Multiway trees of maximum and minimum probability under the random permutation model

[short title: Random multiway trees]

BY ROBERT P. DOBROW AND JAMES ALLEN FILL[1]

Northeast Missouri State University
and The Johns Hopkins University

Abstract

Multiway trees, also known as m-ary search trees, are data structures generalizing binary search trees. A common probability model for anlayzing the behavior of these structures is the random permutation model. The probability mass function Q on the set of m-ary search trees under the random permutation model is the distribution induced by sequentially inserting the records of a uniformly random permutation into an initially empty m-ary search tree. We study some basic properties of the functional Q, which serves as a measure of the "shape" of the tree. In particular, we determine exact and asymptotic expressions for the maximum and minimum values of Q and identify and count the trees achieving those values.

[1]Research was carried out while the first author was a postdoctoral research associate at the National Institute of Standards and Technology, Statistical Engineering Division. The first author's institution will change its name to Truman State University in July, 1996. Research for the second author was supported by NSF grant DMS-9311367.

[2]*AMS 1991 subject classifications.* Primary 60C05; secondary 68P10, 68P05.

[3]*Keywords and phrases.* Multiway trees, m-ary search trees, complete tree, random permutation model.

1 Overview

1.1 Introduction and summary

For integer $m \geq 2$, the m-ary search tree, or multiway tree, generalizes the binary search tree. Search trees are fundamental data structures in computer science. For background we refer the reader to Knuth (1973a,b) and Mahmoud (1992).

An *m-ary tree* is a rooted tree with at most m "children" for each node (vertex), each of which is distinguished as one of m possible types. Recursively expressed, an m-ary tree either is empty or is a node (called the root) with an ordered m-tuple of subtrees, each of which is an m-ary tree.

An *m-ary search tree* is an m-ary tree in which each node has the capacity to contain $m-1$ elements of some linearly ordered set, called the set of *keys*. In typical implementations, at each node one has m pointers to the subtrees. By spreading the input data in m directions instead of only 2, as is the case in a binary search tree, one seeks to have shorter path lengths and thus quicker searches. For instance, a file of half a million keys can be stored in a 100-ary search tree of height as small as 3 but requires height at least 19 for binary search.

The m-ary search tree was evidently first introduced by Muntz and Uzgalis (1971) to solve internal memory problems with large quantities of data. In their framework, one thinks of nodes as pages that reside in external memory (m is thus in the hundreds). Within a node, the keys are usually kept in an array or a linked list in sorted order. A special type of multiway tree, called a B-tree, discovered by Bayer and McCreight (1972), has been shown to be an efficient structure for external searching. Mahmoud and Pittel (1989), Devroye (1990), and Pittel (1994) have studied functionals of random m-ary search trees, such as height. Chapter 3 of Mahmoud (1992) is a rich source of results on the properties of m-ary search trees. There is an extensive computer science literature on multiway trees. There is also a large combinatorics literature on m-ary trees. However, as far as we can determine, the combinatorial work has dealt almost exclusively with m-ary trees on n *nodes*, where here we are concerned with m-ary search trees on n *keys*. Fill and Dobrow (1995) treat problems related to computing the number of such trees.

We consider the space of m-ary search trees on n keys and for simplicity take the keys to be $[n] := \{1, 2, \ldots, n\}$. We associate an m-ary search tree

with a sequence of n distinct keys in the following way:

1. If $n < m$, then all the keys are stored in the root node in increasing order.

2. If $n \geq m$, then the first $m-1$ keys in the sequence are stored in the root in increasing order and the remaining $n-(m-1)$ keys are stored in the subtrees subject to the condition that if $\sigma_1 < \sigma_2 < \cdots < \sigma_{m-1}$ denotes the ordered sequence of keys in the root, then the keys in the jth subtree are those that lie between σ_{j-1} and σ_j, where $\sigma_0 := 0$ and $\sigma_m := n+1$.

3. All the subtrees are m-ary search trees.

The two most common probability models on the space of m-ary search trees are the uniform model (every tree equally likely) and the random permutation, or random insertion, model. Under the uniform model the probability of obtaining any particular tree is just the reciprocal of the number of m-ary search trees on n keys. Fill and Dobrow (1995) study problems associated with determining this number. In this paper we treat the random permutation model.

Let π be a random permutation of $[n]$: each of the $n!$ permutations are equally likely. Consider the process of "building" an m-ary tree by inserting the successive elements of π into an initially empty tree. [We refer the reader to Figure 3.1 in Mahmoud (1992) for an illustration of the growth of a ternary (3-ary) tree from a permutation.] The distribution of trees under the random permutation model is the distribution induced by this construction, and we denote its probability mass function by Q.

In this paper we consider some of the most basic properties of Q. We give a closed form expression for $Q(T)$ (Theorem 1) and identify the trees achieving the minimum and maximum values of Q. We derive exact and asymptotic expressions for the minimum (Theorem 2) and maximum values of Q. The results of Section 3 show that the more a tree T (and, recursively, its subtrees) is balanced, the larger is the value of $Q(T)$. So Q is a crude measure of the "shape" of the tree. Our main result (Theorems 3 and 4) is that the complete tree T_n (defined in Section 3) achieves the maximum value of Q (as intuition might suggest) and that $-\ln Q(T_n) = a(m)(n+1) + O((\log n)^2)$, where $a(m)$ is explicitly derived. Here and throughout this paper our main asymptotic results hold as $n \to \infty$ with m held constant. Since

$a(m)$ is itself of rather complicated form, and since m is often quite large in applications, it is of interest to derive asymptotics for $a(m)$ as $m \to \infty$; this is done in Section 4.3. Issues related to the large-m behavior of such parameters are treated throughout the paper, as are questions of monotonicity in m and in n.

We view the present paper as laying the groundwork for a more extensive investigation of the distribution of $Q(T)$, where T is a random m-ary search tree with distribution either uniform or Q. Much of our present study was initiated by Fill (1995) for the case of *binary* search trees ($m = 2$). We have found simpler proofs for several results in that paper. Note that the analysis of multiway trees for general $m \geq 2$ is considerably more difficult than in the binary case, primarily because of the distinction between node and key for $m \geq 3$.

1.2 A formula for Q

We first establish some notation. Let T be an m-ary search tree and $|T|$ denote the number of keys in T. Call a node *full* if it contains $m - 1$ keys. For $1 \leq j \leq m$, let $L_j(T)$ denote the jth subtree of T. When it is clear to which tree we are referring we will call the subtrees simply $L_1, \ldots, L_m$. For x a node in T, write $T(x)$ for the subtree of T induced by making x the root.

The distribution Q, of course, depends on n but we will suppress that dependence in the notation. We use the standard convention that an empty product equals unity.

Theorem 1 *Let T be an m-ary search tree. Then*

$$Q(T) = \frac{1}{\prod_x \binom{|T(x)|}{m-1}}, \tag{1}$$

where the product is over all full nodes in T.

Proof If $0 \leq |T| < m - 1$, then clearly $Q(T) = 1$. Suppose $|T| \geq m - 1$. Consider the process of selecting n keys without replacement. The probability of first selecting a particular set of $m - 1$ keys for the root of T is $1 \Big/ \binom{|T|}{m-1}$. From the recursive definition of m-ary search trees it follows that

$$Q(T) = \frac{1}{\binom{|T|}{m-1}} Q(L_1) \cdots Q(L_m),$$

and the result follows by iteration. ■

Since there is a unique way to label any unlabeled m-ary search tree on n keys with the keys in $[n]$, we need not be concerned any futher with labels. In particular, if T is an (unlabeled) m-ary search tree on n keys and T' is obtained by permuting the order of the subtrees $L_1, \ldots L_m$, then T' is also an (unlabeled) m-ary search tree on n keys.

A node in an unlabeled m-ary search tree can be drawn as a rectangular box divided into $m-1$ square-box components. Shading of the first j components ($1 \leq j \leq m-1$) indicates that the node is filled to partial capacity j by the inclusion of exactly j keys. For all of the figures drawn in this paper, it happens that all nodes displayed are full.

2 The minimizers of Q

Let M_n denote the set of (unlabeled) m-ary search trees on n keys. For tree T, define $R(T) := 1/Q(T) = \prod_x \binom{|T(x)|}{m-1}$. The functional R takes values in $\{1, 2, \ldots\}$. Note that R satisfies the recursion

$$R(T) = \binom{|T|}{m-1} R(L_1(T)) \cdots R(L_m(T)). \tag{2}$$

From (2) we see that when drawing trees in the plane, the left-to-right position of each subtree plays no role in consideration of $R(\cdot)$. By this symmetry we may, when convenient, restrict attention from M_n to

$$\tilde{M}_n := \{T \in M_n : |L_1(T(x))| \geq |L_2(T(x))| \geq \cdots \geq |L_m(T(x))| \text{ for all } x \in T\}.$$

We may further restrict attention to trees which satisfy a "left-shifted leaves" property. Given $T \in \tilde{M}_n$, construct $\bar{T}$ by shifting the keys of T leftward in two stages as follows. At the first stage, if $|T| \leq m-1$ or if each of the m subtrees $L_j(T)$ either has a full root or is empty, do nothing. Otherwise, let

$$j := \min\{1 \leq k \leq m : 1 \leq |L_k(T)| < m-1\}$$

be the index of the leftmost subtree which is a non-empty but non-full node. Observe that since $T \in \tilde{M}_n$, the subtrees $L_{j+1}(T), \ldots, L_m(T)$ all have non-full (and possibly empty) roots. Let $z := \sum_{i=0}^{m-j} |L_{j+i}(T)|$. Construct a new

tree T' by shifting the keys in $L_{j+1}(T), \ldots, L_m(T)$ as far leftward as possible. That is, construct T' by successively declaring

$$|L_{j+k}(T')| = \min\left(m-1, z - \sum_{i=0}^{k-1} |L_{j+i}(T)|\right),$$

for $k = 0, 1, \ldots, m-j$. For the second stage, recursively apply the two-stage process to each of the m subtrees $L_i(T')$. Call the final result $\bar{T}$.

Since non-full nodes are childless it follows that $R(T) = R(\bar{T})$ and hence that we can further restrict attention to trees in $\tilde{M}_n$ fixed by the leaf-shifting transformation we have described. We denote the set of trees enjoying this "left-shifted leaves" property by $\bar{M}_n$.

In this notation it follows that the unique minimizer of Q in $\bar{M}_n$ is the tree built (as described in Section 1.1) from the reversal permutation $(n, n-1, \ldots, 1)$. Let $T_{\min}$ be any minimizer of Q on n keys. We collect results for $T_{\min}$ in Theorem 2. We first state a lemma which will be used in the proof of that theorem.

Lemma 2.1 *For $n \geq 0$ and $m \geq 2$, let $Q_m(n) := Q(T_{\min})$ and $\rho_m(n) := Q_{m+1}(n)/Q_m(n)$. Then*

$$\rho_m(n) \geq m/(m-1) \quad \textit{for } 2 \leq m \leq n.$$

Proof We establish the result by induction on n for each fixed $m \geq 2$. The following notation will be convenient for general $n \geq 1$:

$$\begin{aligned} n &= k(m-1) + r, \quad \text{with } k \geq 0 \text{ and } 1 \leq r \leq m-1 \\ &= \ell m + s, \qquad\quad \text{with } \ell \geq 0 \text{ and } 0 \leq s \leq m-1. \end{aligned}$$

Then

$$Q_m(n) = \frac{((m-1)!)^k r!}{n!}; \quad Q_{m+1}(n) = \frac{(m!)^\ell s!}{n!}; \quad \rho_m(n) = \frac{(m!)^\ell s!}{((m-1)!)^k r!}.$$

In this notation we will prove the result by induction on $k \geq 1$ for each fixed r satisfying $1 \leq r \leq m-1$.

For the basis of the induction, suppose that $k = 1$ and $1 \leq r \leq m-1$. Then $\ell = 1$ and $s = r-1$ and

$$\rho_m(n) = \frac{m!(r-1)!}{(m-1)!r!} = \frac{m}{r} \geq \frac{m}{m-1}.$$

For the induction step, let $k' = k+1$ and correspondingly

$$r' = r, \quad n' = n + (m-1) = k'(m-1) + r' = \ell' m + s'.$$

We consider two possibilities in turn. In each case we establish $\rho_m(n') \geq \rho_m(n)$, thereby completing the proof:

Case 1: If $s = 0$, then $\ell' = \ell$ and $s' = m-1$ and

$$\rho_m(n') = \frac{(m!)^{\ell'}(s')!}{((m-1)!)^{k'}(r')!} = \frac{(m!)^{\ell}(m-1)!}{((m-1)!)^{k+1}r!} = \rho_m(n).$$

Case 2: If $s \geq 1$, then $\ell' = \ell + 1$ and $s' = s - 1$ and

$$\begin{aligned} \rho_m(n') &= \frac{(m!)^{\ell'}(s')!}{((m-1)!)^{k'}(r')!} = \frac{(m!)^{\ell+1}(s-1)!}{((m-1)!)^{k+1}r!} \\ &= \frac{m}{s}\frac{(m!)^{\ell}s!}{((m-1)!)^{k}r!} = \frac{m}{s}\rho_m(n) \\ &\geq \frac{m}{m-1}\rho_m(n) > \rho_m(n). \end{aligned}$$

■

Theorem 2 *For $n \geq 1$, write $n = k(m-1)+r$ with $k \geq 0$ and $1 \leq r \leq m-1$.*

(a) *For general $m \geq 2$ and $n \geq 1$, the minimum value $Q_m(n)$ of Q over m-ary search trees on n keys is given by*

$$Q_m(n) = \frac{((m-1)!)^k r!}{n!}.$$

If $m-1$ divides n, then

$$Q_m(n) \sim \frac{1}{\sqrt{2\pi n}}\left(\frac{c_m}{n}\right)^n \quad \text{as } n \to \infty,$$

where $c_m = e((m-1)!)^{1/(m-1)} \sim m$ as $m \to \infty$. The constant c_m is strictly increasing in $m \geq 2$.

(b) *For $n \geq m$ there are $m^{k-1}\binom{m-1+r}{r}$ members of M_n that minimize Q.*

(c) *For each $n \geq 0$, $Q_m(n)$ is strictly increasing in m for $2 \leq m \leq n+1$.*

(d) *For each $m \geq 2$, $Q_m(n)$ is strictly decreasing in $n \geq m+1$.*

Proof Part (a) follows directly from Theorem 1 and Stirling's approximation. A standard "stars-and-bars" combinatorics argument gives (b). Part (c) follows immediately from Lemma 2.1, and there is a similar (and simpler) proof of part (d).

■

Remark: The total mass assigned by Q to trees with $Q(T) = Q(T_{\min})$ equals $m^{k-1}\binom{m-1+r}{r}Q_m(n)$. If for simplicity we assume that $m-1$ divides $n \geq m$, then

$$\begin{aligned} m^{k-1}\binom{m-1+r}{r}Q_m(n) &= \frac{1}{n!}\binom{2(m-1)}{m-1}m^{\frac{n}{m-1}-2}\left((m-1)!\right)^{n/(m-1)} \\ &\sim \frac{1}{m^2}\binom{2(m-1)}{m-1}\frac{1}{\sqrt{2\pi n}}\left(\frac{c_m m^{1/(m-1)}}{n}\right)^n \end{aligned}$$

as $n \to \infty$. The constant $c_m m^{1/(m-1)}$ is strictly increasing in $m \geq 2$ and $\sim m$ as $m \to \infty$.

3 The complete tree maximizes Q

At the opposite extreme from the long, stringy trees minimizing Q is the complete tree, which can be defined as follows. Suppose first that $n = m^k - 1$ for integer k. We then say that n is (m-)perfect, and we call the unique tree in M_n with minimum possible height ($= k-1$) the *perfect tree*. For general n, let $k = \lfloor \log_m(n+1) \rfloor$. The *complete tree* can be obtained by attaching to the perfect tree on $m^k - 1$ keys, and as far to the left as possible, $n - (m^k - 1)$ leaves at distance k from the root. In particular, if $n = m^k - 1$, the notions of perfect tree and complete tree coincide. Note that the complete tree, as we have defined it, has the "left-shifted leaves" property.

Define

$$R_n^* := \min_{T \in M_n} R(T)$$

and

$$M_n^* := \{T \in M_n : R(T) = R_n^*\}, \quad \bar{M}_n^* := \bar{M}_n \cap M_n^*.$$

Write T_n for the complete tree on n nodes and set $R_n := R(T_n)$.

We are now prepared to state our main result: the complete tree T_n is the essentially unique maximizer of Q. We take up computation of the value $Q(T_n)$ in Section 4 and of $|M_n^*|$ in Section 5.

Theorem 3 *For $n \geq 0$,*

$$\bar{M}_n^* = \{T_n\}.$$

The next two lemmas describe simple operations that reduce the value of R.

Lemma 3.1 *For $k = 1, \ldots, m-1$, consider trees T and T' as in Figures 1 and 2, respectively. Then*

$$R(T') \left\{ \begin{array}{c} < \\ = \\ > \end{array} \right\} R(T) \quad \textit{according as} \quad \left|T^{m+k}\right| \left\{ \begin{array}{c} < \\ = \\ > \end{array} \right\} \left|T^1\right|.$$

Proof Write t_i for $|T^i|$ and r_i for $R(T^i)$ for $i = 1, \ldots, 2m-1$. Then

$$\begin{aligned} R(T) &= \binom{n}{m-1} \binom{(m-1) + \sum_{i=1}^{m} t_i}{m-1} r_1 \cdots r_{2m-1}, \\ R(T') &= \binom{n}{m-1} \binom{(m-1) + (\sum_{i=2}^{m} t_i) + t_{m+k}}{m-1} r_1 \cdots r_{2m-1}, \end{aligned}$$

and the result follows. ■

Lemma 3.2 *For $1 \leq s < t \leq m$, consider an m-ary tree $T \in \bar{M}_n$ ($n \geq (t+1)(m-1)$) as in Figure 3, and the modification $T'' \in M_n$, shown in Figure 4, obtained by swapping the subtrees T^m and T^{m+1}. Then*

$$R(T'') \left\{ \begin{array}{c} < \\ = \\ > \end{array} \right\} R(T) \quad \textit{according as} \quad |T^m| \left\{ \begin{array}{c} < \\ = \\ > \end{array} \right\} \left|T^{m+1}\right|.$$

Proof For the tree T in Figure 3, write t_i for $|T^i|$ and r_i for $R(T^i)$ for $i = 1, \ldots, 2m$. Let

$$\rho := \binom{n}{m-1} r_1 \cdots r_{2m} \prod_{1 \leq i \leq m : i \neq s,t} R(L_i).$$

Then

$$\begin{aligned} R(T) &= \rho \binom{(m-1) + \sum_{i=1}^{m} t_i}{m-1} \binom{(m-1) + \sum_{i=m+1}^{2m} t_i}{m-1}, \\ R(T'') &= \rho \binom{(m-1) + (\sum_{i=1}^{m-1} t_i) + t_{m+1}}{m-1} \binom{(m-1) + t_m + (\sum_{i=m+2}^{2m} t_i)}{m-1}. \end{aligned}$$

After a little calculation, using the log concavity of the binomial coefficient $\binom{n}{k}$ in $n \geq k$ for fixed $k \geq 0$, one finds that

$$R(T'') \left\{ \begin{array}{c} < \\ = \\ > \end{array} \right\} R(T)$$

according as

$$(t_m - t_{m+1})((t_1 + \cdots + t_{m-1}) - (t_{m+2} + \cdots + t_{2m})) \left\{ \begin{array}{c} < \\ = \\ > \end{array} \right\} 0.$$

If $t_m = t_{m+1}$, then $R(T'') = R(T)$. Otherwise we claim

$$t_1 + \cdots + t_{m-1} > t_{m+2} + \cdots + t_{2m}, \tag{3}$$

from which the other two cases follow. To see (3), observe

$$m \sum_{i=1}^{m-1} t_i \geq (m-1) \sum_{i=1}^{m} t_i \geq (m-1) \sum_{i=m+1}^{2m} t_i \geq m \sum_{i=m+2}^{2m} t_i, \tag{4}$$

where each of the three inequalities follows from the assumption $T \in \bar{M}_n$. If equality holds in (3), then equality must hold throughout (4); but then $t_1 = \cdots = t_{2m}$, contradicting our assumption that $t_m \neq t_{m+1}$. ■

We are now ready for the proof that $\bar{M}_n^* = \{T_n\}$.

Proof *of Theorem 3.* The proof is by (strong) induction on n. The assertion is trivially correct for $n \leq 2(m-1)$. Suppose $2(m-1) < n \leq m^2 - 1$. Then

$$R(T_n) = \binom{n}{m-1},$$

and clearly for any other $T \in \bar{M}_n$ we have $R(T) > R(T_n)$. Given $n > m^2 - 1$, we suppose that $\bar{M}_\ell^* = \{T_\ell\}$ for $0 \leq \ell \leq n-1$ and prove that $\bar{M}_n^* = \{T_n\}$. To do this, we will show that if $T \in \bar{M}_n$ is *not* complete, then there exists $\tilde{T} \in M_n$ with $R(\tilde{T}) < R(T)$.

Note that *any* $T \in \bar{M}_n$ is of the form of T as in Lemma 3.1. If T^{m+1} is empty, then (since $n > m^2 - 1$ and $T \in \bar{M}_n$) T^1 is nonempty and T' of Lemma 3.1 (with $k = 1$) provides the required $\tilde{T}$. By repeated such applications of

Lemma 3.1 we may assume that each of the subtrees $L_1(T), \ldots, L_m(T)$ has a full root, i.e., that T is as shown in Figure 5.

By the induction hypothesis, if any of the $m^2 + m$ subtrees $L_1(T)$, ..., $L_m(T)$, T^1, $T^2, \ldots, T^{m^2}$ is incomplete, replacement of the subtree by the complete tree of the same size will (strictly) reduce the product R. So we may assume that each of these subtrees is complete.

Put $t_i := |T^i|$ for $i = 1, \ldots, m^2$. Since $T \in \bar{M}_n$, $t_{km+1} \geq t_{km+2} \geq \cdots \geq t_{km+m}$ for $k = 0, 1, \ldots m-1$. Using Lemma 3.2 we may assume $t_{km} \geq t_{km+1}$ for $k = 1, \ldots, m-1$; thus t_i is nonincreasing in i. On the other hand, we may also assume

$$t_1 \leq t_{km+1} + \cdots + t_{km+m} + (m-1), \tag{5}$$

for $k = 1, \ldots, m-1$, for otherwise Lemma 3.1 can be used to reduce R.

Write $h_i := \lfloor \log_m(t_i + 1) \rfloor$ for $i = 1, \ldots, m^2$. Then for $k = 1, \ldots, m-1$,

$$\begin{aligned} m^{h_1} - 1 \leq t_1 &\leq t_{km+1} + \cdots + t_{km+m} + (m-1) \\ &\leq m t_{km+1} + (m-1) \\ &< m(m^{h_{km+1}+1} - 1) + (m-1) = m^{h_{km+1}+2} - 1, \end{aligned}$$

so $h_1 \leq h_{km+1} + 1$. By the completeness of L_{k+1}, $h_{km+1} \leq h_{km+m} + 1$. Choosing $k = m-1$ we find $h_1 \leq h_{m^2} + 2$. We claim that $h_1 \leq h_{m^2} + 1$. The claim is clearly true if $h_1 = h_{(m-1)m+1}$ or $h_{(m-1)m+1} = h_{m^2}$. Suppose instead that $h_1 = h_{(m-1)m+1} + 1$ and $h_{(m-1)m+1} = h_{m^2} + 1$. From the completeness of L_m and the second of these equalities it follows that $T^{(m-1)m+1}$ is perfect. Hence

$$(m-1) + \sum_{i=(m-1)m+1}^{m^2} t_i < (m-1) + m(m^{h_{(m-1)m+1}} - 1) = m^{h_1} - 1 \leq t_1,$$

contradicting (5). Therefore $h_1 \leq h_{m^2} + 1$ and we may assume, since T is not complete, that $h_1 = h_{m^2} + 1$. (This is easy to see from our assumptions about subtree completeness and the fact that $t_1 \geq t_2 \geq \cdots \geq t_{m^2}$.)

So we may assume that $h_1 = h_{m^2} + 1$, i.e., that exactly one of the $m^2 - 1$ differences $h_1 - h_2, h_2 - h_3, \ldots, h_{m^2-1} - h_{m^2}$ equals 1, and the others vanish. We consider each of the possible cases.

Case 1: $h_1 = h_2 + 1$ (and $m \geq 3$; otherwise see Case 2 below). We claim that this condition contradicts our assumption that T is not complete. The

condition and the fact that L_1 is complete implies that $T^1, T^3, \ldots, T^m$ are all perfect. But then $T^{m+1}, \ldots, T^{m^2}$ are all perfect also, and T is complete.

Observe that the cases $h_i = h_{i+1} + 1$, $i = 2, \ldots, m-2$, can be treated as in Case 1.

Case 2: $h_{m-1} = h_m + 1$. In this case, all the T^is are perfect except precisely for T^m and T^{m+1}. Letting $a := h_m$, this gives $t_1 = \cdots = t_{m-1} = m^{a+1} - 1$, $t_{m+2} = \cdots = t_{2m} = m^a - 1$, and $|L_3| = \cdots = |L_m| = m^{a+1} - 1$.

(2a) If $t_m + t_{m+1} < (m+1)m^a - 1$, construct $\tilde{T}$ from T by replacing T^m with $T_{t_m + t_{m+1} - (m^a - 1)}$ and T^{m+1} with $T_{m^a - 1}$; that is, let $\tilde{T} = T_n$. Then

$$\begin{aligned} R(\tilde{T}) &= \binom{n}{m-1} \binom{(m-1) + t_1 + \cdots + t_{m+1} - (m^a - 1)}{m-1} (R_{m^{a+1}-1})^{m-1} \\ &\quad \times R_{t_m + t_{m+1} - (m^a - 1)} \binom{(m-1) + m(m^a - 1)}{m-1} (R_{m^a - 1})^m \\ &\quad \times R(L_3) \cdots R(L_m). \end{aligned}$$

Since neither T^m nor T^{m+1} is perfect,

$$m^a - 1 \leq t_m + t_{m+1} - (m^a - 1) < m^{a+1} - 1,$$

and so

$$\begin{aligned} & R_{t_m + t_{m+1} - (m^a - 1)} (R_{m^a - 1})^{m-1} \\ &= \binom{t_m + t_{m+1} + (m-2)(m^a - 1) + (m-1)}{m-1}^{-1} \\ &\qquad \times R_{t_m + t_{m+1} + (m-2)(m^a - 1) + (m-1)} \\ &\leq R_{t_m} R_{t_{m+1}} \cdots R_{t_{2m-1}}, \end{aligned}$$

with the inequality holding by induction. Further,

$$(m-1) + t_1 + \cdots + t_m > (m-1) + t_{m+1} + \cdots + t_{2m}$$

and $t_{m+1} > m^a - 1$, so

$$\begin{aligned} & \binom{(m-1) + \sum_{i=1}^{m+1} t_i - (m^a - 1)}{m-1} \binom{(m-1) + m(m^a - 1)}{m-1} \\ &\quad < \binom{(m-1) + \sum_{i=1}^{m} t_i}{m-1} \binom{(m-1) + \sum_{i=m+1}^{2m} t_i}{m-1}, \end{aligned}$$

where we have again used the log concavity of the binomial coefficient. Thus

$$\begin{aligned} R(\tilde{T}) &< \binom{n}{m-1}\binom{(m-1)+\sum_{i=1}^{m} t_i}{m-1}\binom{(m-1)+\sum_{i=m+1}^{2m} t_i}{m-1} \\ &\quad \times R_{t_1}\cdots R_{t_{2m}} R(L_3)\cdots R(L_m) \\ &= R(T), \end{aligned}$$

as desired.

(2b) If $t_m + t_{m+1} \geq (m+1)m^a - 1$, construct $\tilde{T}$ from T by replacing T^m with $T_{m^{a+1}-1}$ and T^{m+1} with $T_{t_m+t_{m+1}-(m^{a+1}-1)}$; that is, again let $\tilde{T} = T_n$. By calculations similar to those for Case 2a, $R(\tilde{T}) < R(T)$, as desired; we leave the details to the reader.

Observe that the cases $h_{km-1} = h_{km} + 1$, $k = 1, \ldots, m-1$, can be treated essentially as in Case 2.

Case 3: $h_m = h_{m+1} + 1$ (and $m \geq 3$; otherwise see Case 4 below). We claim that this condition contradicts our assumption that T is not complete. Letting $a := h_{m+1}$, the condition implies that all the T^is are perfect except for possibly T^1 and T^{m+1}, with

$$t_2 = \cdots = t_m = m^{a+1} - 1 \quad \text{and} \quad t_{m+2} = \cdots = t^{m^2} = m^a - 1.$$

But by (5),

$$m^{a+1} - 1 = t_2 \leq t_1 \leq (m-1) + \sum_{i=(m-1)m+1}^{m^2} t_i = m^{a+1} - 1.$$

Thus T^1 is perfect and hence T is complete.

Observe that the cases $h_{km+i} = h_{km+i+1} + 1$, $k = 1, \ldots, m-2$ and $i = 0, \ldots, m-2$, can be handled essentially as in Case 3.

Case 4: $h_{(m-1)m+i} = h_{(m-1)m+i+1} + 1$ for some $i = 0, \ldots, m-1$. Then $T^2, \ldots, T^m$ are perfect, T_1 and L_m are complete but not perfect, and $L_2, \ldots, L_{m-1}$ are perfect, with $m^a - 1 < t_1 < m^{a+1} - 1$, $m^a - 1 < |L_m| < m^{a+1} - 1$,

$$t_2 = \cdots = t_m = m^a - 1, \quad \text{and} \quad |L_2| = \cdots = |L_{m-1}| = m^{a+1} - 1,$$

for some $a \geq 1$. We show that the choice $\tilde{T} = T_n$ again works, i.e., that $R_n < R(T)$. As for Case 2, we divide the analysis into two subcases.

(4a) If $t_1 + |L_m| < (m+1)m^a - 1$, then

$$m^a - 1 < n - (m-1)m^{a+1} = t_1 + |L_m| - (m^a - 1) \le m^{a+1} - 1,$$

so

$$\begin{aligned}
R_n &= \binom{n}{m-1} \left(R_{m^{a+1}-1}\right)^{m-1} R_{n-(m-1)m^{a+1}} \\
&= \binom{n}{m-1} \left[\binom{m^{a+1}-1}{m-1} \left(R_{m^a-1}\right)^m\right]^{m-1} R_{n-(m-1)m^{a+1}} \\
&= \binom{n}{m-1}\binom{m^{a+1}-1}{m-1}^{m-1} \left(R_{m^a-1}\right)^{(m-1)^2} R_{n-(m-1)m^{a+1}} \left(R_{m^a-1}\right)^{m-1} \\
&= \binom{n}{m-1}\binom{m^{a+1}-1}{m-1}^{m-1} \left(R_{m^a-1}\right)^{(m-1)^2} \\
&\qquad \times \binom{n-(m-1)^2m^a}{m-1}^{-1} R_{n-(m-1)^2m^a} \\
&< \binom{n}{m-1}\binom{m^{a+1}-1}{m-1}^{m-1} \left(R_{m^a-1}\right)^{(m-1)^2} \left(R_{m^a-1}\right)^{m-2} R(T^1)R(L_m) \\
&= \binom{n}{m-1} \left[\binom{m^{a+1}-1}{m-1} R(T^1) \left(R_{m^a-1}\right)^{m-1}\right] \times \\
&\qquad \times \left[\binom{m^{a+1}-1}{m-1} \left(R_{m^a-1}\right)^m\right]^{m-2} R(L_m) \\
&= \binom{n}{m-1} \left[\binom{m^{a+1}-1}{m-1} R(T^1) \left(R_{m^a-1}\right)^{m-1}\right] R(L_2)\cdots R(L_m) \\
&< \binom{n}{m-1} \left[\binom{t_1+(m-1)m^a}{m-1} R(T^1) \left(R_{m^a-1}\right)^{m-1}\right] R(L_2)\cdots R(L_m) \\
&= \binom{n}{m-1} R(L_1)\cdots R(L_m) = R(T),
\end{aligned}$$

where the first inequality holds by induction, since

$$(m-1) + (m-2)(m^a-1) + t_1 + |L_m| = n - (m-1)^2m^a$$

and neither T^1 nor L_m is perfect.

(4b) If $t_1 + |L_m| \ge (m+1)m^a - 1$, then similar calculations show again that $R_n < R(T)$; we leave the details to the reader.

This exhausts the possible cases and the proof is complete.

■

Remark: It is easy to check that virtually the same proof extends Theorem 3 to show that the complete tree T_n is the unique minimizer in $\bar{M}_n$ of any functional g of the form

$$g(T) = \sum_x f(|T(x)|),$$

where the sum is over full nodes $x \in T$, with f strictly increasing and strictly concave (over $\{m-1, m, \ldots\}$). The theorem is the special case $f(y) \equiv \log \binom{y}{m-1}$.

If f is assumed only to be nondecreasing and concave, then we still have the result that $T_n \in \bar{M}_n^*$. An example is $f(x) \equiv x$. Here uniqueness fails: It is easy to check that $T \in M_n$ minimizes $\sum |T(x)|$ (whether or not the sum is extended to *all* nodes in T) if and only if, with $h := \lfloor \log_m(n+1) \rfloor$, (i) T is "perfect through depth $h-1$," i.e., T has m^k full nodes at depth k for $k = 0, 1, \ldots, h-1$, and (ii) T has height at most h.

4 Analysis of maximum value of Q

In this section we investigate the asymptotic behavior of the mode of Q, or, equivalently, of the minimum value $R_n^* = R_n = R(T_n)$ of $R(T) \equiv 1/Q(T)$, achieved when T is the complete tree T_n.

4.1 An exact expression for perfect trees and a lower bound in general

Analysis of R_n is most straightforward when $n = m^k - 1$, so we begin with this perfect tree case. For integer $\nu \geq 1$, define

$$s_m(\nu) := \sum_{j=1}^{\infty} m^{-j} \ln \binom{m^j \nu - 1}{m-1} \geq 0. \tag{6}$$

For $n \geq 0$, set

$$\hat{R}_n := \exp[s_m(1)(n+1) - s_m(n+1)]. \tag{7}$$

We then have the following exact solution:

Proposition 4.1 *If* $0 \le n = m^k - 1$, *then* $R_n = \hat{R}_n$.

Proof Writing r_k for R_{m^k-1}, (2) gives the recurrence relation

$$r_0 = 1; \quad r_k = \binom{m^k - 1}{m-1} r_{k-1}^m, \quad k \ge 1,$$

whose solution is

$$r_k = \prod_{j=1}^{k} \binom{m^j - 1}{m-1}^{m^{k-j}}, \quad k \ge 0.$$

Therefore

$$\begin{aligned}
\ln R_n &= \sum_{j=1}^{k} m^{k-j} \ln \binom{m^j - 1}{m-1} = (n+1) \sum_{j=1}^{k} m^{-j} \ln \binom{m^j - 1}{m-1} \\
&= (n+1) s_m(1) - m^k \sum_{j=k+1}^{\infty} m^{-j} \ln \binom{m^j - 1}{m-1} \\
&= (n+1) s_m(1) - \sum_{j=1}^{\infty} m^{-j} \ln \binom{m^j m^k - 1}{m-1} \\
&= (n+1) s_m(1) - s_m(n+1).
\end{aligned}$$

■

We next show that, for every $n \ge 0$, $\hat{R}_n$ provides a lower bound on R_n.

Lemma 4.1 *For every* $n \ge 0$, $R_n \ge \hat{R}_n$.

Proof The proof is by (strong) induction on n. For $0 \le n \le m-2$ we have equality: $R_n = 1 = \hat{R}_n$. The key to the induction step is the recurrence

$$R_n = \binom{n}{m-1} R_{\nu_1} \cdots R_{\nu_m}, \quad n \ge m-1,$$

for some $\nu_i \ge 0$ with $\sum_{i=1}^{m} \nu_i = n - (m-1)$. Then, by induction,

$$\begin{aligned}
\ln R_n &\ge \ln \binom{n}{m-1} + \sum_{j=1}^{m} \ln \hat{R}_{\nu_j} \\
&= \ln \binom{n}{m-1} + \sum_{j=1}^{m} [s_m(1)(\nu_j + 1) - s_m(\nu_j + 1)]
\end{aligned}$$

$$
\begin{aligned}
&= \ln \binom{n}{m-1} + s_m(1)(n+1) - \sum_{j=1}^{m} s_m(\nu_j + 1)] \\
&\geq \ln \binom{n}{m-1} + s_m(1)(n+1) - m s_m\left(\frac{n+1}{m}\right),
\end{aligned}
$$

where the last inequality follows by the concavity of s_m, which in turn follows from the log concavity of binomial coefficients. But

$$
\begin{aligned}
m s_m\left(\frac{n+1}{m}\right) &= m \sum_{j=1}^{\infty} m^{-j} \ln \binom{m^{j-1}(n+1) - 1}{m-1} \\
&= \sum_{j=0}^{\infty} m^{-j} \ln \binom{m^{j}(n+1) - 1}{m-1} \\
&= \ln \binom{n}{m-1} + s_m(n+1).
\end{aligned}
$$

So

$$
\ln R_n \geq s_m(1)(n+1) - s_m(n+1) = \ln \hat{R}_n,
$$

as desired. ■

4.2 Asymptotics

It is simple to derive an asymptotic expansion (as $\nu \to \infty$, with m fixed) for $s_m(\nu)$ at (6). We content ourselves with the following:

$$
s_m(\nu) = \ln \nu - (m-1)^{-1} \ln \left(\frac{m^{m+1}}{m!}\right) + O(\nu^{-1}).
$$

We therefore have the following result.

Lemma 4.2

$$
\hat{R}_n = (1 + O(n^{-1}))\hat{\hat{R}}_n \text{ as } n \to \infty,
$$

where

$$
\hat{\hat{R}}_n := C_m (n+1)^{-1} \exp[s_m(1)(n+1)], \quad n \geq 0,
$$

with

$$
C_m := \left(\frac{m!}{m^{m+1}}\right)^{1/(m-1)}.
$$

Remark: The factor C_m is strictly decreasing in m. It equals $1/4$ at $m = 2$ and approaches $1/e$ as $m \to \infty$.

According to Proposition 4.1, the preceding lemma gives precise asymptotics for R_n when n is perfect, i.e., when n is of the form $n = m^k - 1$. Pinning down the asymptotics for general n is not so easy. Here we will be content to establish the following result by finding a suitable upper bound on R_n to serve as a companion to Lemmas 4.1 and 4.2.

Theorem 4 *Write $Q_n = 1/R_n$ for the maximum value of Q. Then, as $n \to \infty$ (with $m \geq 2$ fixed),*

$$\ln R_n = s_m(1)(n+1) + O\left((\log n)^2\right),$$

where $s_m(1)$ is defined at (6).

Proof We first derive an exact expression for R_n. Consider $m \geq 2$ and $n \geq m - 1$. Let

$$k := \lfloor \log_m(n+1) \rfloor, \quad \alpha := \left\lfloor \frac{(n+1) - m^k}{(m-1)m^{k-1}} \right\rfloor,$$

and

$$r := (n+1) - m^k - \alpha(m-1)m^{k-1}.$$

Then

$$n + 1 = m^k + \alpha(m-1)m^{k-1} + r,$$

and $k \geq 1$, $0 \leq \alpha < m$, and $0 \leq r < (m-1)m^{k-1}$. Moreover,

$$n - (m-1) = \alpha(m^k - 1) + \rho + \beta(m^{k-1} - 1),$$

where $\beta := (m-1) - \alpha$ satisfies $0 \leq \beta < m$ and $\rho := m^{k-1} - 1 + r$ satisfies $m^{k-1} - 1 \leq \rho < m^k - 1$. Recall the notation r_k for R_{m^k-1}. Then

$$R_n = \binom{n}{m-1} r_k^{\alpha} r_{k-1}^{m-1-\alpha} R_{\rho}. \tag{8}$$

The foregoing can be easily iterated, as follows. Consider $m \geq 2$ and $n \geq 0$. Let $k := \lfloor \log_m(n+1) \rfloor \geq 0$ and write

$$n + 1 = m^k + \sum_{j=0}^{k-1} b_j (m-1) m^j + b_{-1}. \tag{9}$$

This gives a useful mixed-radix expansion of $n+1-m^k$. Here $0 \le b_{-1} \le m-2$ and $0 \le b_j \le m-1$ $(0 \le j \le k-1)$ are integers. For $k \ge 1$, define

$$\rho(n) := [m^{k-1} + b_{k-2}(m-1)m^{k-2} + \cdots + b_0(m-1)m^0 + b_{-1}] - 1.$$

Note that the jth iterate of ρ is

$$\rho_j(n) = [m^{k-j} + b_{k-j-1}(m-1)m^{k-j-1} + \cdots + b_0(m-1)m^0 + b_{-1}] - 1,$$

for $0 \le j \le k$. (In particular, $\rho_0(n) = n$, $\rho_1 = \rho$, and $\rho_k(n) = b_{-1}$.) If $k \ge 1$, (8) can be written

$$R_n = \binom{n}{m-1} r_k^{b_{k-1}} r_{k-1}^{m-1-b_{k-1}} R_{\rho(n)}.$$

Iterating this gives

$$R_n = R_{\rho_k(n)} \prod_{j=0}^{k-1} \left[\binom{\rho_j(n)}{m-1} r_{k-j}^{b_{k-1-j}} r_{k-1-j}^{m-1-b_{k-1-j}} \right]$$

if $k \ge 1$ (i.e., if $n \ge m-1$) and hence

$$R_n = \left(\prod_{j=0}^{k-1} \binom{\rho_j(n)}{m-1} \right) \left(\prod_{j=0}^{k-1} r_j^{(m-1)-(b_j - b_{j-1})} \right) r_k^{b_{k-1}}$$

in general.

We will use this last expression for R_n to derive a suitable upper bound. From Proposition 4.1, (7), and (6), it is clear that for perfect $n \ge 0$,

$$R_n \le \exp[s_m(1)(n+1)],$$

i.e.,

$$r_j \le \exp[s_m(1)m^j] \text{ for } j \ge 0.$$

Hence

$$\begin{aligned}
&\left(\prod_{j=0}^{k-1} r_j^{(m-1)-(b_j - b_{j-1})} \right) r_k^{b_{k-1}} \\
&\quad \le \quad \exp\left[s_m(1) \left(\sum_{j=0}^{k-1} m^j((m-1) - (b_j - b_{j-1})) + b_{k-1}m^k \right) \right] \\
&\quad = \quad \exp\left[s_m(1) \left((m^k - 1) + \sum_{j=0}^{k-1} b_j(m-1)m^j + b_{-1} \right) \right] \\
&\quad = \quad \exp\left[s_m(1)(n+1) \right].
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\prod_{j=0}^{k-1}\binom{\rho_j(n)}{m-1} &< \prod_{j=0}^{k-1}\binom{m^{k+1-j}-1}{m-1} = \prod_{j=2}^{k+1}\binom{m^j-1}{m-1} \\
&= \prod_{j=1}^{k+1}\binom{m^j-1}{m-1} = \prod_{j=1}^{k+1} m^{-(j-1)}\binom{m^j}{m} \\
&\le \prod_{j=1}^{k+1}\left[m^{-(j-1)}\frac{m^{jm}}{m!}\right] = \left(\frac{m^m}{m!}\right)^{k+1}\prod_{j=0}^{k}\left(m^{m-1}\right)^j \\
&= \left(\frac{m^m}{m!}\right)^{k+1}\left(m^{m-1}\right)^{\frac{1}{2}k(k+1)} \le \left(\frac{m^m}{m!}\right)^{k+1}\left(m^{m-1}\right)^{\frac{1}{2}(k+1)^2}.
\end{aligned}$$

Therefore, for $n \ge 0$,

$$\begin{aligned}
R_n \le \exp\Big[&s_m(1)(n+1) + \frac{1}{2}\left((\log_m(n+1))+1\right)^2(m-1)\ln m \\
&+ (\log_m(n+1)+1)\ln\left(\frac{m^m}{m!}\right)\Big].
\end{aligned}$$

This gives the desired upper bound and the result follows. ■

Remark: Comparison of Theorems 2 and 4 reveals a large discrepancy between the maximum and minimum probabilities under the random permutation model. The negative logarithm of the maximum is on the order of n, while that of the minimum is on the order of $n \log n$.

4.3 Computation of $s_m(1)$

The constant $s_m(1) = \sum_{j\ge1} m^{-j} \ln\binom{m^j-1}{m-1}$ is easily computed to a high degree of numerical accuracy. In Table 1, we give, for selected values of m, the value of $s_m(1)$ rounded to seven decimal places. Asymptotics for $s_m(1)$ as $m \to \infty$ are also easy to derive:

$$s_m(1) = \hat{s}_m + O\left(m^{-4}\right),$$

where

$$\begin{aligned}
\hat{s}_m := \; & m^{-1}\ln m + m^{-1} + \frac{1}{2}m^{-2}\ln m - \frac{1}{2}(\ln(2\pi)-1)m^{-2} + \frac{1}{2}m^{-3} \\
&+ \left(\frac{5}{4} - \frac{1}{2}\ln(2\pi)\right)m^{-3} + \frac{1}{2}m^{-4}\ln m.
\end{aligned}$$

For comparison purposes, the values of $\hat{s}_m$ are also shown in Table 1, along with the relative error

$$\text{rel. err.} := \frac{\hat{s}_m - s_m(1)}{s_m(1)}.$$

As suggested by Table 1, $s_m(1)$ vanishes monotonically as m increases.

Proposition 4.2

$$s_m(1) \downarrow 0 \quad \textit{as } m \to \infty.$$

Proof We have

$$s_m(1) = \sum_{j=2}^{\infty} m^{-j} \ln \binom{m^j - 1}{m - 1}. \tag{10}$$

Monotonicity is established by computing $s_m(1)$ for $m = 2, 3, 4$ and showing that each term of (10) decreases monotonically for $m \geq 4$. (In fact, all but the $j = 2$ term are monotone for $m \geq 2$.) We omit the details of this rather straightforward but somewhat tedious argument. ■

Table 1.

m	$s_m(1)$	$\hat{s}_m$	rel. err. (in %)
2	.9457553	.9348476	−1.153
3	.7542073	.7534105	−0.106
4	.6324112	.6324225	+0.002
5	.5476142	.5476926	+0.014
6	.4848493	.4849132	+0.013
7	.4363125	.4363578	+0.010
8	.3975295	.3975611	+0.008
9	.3657445	.3657668	+0.006
10	.3391635	.3391795	+0.005
25	.1707875	.1707881	+0.0004
50	.0988739	.0988739	
100	.0562427	.0562427	
250	.0261235	.0261235	
500	.0144400	.0144400	
1,000	.0079108	.0079108	

5 The number of maximizers of Q

How many trees maximize Q? Call this multiplicity $\mu_n := |M_n^*|$. It is easy to see that

$$\mu_n = 1 \text{ for } 0 \le n \le m-2 \tag{11}$$

and that, for $m-1 \le n \le m^2-2$, μ_n is equal to the number of compositions of $n-(m-1)$ into m nonnegative parts of size at most $m-1$. There are several different expressions obtainable for μ_n, including the following, which is easily derived using generating functions:

$$\mu_n = \sum_{j=0}^{\lfloor \frac{n-(m-1)}{m} \rfloor} (-1)^j \binom{m}{j} \binom{n-jm}{m-1}. \tag{12}$$

Computation of μ_n for larger values of n is facilitated by the expansion at (9). The results (11) and (12) can be expressed in terms of (9) as $\mu_n = 1$ if $k = 0$ and

$$\begin{aligned} \mu_n &= \sum_{j=0}^{b_0+\left\lfloor \frac{b_{-1}-b_0}{m} \right\rfloor} (-1)^j \binom{m}{j} \binom{(b_0+1)(m-1)+b_{-1}-jm}{m-1} \\ &= \sum_{j=0}^{b_0-\mathbf{1}(b_{-1}<b_0)} (-1)^j \binom{m}{j} \binom{(b_0+1)(m-1)+b_{-1}-jm}{m-1}, \end{aligned} \tag{13}$$

if $k = 1$, where $\mathbf{1}(A)$ is the indicator of A. If $n = m^k - 1$ for some $k \ge 0$, then of course $\mu_n = 1$. For general n satisfying $n \ge m^2 - 1$, i.e., $k \ge 2$, with n *not* perfect, it is not hard to see that

$$\begin{aligned} \mu_n &= \left\{ \begin{array}{ll} \binom{m}{b_{k-1}, m-b_{k-1}} & \text{if } \rho(n) = m^{k-1}-1 \\ \binom{m}{b_{k-1}, m-1-b_{k-1}, 1} & \text{if } \rho(n) \ne m^{k-1}-1 \end{array} \right\} \times \mu_{\rho(n)} \\ &= \left\{ \begin{array}{ll} \binom{m}{b_{k-1}} & \text{if } b_{-1} = b_0 = \cdots = b_{k-2} = 0 \\ m\binom{m-1}{b_{k-1}} & \text{if } b_j \ne 0 \text{ for some } -1 \le j \le k-2 \end{array} \right\} \times \mu_{\rho(n)}. \end{aligned}$$

Iterating this, we find

$$\mu_n = m^{k-1} S \prod_{j=1}^{k-1} \binom{m-1}{b_j}$$

if $b_{-1} \neq 0$, where S is the sum on the right at (13), and

$$\mu_n = \left(\prod_{j=1}^{f} \binom{m}{b_j}\right) \left(\prod_{j=f+1}^{k-1} \left[m\binom{m-1}{b_j}\right]\right) S$$

if $b_{-1} = b_0 = \cdots = b_{f-1} = 0 \neq b_f$ for some $f \in \{0, \ldots, k-1\}$.

Rearranging and combining cases, we obtain the following summary of the values of μ_n, the number of m-ary search trees on n keys that maximize Q, for a mutually exclusive and exhaustive list of possibilities for n:

Theorem 5 (a) *If $0 \leq n \leq m-2$, then $\mu_n = 1$.*

Henceforth assume $n \geq m-1$ and consider the expansion

$$n+1 = m^k + \sum_{j=0}^{k-1} b_j (m-1) m^j + b_{-1}, \tag{14}$$

with $k := \lfloor \log_m(n+1) \rfloor \geq 1$.

(b) *If $b_{-1} = b_0 = \cdots = b_{k-1} = 0$, then $\mu_n = 1$.*

Henceforth we adopt the notation

$$b_{-1} = b_0 = \cdots = b_{f-1} = 0 \neq b_f \tag{15}$$

for $f \in \{-1, \ldots, k-1\}$.

(c) *If $f = -1$ or $f = 0$, then*

$$\begin{aligned} \mu_n &= m^{k-1} \left[\sum_{j=0}^{b_0 - \mathbf{1}(b_{-1} < b_0)} (-1)^j \binom{m}{j} \binom{(b_0+1)(m-1) + b_{-1} - jm}{m-1}\right] \\ &\quad \times \prod_{j=1}^{k-1} \binom{m-1}{b_j}. \end{aligned}$$

(d) *If $1 \leq f \leq k-1$, then*

$$\mu_n = m^{k-1-f} \binom{m}{b_f} \prod_{j=f+1}^{k-1} \binom{m-1}{b_j}.$$

Example: Choosing $m = 2$, we *always* have $b_{-1} = 0$, and (14) gives the binary expansion of $n + 1$. If n is not perfect (and thus $n \geq 2$), then the index f defined at (15) satisfies $0 \leq f \leq k - 1$ and equals π_{n+1}, where $2^{\pi_{n+1}}$ is the highest power of 2 that divides $n+1$. If $\pi_{n+1} = 0$, then $b_0 = 1$ and part (c) of Theorem 5 asserts $\mu_n = 2^{k-1}2 = 2^k = 2^{\lfloor \lg(n+1) \rfloor - \pi_{n+1}}$. If $\pi_{n+1} \geq 1$, then $b_f = 1$ and part (d) asserts $\mu_n = 2^{k-1-f}2 = 2^{k-f} = 2^{\lfloor \lg(n+1) \rfloor - \pi_{n+1}}$. Since we also have

$$\mu_n = 1 = 2^{\lfloor \lg(n+1) \rfloor - \pi_{n+1}}$$

if n is perfect, we have rederived Remark 2.4 in Fill (1995) from the above theorem.

Acknowledgments

The authors thank an anonymous referee for suggestions leading to improved exposition.

References

Bayer, R. and McCreight, E. (1972). Organization and maintenance of large ordered indexes. *Acta Inform.* **1** 173–189.

Devroye, L. (1990). On the height of random m-ary search trees. *Rand. Struct. Alg.* **1** 191–203.

Fill, J. A. (1995). On the distribution for binary search trees under the random permutation model. Technical Report #537, Department of Mathematical Sciences, The Johns Hopkins University. *Random Structures and Algorithms*, to appear.

Fill, J. A. and Dobrow, R. P. (1995). The number of m-ary search trees on n keys. Technical Report #543, Department of Mathematical Sciences, The Johns Hopkins University.

Knuth, D. (1973a). *The Art of Computer Programming,* Vol. 1: *Fundamental Algorithms,* 2nd ed. Addison-Wesley, Reading, Mass.

Knuth, D. (1973b). *The Art of Computer Programming,* Vol 3: *Sorting and Searching,* 2nd ed. Addison-Wesley, Reading, Mass.

Mahmoud, H. (1992). *Evolution of Random Search Trees.* Wiley, New York.

Mahmoud, H. and Pittel, B. (1989). Analysis of the space of search trees under the random insertion algorithm. *J. Alg.* **10** 52–75.

Muntz, R. and Uzgalis, R. (1971). Dynamic storage allocation for binary search trees in a two-level memory. *Proceedings of Princeton Conference on Information Sciences and Systems,* Vol. 4, 345–349.

Pittel, B. (1994). Note on the heights of random recursive trees and random m-ary search trees. *Rand. Struct. Alg.* **5** 337–347.

Robert P. Dobrow
Division of Mathematics and Computer Science
Northeast Missouri State University
Kirksville, MO 63501
bdobrow@cs-sun1.nemostate.edu

James Allen Fill
Department of Mathematical Sciences
The Johns Hopkins University
Baltimore, MD 21218-2692
jimfill@jhu.edu