_{1n}:

Lemma 1. If T_{1n} with cost W_{1n} is a binary tree with the interval $[1,n]$ as the set of its leaves, then,

$$W_{1n} - H_{1n} = \sum_{(i,j)} P_{ij}\left(1 - H\left(\frac{P_{ik}}{P_{ij}}, 1 - \frac{P_{ik}}{P_{ij}}\right)\right),$$

where (i,j) runs through all the internal nodes of T_{1n}, and k is the index of the root (i,j).

Proof. Let $E_{ij} = W_{ij} - H_{ij}$. By (4) and (5) we get,

$$E_{ij} = e_{ij} + \frac{P_{ik}}{P_{ij}} E_{ik} + \frac{P_{k+1,j}}{P_{ij}} E_{k+1,j}, \; i \leq k < j, E_{ii} = 0, \tag{6}$$

where $e_{ij} = 1 - H\left(\frac{P_{ik}}{P_{ij}}, 1 - \frac{P_{ik}}{P_{ij}}\right)$ for $i < j$ and $e_{ii} = 0$. We shall verify that

$$E_{ij} = \frac{1}{P_{ij}} \sum_{(r,s)} P_{rs} \, e_{rs} \tag{7}$$

satisfies (6). In fact, write this sum as

$$E_{ij} = \frac{1}{P_{ij}} \left(P_{ij}\, e_{ij} + \sum_{(r,s)}^{L} P_{rs}\, e_{rs} + \sum_{(r,s)}^{R} P_{rs}\, e_{rs} \right)$$

Where $\sum^{L}$ and $\sum^{R}$ denote summations over the nodes in the left and the right subtrees T_{ik} and $T_{k+1,j}$ of T_{ij}, respectively. But these summands are just $P_{ik}\, E_{ik}$ and $P_{k+1,j}\, E_{k+1,j}$. As $P_{1n} = 1$ and $e_{rr} = 0$ Eq. (7), then, implies the lemma.

We shall now consider the well-known weight balanced tree T_{1n} defined as follows: For each node (i,j) of the tree the root index $k(i,j)$ is the least integer satisfying the inequality,

$$\left| \frac{P_{i,k(i,j)}}{P_{ij}} - \frac{1}{2} \right| \leq \left| \frac{P_{ir}}{P_{ij}} - \frac{1}{2} \right| \text{ for all } r \varepsilon [i, j-1]\,. \tag{8}$$

As an example of such a tree consider the interval [1,11], where the probabilities are given by (1/62) · (8,6,2,3,4, 7,11,9,8,1,3). By calculating the cumulative sums from left to right, (1/62) · (8,14,16,19,23,30,41,50,58,59,62), we can pick the sequence of the root indices $k(i,j)$ by (8) as follows: $k(1,11) = 6$, $k(1,6) = 2$, $k(7,11) = 8$, $k(1,2) = 1$, $k(3,6) = 5$, $k(3,5) = 4$, $k(3,4) = 3$, $k(7,11) = 8$, $k(7,8) = 7$, $k(9,11) = 9$, $k(10,11) = 10$. These nodes define a tree, which, in fact, is optimal (the example is from [4]). This is more the rule than the exception; it requires a bit of searching to construct a tree where (8) holds that is not optimal.

We shall find an upper bound for the incremental cost e_{ij} for trees satisfying (8). The two mutually exclusive cases arise according to the position of $k(i,j) \triangleq k$:

$$\begin{aligned} &1.\ \frac{P_{ik}}{P_{ij}} \leq \frac{1}{2}, \\ &2.\ \frac{P_{ik}}{P_{ij}} > \frac{1}{2}. \end{aligned} \tag{9}$$

Since $P_{i,k+1} = P_{ik} + p_{k+1}$ we have in the two cases:

$$\begin{aligned} &1.\ \text{a)}\ \frac{1}{2} - \frac{P_{ik}}{P_{ij}} \leq \frac{p_{k+1}}{P_{ij}} \text{ for } k < j-1, \\ &\qquad \text{b)} \qquad \leq \frac{p_{k+1}}{2P_{ij}} \text{ for } k = j-1, \\ &2.\ \text{a)}\ \frac{P_{ik}}{P_{ij}} - \frac{1}{2} < \frac{p_k}{P_{ij}} \quad \text{for } k > i \\ &\qquad \text{b)} \qquad < \frac{p_k}{2P_{ij}} \quad \text{for } k = i. \end{aligned} \tag{10}$$

The function $e(p) = 1 - H(p, 1-p)$ is an even convex function about the point 1/2; it has the value 1 for $p = 0$ and $p = 1$. Hence, the line connecting the points (1/2,0) and (1,1) is not below the graph of $e(p)$ for $1/2 \leq p \leq 1$, and we have the inequality

$$1 - H(p, 1-p) \leq 2|p - 1/2|; \qquad 0 \leq p \leq 1\,, \tag{11}$$

where the equality holds for $p = 0$, 1/2 and 1.

Theorem 1. Let $[1,n]$ have the probabilities $(p_1, \cdots, p_n)$. The cost W_{1n} of the associated weight balanced tree where (8) holds satisfies the inequalities

$$H_{1n} \leq W_{1n} \leq H_{1n} + 3.$$

Proof. By Lemma 1 and (11) we have

$$W_{1n} - H_{1n} \leq 2 \sum_{(i,j)} b(i,j)\,, \tag{12}$$

where $b(i,j)$ denotes p_{k+1} in case (1a) of (10), $p_{k+2}/2$ in case (1b), p_k in case (2a), and p_k in case (2b). We shall have to analyze this a bit further.

In case (1a) of (10) the root index of the subtree T_{ij} is at $k(i,j) = k$, and $b(i,j) = p_{k+1}$. Only a subtree $T_{k+1,r}$ of T_{ij} can have its root index at $k+1$, and hence $b(k+1,r)$ be $p_{k+1}/2$ if case (2b) holds for this subtree. Any other $b(i',j')$ must therefore be either p_s or $p_s/2$ for some $s \neq k+1$. Similarly, in case (2a) of (10) $b(i,j) = p_k$, and only $b(s,k)$ for some s can be $p_k/2$ if case (1b) holds for this subtree T_{sk}. Any other $b(i',j')$ is either p_s or $p_s/2$ for some $s \neq k$.

All told, (12) then gives

$$W_{1n} - H_{1n} \leq 2\left(\sum_{r=1}^{n} p_r + \frac{1}{2} \sum_{r=1}^{n} p_r \right) = 3.$$

With the basic inequality $W_{ij} \geq H_{ij}$ this proves the theorem.

It follows from Theorem 1 that the relative cost W_{1n}/H_{1n} approaches 1 with the rate better than $3/H_{1n}$ as H_{1n} approaches infinity; i.e., as the size of the weighted set $[1,n]$, measured in terms of entropy, approaches infinity. For the tree in our example above $H_{1n} = 3.18$ and $W_{1n} = 3.29$, so that the relative cost 1.03 is only a fraction above one.

3. Special cases

It is not known to us whether the upper bound given in Theorem 1 can be lowered. However, if we impose a smoothness condition that requires the quotient p_i/p_{i+1} to be near one, we can reduce the maximum difference $W_{1n} - H_{1n}$; we shall aim at reducing this difference to 2, which is what one gets with the optimal tree [1]. (We should mention that the upper bound $H_{1n} + 2$ for the best alphabetical code was derived only for the special case that the alphabet in question is the nth extension of an alphabet. The derivation extends easily, however, to the general case). Such a smoothness condition does not, perhaps, restrict too seriously the allowable probability distributions for large values for n, because often the data adjacent to each other are requested roughly with the same frequency.

It is also possible to derive from Lemma 1 a different sort of bound, applicable to all binary trees satisfying a smoothness condition similar to the preceding one. Since

this bound is of some interest and more easily obtained we begin with its derivation.

Let β_{ij} be the smaller of the two numbers P_{ik}/P_{ij} and $1-(P_{ik}/P_{i,j})$, and let β be the minimum of the β_{ij}'s as (i,j) runs through the internal nodes of a tree T_{1n}. Here, k is the index of the root (i,j) of the subtree T_{ij} of T_{1n}. Then

$$1-H\left(\frac{P_{ik}}{P_{ij}},\,1-\frac{P_{ik}}{P_{ij}}\right)\leq 1-H(\beta,1-\beta),$$

and Lemma 1 gives

$$W_{1n}-H_{1n}\leq (1-H(\beta,1-\beta))\sum_{(i,j)}P_{ij}.$$

The sum in the right hand side is W_{1n} by [10, p. 405]; this can also be seen by putting $e_{ij}=1$ in (6), which reduces it to (4). Hence,

$$W_{1n}\leq \frac{1}{H(\beta,1-\beta)}H_{1n}, \tag{13}$$

which is meaningful for $\beta>0$; or, whenever all $p_i>0$. This bound reduces to one given in [8–9] if we put $p_i=1/n$.

Returning to the main task we begin with the inequality,

$$1-H(p,1-p)\leq \frac{4}{3}\left|p-\frac{1}{2}\right| \text{ for } 0.1\leq p\leq 0.9, \tag{14}$$

which can be directly verified. The reason why we put the indicated bounds for p is to reduce the coefficient in front of $|p-1/2|$ to 4/3 (compare with (11)).

Theorem 2. If $1/1.6\leq p_i/p_{i+1}\leq 1.6$ for all i in $[1,n-1]$, then the cost W_{1n} of the weight balanced tree satisfies:

$$H_{1n}\leq W_{1n}\leq H_{1n}+2.$$

Proof. We shall show that the inequality,

$$\left|\frac{P_{ik}}{P_{ij}}-\frac{1}{2}\right|<0.4, \tag{15}$$

holds for all subintervals $[i,j]$ of $[1,n]$. This then ensures that (14) holds for $p=P_{ik}/P_{ij}$, rather than the weaker estimate (11). The theorem is then proved just as Theorem 1 was done by replacing (11) by (14).

To show (15) we shall consider the four mutually exclusive cases according to the value of $m\triangleq j-i+1$: 1) $m=2$, 2) $m=3$, 3) $m=4$, and 4) $m>4$. Let $r=\max_{i\in[1,n-1]}(p_i/p_{i+1},\,p_{i+1}/p_i)$. Consider case 1). We have,

$$\left|\frac{P_{ii}}{P_{i,i+1}}-\frac{1}{2}\right|\leq\left|\frac{1}{2}-\frac{1}{1+r}\right|<0.4.$$

In case 2) let $p_{i+2}\leq p_i$. Then either $p_{i+1}\geq p_i$ or $p_{i+1}<p_i$. In the former case (8) is true for $k=i$, and.

$$\frac{1}{2}\geq\frac{P_{ik}}{P_{ij}}=\frac{1}{1+\dfrac{p_{i+1}+p_{i+2}}{p_i}}\geq\frac{1}{2+r}\geq\frac{1}{3.6}.$$

Therefore, (15) holds. In the latter case we first verify that

$$\frac{p_i}{p_i+p_{i+1}+p_{i+2}}\leq\frac{1}{1+r^{-1}+r^{-2}}<\frac{1}{2}.$$

Next, the left-most expression cannot be smaller than 1/3. Hence, again (15) holds. Finally, if $p_{i+2}>p_i$ then (8) is true for $k=i+1$, and by symmetry of P_{ij} with respect to p_i and p_{i+2} this case reduces to the preceding one and (15) holds true.

In case 3) we have by (8):

$$\left|\frac{P_{ik}}{P_{ij}}-\frac{1}{2}\right|\leq\left|\frac{P_{i,i+1}}{P_{ij}}-\frac{1}{2}\right|.$$

Let first

$$\frac{P_{i,i+1}}{P_{ij}}=\frac{1}{1+\dfrac{p_{i+2}+p_{i+3}}{p_i+p_{i+1}}}\leq\frac{1}{2}.$$

The largest value the quotient $(p_{i+2}+p_{i+3})/(p_i+p_{i+1})$ possibly can have is $(r^2+r^3)/(1+r)=r^2$. Therefore,

$$0.28\leq\frac{1}{1+r^2}\leq\frac{P_{i,i+1}}{P_{ij}}\leq\frac{P_{ik}}{P_{ij}}\leq\frac{1}{2},$$

and (15) holds true. Consider next the case, $P_{i,i+1}/P_{ij}>1/2$. We have

$$\frac{P_{i,i+1}}{P_{ij}}=1-\frac{1}{1+\dfrac{p_i+p_{i+1}}{p_{i+2}+p_{i+3}}}\leq 1-\frac{1}{1+r^2}<0.72,$$

and, again, (15) is satisfied.

In the final case, $m>4$, we use the estimates (10) to obtain:

$$\left|\frac{P_{ik}}{P_{ij}}-\frac{1}{2}\right|\leq\frac{\hat{p}}{P_{ij}}=\left(\sum_{r=i}^{j}\frac{p_r}{\hat{p}}\right)^{-1},\ \hat{p}=\max_{k\in[i,j]}p_k.$$

The sum is not smaller than $1+r^{-1}+r^{-2}+\cdots+r^{-5}$, and, therefore,

$$\left|\frac{P_{ik}}{P_{ij}}-\frac{1}{2}\right|\leq(1+r^{-1}+\cdots+r^{-5})^{-1}<0.4.$$

Hence (15) holds. This is where we got the value 1.6 for r; the term $(1+r^{-1}+\cdots+r^{-5})^{-1}$ is then about 0.4. The proof is complete.

Let us point out to this end that in the extreme case with the uniform distribution, $p_i=1/n$, one can easily show using the formula for the optimal cost [10, p. 400] that the balanced tree is optimal, and that its cost is bounded from above by $H_{1n}+(1+\log_2\log_2 e-\log_2 e)$, where H_{1n} now equals $\log_2 n$. The second term is about 0.08, and it provides the lowest upper bound for the difference $W_{1n}-H_{1n}$ that can be achieved by smoothness conditions of the type studied.

References

1. E. N. Gilbert and E. F. Moore, "Variable-Length Binary Encodings," *Bell System Tech. J.* **38,** 933 (1959).
2. E. Wong, "A Linear Search Problem," *SIAM Review* **6,** 168 (1964).
3. D. E. Knuth, "Optimum Binary Search Trees," *Acta Informatica* **1,** 14 (1971).
4. T. C. Hu, and A. C. Tucker, "Optimal Computer Search Trees and Variable-Length Alphabetical Codes," *SIAM J. Appl. Math.* **21,** 514 (1971).
5. J. Nievergelt, and C. K. Wong, "On Binary Search Trees," *Proc. IFIP Congress* **71,** 91 (1971).
6. C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Tech. J.* **27,** 379 (1948).
7. R. M. Fano, *Transmission of Information*, MIT Press and John Wiley & Sons, Inc., New York, 1961.
8. J. Nievergelt, and C. K. Wong, *Upper Bounds for the Total Path Length of Binary Trees*, IBM T. J. Watson Res. Center Report RC 3075, 1970.
9. J. Nievergelt, and E. M. Reingold, "Binary Search Trees of Bounded Balance," *Proc. of Fourth Annual ACM Symposium on Theory of Computing*, Denver, Colorado, pp. 137–142 (1972).
10. D. E. Knuth, *The Art of Computer Programming*, Vol. 1, Addison-Wesley, New York, 1968.

Received December 5, 1972; revised January 9, 1973

The author is located at the IBM Research Division Laboratory, Monterey and Cottle Roads, San Jose, California 95114.