A Dichotomous Search for a Geometric Random Variable

REFAEL HASSIN

Tel-Aviv University, Tel-Aviv, Israel

(Received March 1980; accepted December 1983)

We are given a two-state system that starts in state 0, ends in state 1, and makes a single transition from state 0 to state 1 during N periods. If the system is in state 0, it moves to state 1 in the next period with a known positive time-independent probability. Once it reaches state 1, it remains there. By observing the state of the system at some intermediate period, we can learn whether this transition occurred earlier or not. An optimal search strategy minimizes the expected number of observations needed to locate the exact transition time. In this paper we show how to compute efficiently and how to approximate the optimal strategy. Applications to the problem arise in the areas of quality control and maintenance of communication and supply lines.

IN A TWO-STATE (0-1) stochastic process $\{I_j,\ j = 0, 1, \cdots\}$ with initial state $I_0 = 0$, we observe that $I_N = 1$. Once in state 1 the system remains there. Moreover, if it is in state 0, it stays there with probability $\alpha < 1$, i.e. $P(I_j = 0)/(I_{j-1} = 0) = \alpha$. Thus, there is a unique time period $t \in \{1, \cdots, N\}$ satisfying

$$I_0 = I_1 = \cdots = I_{t-1} = 0 \quad \text{and} \quad I_t = I_{t+1} = \cdots = I_N = 1. \qquad (1)$$

At a fixed cost (say 1 dollar), we can observe I_n for any single choice of n and thus can determine whether $t \leq n$ or $t > n$.

Our objective is to develop a search strategy that minimizes the expected cost of the search for the number t satisfying Equation 1. Such a strategy is said to be "optimal."

Applications of dichotomous search to computer codes are well-known; for example, see Gilbert and Moore [1959] and Knuth [1973]. We now describe some additional applications of our problem.

(a) An item produced by a certain machine is found to be defective. It is known to be the Nth item produced since the machine was last inspected and found to be operating properly. The distribution of failures in the machine is known to be exponential with a mean equal to t production periods (one item is produced during each period). Let $\alpha = \exp(-1/t)$. Then for any $k = 1, 2, \cdots$, α is the probability that the kth

Subject classification: 751 dichotomous search.

0030-364X/84/3202-0423 $01.25

item is not defective, given that the $(k-1)$st item is not defective. If the machine fails during the kth production period, then items k, $k+1$, $k+2, \cdots$, will be defective. The producer's goal is to minimize the expected number of inspections required to locate the first defective item. It searches by inspecting items in sequence and learns whether the machine failed before or after completing the production of this item. The search cost may be high, especially when the items have already been supplied to customers and the information about the Nth item has come from a customer's complaint.

(b) A communication system consists of $N-1$ transmitting stations. A message is sent from a source to the first transmitting station, then to the second station and so forth, until it is sent from the $(N-1)$st station to the final destination. The number of messages a station transmits until it fails is known to be geometrically distributed with a parameter α. Given that a message has been sent from the source and not arrived at the destination, our goal is to locate the defective transmitter by checking whether the message reached the Nth station or not, thus learning whether the transmission failure occurred before or after the Nth station.

(c) The value of a random variable that is geometrically distributed with a parameter α must be found using dichotomous search. This situation is a special case of the problem with $N = \infty$.

In order to construct a lower bound for the average number of comparisons required to sort a table of N items, Morris (1969) has investigated the case in which the a priori probability for the transition time from state 0 to state 1 is uniform on $1, 2, \cdots, N$ (i.e., $\Pr(t = j) = 1/N$ for $j = 1, \cdots, N$). This case is a limiting version of our problem where $\alpha \to 1$. Several other papers on this subject are available with the assumption of a uniform a priori probability distribution. Gal [1974] considered the problem as a zero-sum two-person game in which a searcher tries to identify an integer that had been chosen by a hider from $\{1, \cdots, N\}$. Murakami [1976] finds a minimax optimal strategy for situations in which the searcher incurs the costs of travel from one search point to another in addition to the cost of the inspection itself. Cameron and Narayanamurthy [1964] and Murakami [1971] treat the case in which the costs associated with overestimates and underestimates of the transition point differ. Hassin and Megiddo [1980] treated generalizations of the problem in which the objective is to locate a vector of unknowns $t_1, \cdots, t_k$, where $t_j \in \{1, \cdots, N\}$ for $j = 1, \cdots, k$.

Several authors have discussed the more general problem in which the distribution of t is arbitrary. The most efficient method, which was developed by Hu and Tucker [1971] and improved by Knuth [1973], requires $0(N \log N)$ operations to construct the optimal search strategy. However, as we show in this paper, a dynamic programming procedure that is inefficient in general (cf. Gilbert and Moore) becomes very efficient for our problem, and requires only $0(N)$ operations. We also obtain a simple formula that supplies an approximate solution, which is sufficiently accurate for most real applications.

1. DYNAMIC PROGRAMMING FORMULATION

Let q_j denote the probability that we observe state 0 after j transitions. Then by our assumptions,

$$q_j = P_r(I_j = 0/I_N = 1) = \alpha^j(1 - \alpha^{N-j})/(1 - \alpha^N). \tag{2}$$

Let $f(N)$ denote the expected cost of search under the optimal strategy, in a problem of length N; then $f(1) = 0$ and

$$f(N) = 1 + \min_{x=1,\cdots,N-1}\{q_x f(N - x) + (1 - q_x)f(x)\},$$

or

$$f(N) = 1 + \min_{x=1,\cdots,N-1}\{[\alpha^x(1 - \alpha^{N-x})/(1 - \alpha^N)]f(N - x) + [(1 - \alpha^x)/(1 - \alpha^N)]f(x)\}. \tag{3}$$

Define $F(N) = (1 - \alpha^N)f(N)$; then $F(1) = 0$ and Equation 3 becomes

$$F(N) = (1 - \alpha^N) + \min_{x=1,\cdots,N-1}\{\alpha^x F(N - x) + F(x)\}. \tag{4}$$

Let $x_N{}^*$ denote the argument that minimizes the righthand side of (4); then,

$$F(N) = (1 - \alpha^N) + \alpha^{x_N{}^*}F(N - x_N{}^*) + F(x_N{}^*).$$

Remark. We are now in a position to present another application of the problem. Consider the same problem, but with I_N unknown. Thus, we still want to identify the integer t such that $I_j = 0$ if, and only if, $j \in \{0, 1, \cdots, t - 1\}$, but now $t = N + 1$ is also possible. This problem can be solved by dynamic programming involving the function $f(N)$ defined above. Let $h(N)$ be the expected cost of the search under an optimal policy; then,

$$h(N) = 1 + \min_x\{\Pr(I_x \neq 0)f(x) + \Pr(I_x = 0)h(N - x)\}$$
$$= 1 + \min_x\{(1 - \alpha^x)f(x) + \alpha^x h(N - x)\}.$$

We do not deal with this extension of the problem, but note only that knowledge of $f(x)$ is necessary to solve it.

2. 2-TREES

A *2-tree* is a rooted tree, as shown in Figure 1, in which every node has two or zero edges emanating upward from it and every node, except the root, is met by an edge from a lower node. Nodes having no edges emanating upward from them are called *terminal nodes.* We number the terminal nodes of a 2-tree from left to right, as shown in Figure 1.

The number of edges on the path from the root to node j is called the *level* of this node, and is denoted by $l(j)$. In a 2-tree with N terminal nodes,

$$\sum_{j=1}^{N} (1/2)^{l(j)} = 1. \tag{5}$$

For example, the tree of Figure 1 has $l(1) = l(2) = l(3) = 2$, $l(4) = 3$, and $l(5) = l(6) = 4$, so that $1/4 + 1/4 + 1/4 + 1/8 + 1/16 + 1/16 = 1$.

A 2-tree satisfying $l(1) \leq l(2) \leq \cdots \leq l(N)$ is said to be *nondecreasing* (e.g., the tree of Figure 1).

LEMMA 1. *There is a one-to-one correspondence between nondecreasing sequences of integers satisfying* (5) *and nondecreasing* 2*-trees.*

Proof. By definition $l(1) \leq l(2) \leq \cdots \leq l(N)$ for a nondecreasing 2-tree. It remains to prove that for every nondecreasing sequence of

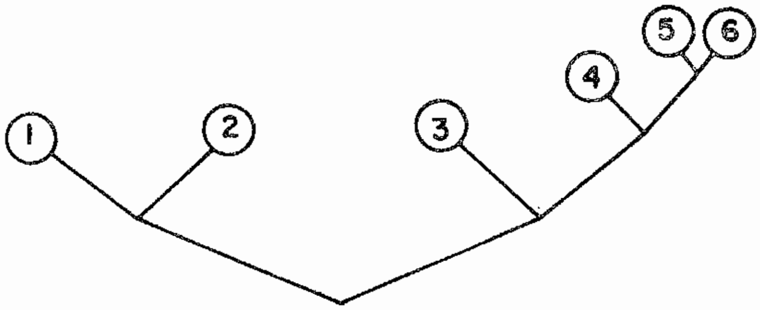

Figure 1

integers satisfying (5), there exists a 2-tree. We first prove that such a sequence satisfies $\sum_{j=1}^{k} (1/2)^{l(j)} = 1/2$ for some integer k ($k = 2$ in Figure 1). Suppose no such k exists. Then, there is an integer m such that $1/2 > \sum_{j=1}^{m} (1/2)^{l(j)} > 1/2 - (1/2)^{l(m+1)}$. Multiplying by $2^{l(m)}$, we obtain

$$2^{l(m)-1} > \sum_{j=1}^{m} 2^{l(m)-l(j)} > 2^{l(m)-1} - (1/2)^{l(m+1)-l(m)}.$$

These inequalities contradict the assumption that $l(1) \leq l(2) \leq \cdots \leq l(m+1)$, since in this case all terms except for $(1/2)^{l(m+1)-l(m)}$ are integers. The 2-tree that corresponds to $l(1), \cdots, l(N)$ can be constructed recursively by locating k and constructing the 2-trees that correspond to $l(1) - 1, \cdots, l(k) - 1$ and to $l(k + 1) - 1, \cdots, l(N) - 1$.

3. 2-TREE FORMULATION

A search strategy may be described as follows: First, we inspect I_x and learn whether $t \leq x$ (if $I_x = 1$) or $t > x$ (if $I_x = 0$). We next choose to inspect the set $\{1, \cdots, x\}$ if $t \leq x$, or the set $\{x + 1, \cdots, N\}$ if $t > x$.

Thus, every inspection partitions the set of possible solutions into two sets and the search process is repeated until the set of possible solutions contains a single solution. This process may be described by a 2-tree. For example, the tree of Figure 1 describes a search strategy for $N = 6$. First, we inspect I_2 and discover whether the solution is in $\{1, 2\}$ or in $\{3, 4, 5, 6\}$. In the first case, we next inspect I_1, while in the second case, we inspect I_3. If $I_3 = 0$, then I_4 is inspected to find whether $t = 4$ or $t \geq 5$, and finally, if $I_4 = 0$, inspection of I_5 determines whether $t = 5$ or $t = 6$. Notice that every inspection corresponds to a node that is one edge higher than the previous one, until the tth terminal node is reached. Therefore, the cost of the search is $l(t)$.

Let $p(k)$ be the probability that the transition from state 0 into state 1 was the kth. Then, by our assumptions

$$p(k) = \alpha^{k-1}(1 - \alpha)/(1 - \alpha^N).$$

Thus,

$$f(N) = \sum_{k=1}^{N} p(k)l(k) = [(1 - \alpha)/(1 - \alpha^N)] \sum_{k=1}^{N} l(k)\alpha^{k-1},$$

and, also,

$$F(N) = (1 - \alpha) \sum_{k=1}^{N} l(k)\alpha^{k-1}. \quad (6)$$

The value of $x_N{}^*$ that corresponds to $l(1), \cdots, l(N)$ is given by

$$\sum_{k=1}^{x_N^*} (1/2)^{l(k)} = 1/2. \quad (7)$$

The problem can be reformulated as follows: Find a 2-tree that minimizes $\sum_{k=1}^{N} l(k)\alpha^{k-1}$. We call such a tree *optimal.*

THEOREM 1. *The optimal tree is nondecreasing and the levels of its terminal nodes solve* (8a)–(8c):

$$\text{Minimize} \quad \sum_{k=1}^{N} l(k)\alpha^{k-1} \quad (8a)$$

subject to:

$$\sum_{k=1}^{N} (1/2)^{l(k)} = 1 \quad (8b)$$

$$l(1), \cdots, l(N) \text{ are integers.} \quad (8c)$$

Proof. Since (8b) and (8c) are symmetrical with respect to $l(1), \cdots, l(N)$, and since α^{k-1} decreases in k, the solution to (8a)–(8c) is a nondecreasing sequence. By Lemma 1, there exists a 2-tree with these levels, and since every 2-tree satisfies (8b) and (8c), this tree is optimal.

COROLLARY 1. $x_N{}^* \leq N/2$.

Proof. Suppose the root of the optimal tree with the edges emanating from it are removed. Two 2-trees with $x_N{}^*$ and $N - x_N{}^*$ terminal nodes result. Hence $\sum_{k=1}^{x_N^*} (½)^{l(k)-1} = \sum_{k=x_{N^*}+1}^{N} (½)^{l(k)-1} = 1$. Since $\{l(j)\}$ is nondecreasing, $x_N^* \leq N/2$.

4. PROPERTIES OF THE SOLUTION

In this section we introduce theorems concerning optimal strategies. The main result is a restatement of Equation 4 which allows more efficient computation of the optimal policy. In our discussion, we let $R(k)$ denote the unique root of the equation $\alpha^k + \alpha^{k+1} = 1$ in the interval $0 < \alpha < 1$.

THEOREM 2. *If $\alpha < R(1)$, then for every N, $x_N^* = 1$, and*

$$F(N) = 1 + \alpha + \alpha^2 + \cdots + \alpha^{N-2} - (N-1)\alpha^N. \tag{9}$$

Proof. Clearly $x_2^* = 1$ and $F(2) = (1 - \alpha^2)f(2) = 1 - \alpha^2$. Suppose the theorem holds for $N = 1, 2, \cdots, k-1$. Using equation (9) for $F(x)$ and $F(k-x)$, we see that Equation 4 becomes:

$$\begin{aligned} F(k) = \min\{&[1 + \alpha + \alpha^2 + \cdots + \alpha^{x-2} - (x-1)\alpha^x] \\ &+ [\alpha^x + \alpha^{x+1} + \cdots + \alpha^{k-2} - (k-x-1)\alpha^k]\} \\ &+ (1-\alpha^k) = \min\{1 + \alpha + \cdots + \alpha^{k-2} - (k-1)\alpha^k + g(x)\} \end{aligned}$$

where the minimum is taken over $x = 1, \cdots, k-1$, and $g(x) = 1 - \alpha^{x-1} - (x-1)(\alpha^x - \alpha^k)$. The condition $\alpha < R(1)$ implies $1 - \alpha - \alpha^2 > 0$. Thus, for $x \geq 2$, $1 > \alpha + \alpha^2 > (\alpha^2 + \alpha^3) + \alpha^2 > (\alpha^3 + \alpha^4) + \alpha^3 + \alpha^2 > \cdots > \alpha^{x-1} + (x-1)\alpha^x$.

Therefore, for $x \geq 2$, $g(x) > 0 = g(1)$ and for every value of k, $g(x)$ is minimal when $x = 1$. In this case the equation for $F(k)$ also becomes equivalent to (9), thus completing the inductive proof.

Figure 2 represents the optimal tree for $\alpha < R(1)$.

THEOREM 3. *In an optimal tree, $l(k+1) \in \{l(k), l(k)+1\}$.*

Proof. Recalling Theorem 1, we need only prove that $l(k+1) \leq l(k) + 1$. By Theorem 3 and Figure 2, the condition holds for $\alpha \leq R(1)$. Suppose $\alpha > R(1)$ and $l(k+1) > l(k) + 1$ for some $k \in \{1, \cdots, N-1\}$ and a nondecreasing tree T. Clearly, $l(N-1) = l(N)$ in every nondecreasing tree, and thus we can define $r = \min\{i \mid i \geq k+1, l(i) = l(i+1)\} \leq N-1\}$. Consider the sequence $l'(1), \cdots, l'(N)$ defined by

$$l'(j) = \begin{cases} l(k) + 1 & j = k, k+1 \\ l(j-1) & k+1 < j \leq r \\ l(r+1) - 1 = l(r) - 1 & j = r+1 \\ l(j) & \text{otherwise.} \end{cases}$$

This sequence is nondecreasing and

$$\sum_{j=1}^{N} (1/2)^{l'(j)} = 2\cdot(1/2)^{l(k)+1} + \sum_{j=k+2}^{r} (1/2)^{l(j-1)} + (1/2)^{l(r)-1} + \sum_{j\notin\{k,\cdots,r+1\}} (1/2)^{l(j)}$$
$$= (1/2)^{l(k)} + \sum_{j=k+1}^{r-1} (1/2)^{l(j)} + (1/2)^{l(r)} + (1/2)^{l(r+1)} + \sum_{j\notin\{k,\cdots,r+1\}} (1/2)^{l(j)} = 1.$$

Therefore, by Lemma 1, this sequence corresponds to a tree T'. Figure 3 illustrates the transformation with $N = 8$, $k = 2$ and $r = 3$. Let F and F' be the costs associated with T and T', respectively. By (6),

$$F' - F = (1-\alpha)[\alpha^{k-1} - (l(k+1) - l(k) - 1)\alpha^{k} - \sum_{j=k+2}^{r} (l(j) - l(j-1))\alpha^{j-1} - \alpha^{r}],$$

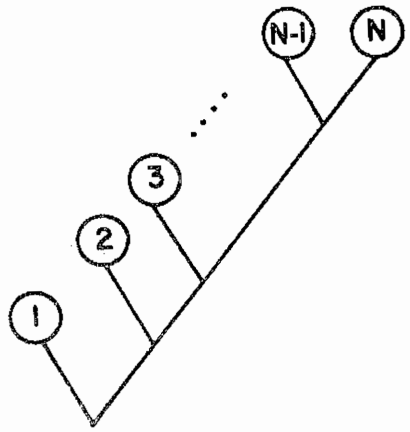

Figure 2

and since $l(k+1) - l(k) \geq 2$ and $l(j) - l(j-1) \geq 1$ for $j = k+2, \cdots, r$, $F' - F \leq (1-\alpha)(\alpha^{k-1} - \alpha^{k} - \alpha^{k+1}) = (1-\alpha)\alpha^{k-1}(1-\alpha-\alpha^{2})$. This difference is negative since $\alpha > R(1)$, thus T cannot be optimal.

Let $k_1, k_2, \cdots, k_m$ be the terminal nodes of a tree satisfying Theorems 1 and 3 in which $l(k_j) < l(k_j + 1)$, and let F be the cost associated with this tree. Then,

$$F = (1-\alpha)[l(1)(1 + \alpha + \cdots \alpha^{k_1-1}) + (l(1)+1)(\alpha^{k_1} + \cdots \alpha^{k_2-1}) + \cdots + (l(1)+m)(\alpha^{k_m} + \cdots + \alpha^{N-1})],$$

or

$$F = l(1) + \alpha^{k_1} + \alpha^{k_2} + \cdots + \alpha^{k_m} - (l(1)+m)\alpha^{N}. \qquad (10)$$

For example, the tree in Figure 1 has $l(1) = 2$, $m = 2$, $k_1 = 3$, and $k_2 = 4$, so that $F = 2 + \alpha^3 + \alpha^4 - 4\alpha^6$.

The next theorem can be used for approximating the solution by the construction of one of the few (often unique) 2-trees satisfying it.

THEOREM 4. *Let δ_j $j = 0, \cdots, m$ denote the number of terminal nodes $i \in \{1, \cdots, N\}$ with $l(i) = l(1) + j$. Suppose $R(k-1) < \alpha < R(k)$. Then, the optimal tree satisfies*

$$\delta_j \leq k + 1 \qquad j = 0, \cdots, m$$

and

$$\delta_j \geq k - 1 \qquad j = 1, \cdots, m-1.$$

Proof. Let T be a 2-tree with cost F and levels $l(1), \cdots, l(N)$, such

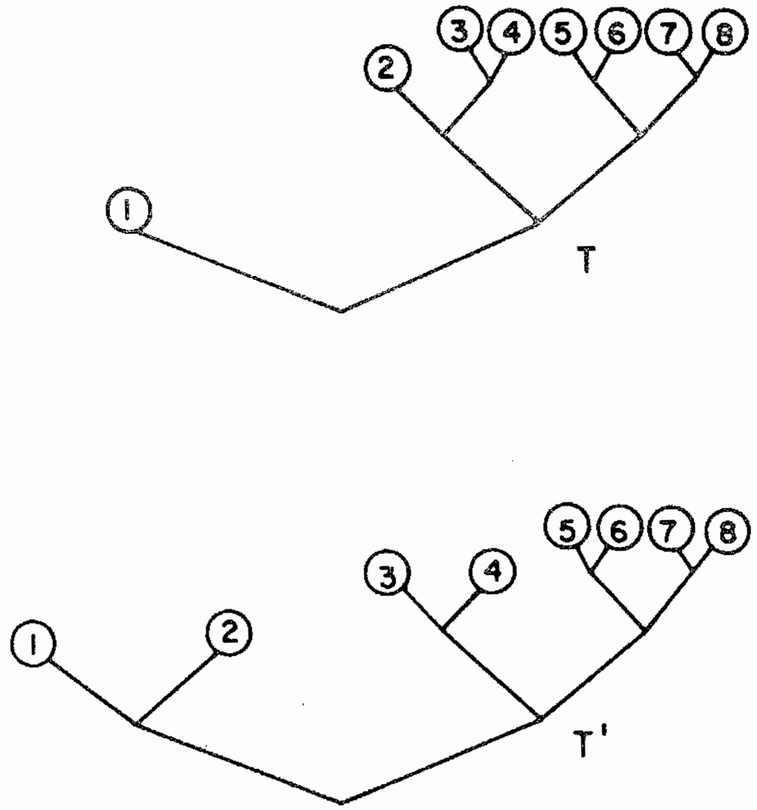

Figure 3

that for some i, $\bar{l}$, and $\delta \geqq 3$

$$l(i-1) = \bar{l} - 1, \qquad l(i) = l(i+1) = \cdots = l(i+\delta-1) = \bar{l},\ l(i+\delta) = \bar{l} + 1.$$

Let T' be the 2-tree with cost F' and the levels $l'(1), \cdots, l'(N)$ satisfying

$$l'(j) = \begin{cases} \bar{l} - 1 & j = i \\ \bar{l} + 1 & j = i + \delta - 2, \qquad i + \delta - 1 \\ l(j) & \text{otherwise.} \end{cases}$$

Thus T' is obtained from T by "merging" nodes i and $i + 1$ and "splitting" node $i + \delta - 1$. Figure 4 illustrates the transformation with $N = 8$, $\bar{l} = 3$, $\delta = 5$ and $i = 2$. By (6), $F - F' = (1 - \alpha)(\alpha^{i-1} - \alpha^{i+\delta-3} - \alpha^{i+\delta-2})$. If $\delta \geq k + 2$, then $F - F' \geq (1 - \alpha)\alpha^{i-1}(1 - \alpha^k - \alpha^{k+1})$, and since $\alpha < R(k)$, this expression is positive and T cannot be optimal. Note that T' has $\delta - 3$ nodes with $l(j) = \bar{l}$. If $\delta - 3 \leq k - 2$, then $F - F' \leq (1 - \alpha)\alpha^{i-1}(1 - \alpha^{k-1} - \alpha^k)$, and since $\alpha > R(k - 1)$, this expression is negative and T' cannot be optimal. Thus, only trees satisfying the condition of the theorem can be optimal.

The following extension of Theorem 2 gives an upper bound for $x_N{}^*$:

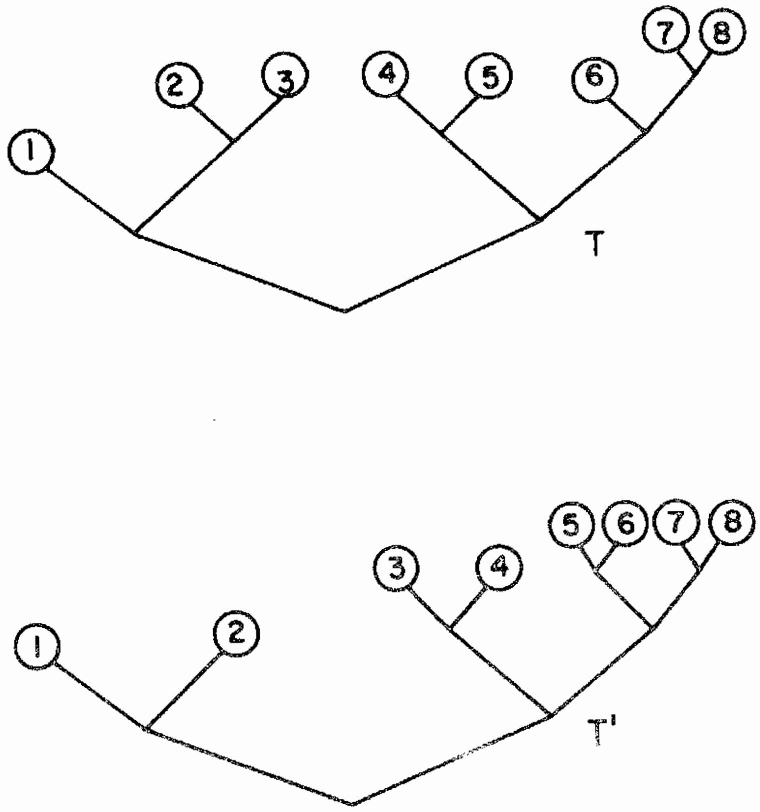

Figure 4

THEOREM 5. *If $\alpha < R(k)$, then $x_N{}^* \leq k$ for every N.*

Proof. Suppose $\alpha < R(k)$ and $x_N{}^* = x \geq k + 1$. Let Δ_e be the number of terminal nodes $j \leq x$ such that $l(j) = l(x)$. Let Δ_r be the number of terminal nodes $j > x$ such that $l(j) = l(x)$. By Theorem 4, no more than $k + 1$ terminal nodes have level $l(x)$, thus $\Delta_r + \Delta_e \leq k + 1 \leq x$ or $\Delta_r + x \leq 2x - \Delta_e$. However, since $l(j)$ is nondecreasing and by (7),

$$\tfrac{1}{2} = \sum_{j=1}^{x} (\tfrac{1}{2})^{l(j)} > (x - \Delta_e)(\tfrac{1}{2})^{l(x)-1} + \Delta_e(\tfrac{1}{2})^{l(x)}$$
$$= (2x - \Delta_e)(\tfrac{1}{2})^{l(x)} \geq (\Delta_r + x)(\tfrac{1}{2})^{l(x)}.$$

By Theorem 4, no more than $k + 1$ terminal nodes have the same level, and since $l(j)$ is nondecreasing

$$\tfrac{1}{2} = \sum_{j=x+1}^{N} (\tfrac{1}{2})^{l(j)} < \Delta_r(\tfrac{1}{2})^{l(x)} + \sum_{i=1}^{\infty} (k + 1)(\tfrac{1}{2})^{l(x)+i}$$
$$= \Delta_r(\tfrac{1}{2})^{l(x)} + (k + 1)(\tfrac{1}{2})^{l(x)} \leq (\Delta_r + x)(\tfrac{1}{2})^{l(x)}.$$

This inequality contradicts the previous one.

Theorem 6, which is the major result of this section, allows more efficient computation of the optimal solution:

THEOREM 6. $x^*_{N+1} \in \{x_N^*, x_N^* + 1\}$.

Let T_N and T_{N+1} denote the optimal trees with N and $N + 1$ terminal nodes, respectively. Let $l(1), \cdots, l(N)$ and $l'(1), \cdots, l'(N + 1)$ be the levels corresponding to these trees. The theorem claims the existence of an index k such that $l'(j) = l(j)$ for $j = 1, \cdots, k - 1$, $l'(k) = l'(k + 1) = l(k)$, and $l'(j) = l(j - 1)$ for $j = k + 1, \cdots, N + 1$. Graphically, we construct T_{N+1} from T_N by "splitting" a single terminal node. To see how this observation follows from the theorem's statement, note that if $x^*_{N+1} = x_N^*$, then $l'(j) = l(j)$ for $j = 1, \cdots, x_N^*$; if $x^*_{N+1} = x_N^* + 1$, then $l'(j) = l(j - 1)$ for $j = x_N^* + 1, \cdots, N + 1$; therefore, the index k can be determined by successive examinations of the remainder of the sequence.

Proof. The theorem clearly holds for $N = 2$, and we now show that if it holds for $N = 1, \cdots, k$, then it holds for $N = k + 1$. Let $x(k) = x$ and $x(k + 1) = x + R$, then we want to prove that $R \in \{0, 1\}$. We prove only that $R \leq 1$, since the proof of $R \geq 0$ is very similar.

Suppose that $R \geq 2$, then $x(k + 1) > x(k) + 1$ and $(k + 1) - x(k + 1) < k - x(k)$, and we can define

$$p = \begin{cases} x & \text{if } l(x) \neq l'(x + R) \\ \min\{i \mid i \leq x,\ l(j) = l'(j + R) \\ \quad \text{for } j = i + 1, \cdots, x\} & \text{if } l(x) = l'(x + R), \end{cases}$$

$$q = \begin{cases} x + 1 & \text{if } l(x + 1) \neq l'(x + R + 1) \\ \max\{i \mid i \geq x + 1,\ l(j) = l'(j + R) \\ \quad \text{for } j = x + 1, \cdots, i - 1\} & \text{if } l(x + 1) = l'(x + R + 1), \end{cases}$$

$$r = \min\{i \mid i \geq q + 1,\ l(i) = l(i - 1)\}.$$

By Theorem 3, $l(x + 1) \in \{l(x), l(x) + 1\}$ and $l'(x + R + 1) \in \{l'(x + R), l'(x + R) + 1\}$. By the induction hypothesis applied to the left and right components of T_N and T_{N+1}, $l'(x + R + 1) \leq l(x + 1)$ and $l'(x + R) \geq l(x)$, so that either $l(x) = l'(x + R)$ or $l(x + 1) = l'(x + R + 1)$, and

$q - p \geq 2$. By the definitions of p, q and r, we have

$$\begin{aligned} l(p) + 1 &= l'(p + R), \\ l(j) &= l'(j + R) \qquad j = p + 1, \cdots, q - 1, \\ l(q) &= l'(q + R) + 1, \\ l(j) &= l(j - 1) + 1 \qquad j = q + 1, \cdots, r - 1, \quad \text{and} \\ l(r) &= l(r - 1). \end{aligned}$$

Let $\bar{T}_N$ be the tree with levels $\bar{l}(1), \cdots, \bar{l}(N)$ and cost $\bar{F}(N)$, obtained from T_N by splitting node p and merging nodes $r - 1$ and r. Thus,

$$\bar{l}(j) = \begin{cases} l(p) + 1 & j = p, p + 1, \\ l(j - 1) & j = p + 2, \cdots, r - 1, \\ l(r) - 1 & j = r, \\ l(j) & \text{otherwise.} \end{cases}$$

Since T_N is optimal, (10) implies that

$$\bar{F}(N) - F(N) = (\alpha^{p-1} - \alpha^p) + \textstyle\sum_{p<k_j<q-1} (\alpha^{k_j+1} - \alpha^{k_j}) + \textstyle\sum_{j=q-1}^{r-3} (\alpha^{j+1} - \alpha^j) + (\alpha^r - \alpha^{r-2}) \geq 0.$$

Since $0 < \alpha < 1$, the above inequality implies that

$$\alpha^{p-1}(1 - \alpha) - (1 - \alpha) \textstyle\sum_{p<k_j<q-1} \alpha^{k_j} - \alpha^{r-2}(1 - \alpha^2) \geq 0. \quad (11)$$

Let $p' = \min\{i \mid i \leq p + R,\ l'(i) = l'(i - 1) = l'(p + R)\}$, and let

$$q' = \min\{i \mid i \geq q + R,\ l'(i) = l'(q + R) + 1\}.$$

Let $\bar{T}_{N+1}$ be the tree with levels $\bar{l}'(1), \cdots, \bar{l}'(N + 1)$ and cost $\bar{F}(N + 1)$ obtained from T_{N+1} by splitting node $q' - 1$ and merging nodes $p' - 1$ and p'. Let $\bar{k}_j'$ be the corresponding indices. Then,

$$\bar{k}_j' = \begin{cases} p' - 1 & k_j' = p' - 2, \\ k_j' - 1 & j \in J = \{i \mid p' < k_j' < q'\}, \\ q' - 3 & k_j' = q' - 1, \\ k_j' & \text{otherwise.} \end{cases}$$

Since T_{N+1} is optimal, (10) implies that

$$F(N + 1) - \bar{F}(N + 1) = (\alpha^{p'-2} - \alpha^{p'-1}) + \textstyle\sum_{j\in J} (\alpha^{k_j'} - \alpha^{k_j'-1}) + (\alpha^{q'-1} - \alpha^{q'-3}) \leq 0.$$

Therefore, $\alpha^{p'-2}(1 - \alpha) - [(1 - \alpha)/\alpha] \sum_{j\in J} \alpha^{k_j'} - \alpha^{q'-3}(1 - \alpha^2) \leq 0$. Dividing by α^{R-1} and noting that $p' \leq p + r$, $k_j' = k_j + R$ $j \in J$, $q' \geq q + R$ and $0 < \alpha < 1$, we obtain a contradiction with (11).

COROLLARY. *Equation 4 may be modified as follows:*

$$F(N) = (1 - \alpha^N) + \min_{x=x^*_{N-1},x^*_{N-1}+1}\{\alpha^x F(N-x) + F(x)\}.$$

Only x^*_{N-1} and $x^*_{N-1}+1$ are considered as candidates to be optimal, and thus only $0(N)$ operations are needed to construct the optimal strategy.

We conclude this section with a theorem concerning the optimal solution obtained when α is close to 1.

THEOREM 7. *Let ρ and r be the nonnegative integers satisfying $\rho < 2^r$ and $N = 2^r + \rho$. Suppose $\alpha > R(N)$. Then,*

$$x_N{}^* = \begin{cases} 2^{r-1} & \rho \leq 2^{r-1} \\ \rho & \rho > 2^{r-1}, \end{cases}$$

and

$$F(N) = r + \alpha^{N-2\rho} - (r+1)\alpha^N. \qquad (12)$$

Proof. Suppose $\alpha > R(N)$. Then by Theorem 4, $l(1)$ and $l(N)$ cannot differ by more than 1. Therefore, the optimal tree satisfies

$$l(j) = \begin{cases} r & j \leq N - 2\rho \\ r+1 & j \geq N - 2\rho, \end{cases}$$

so that $\sum_{i=1}^{N} (½)^{l(j)} = (N - 2\rho)\cdot(½)^r + 2\rho(½)^{r+1} = (N-\rho)(½)^r = 1$. The equations for $x_N{}^*$ and $F(N)$ result from (7) and (10), respectively.

It is interesting to note that even for α very close to 1, $x_N{}^*$ may be smaller than $\lfloor N/2 \rfloor$. For example, x^*_{12} is always less than or equal to 4. The limiting value of $f(N)$, which is the expected cost of the search when the a priori distribution is uniform, is easily obtained from (12):

$$\lim_{\alpha\to 1} f(N) = \lim_{\alpha\to 1} F(N)/(1 - \alpha^N) = r + 2(\rho/N).$$

5. APPROXIMATE SOLUTION

Obtaining a simple easy-to-use formula for $x_N{}^*$ and $F(N)$ is difficult, because of the discrete nature of the problem and, particularly, because the discreteness of $x_N{}^*$ and $l(k)$. However, we can obtain such a formula if we minimize (8a) subject to (8b), thus treating $l(k)$ as a continuous variable. At the end of this section, we compare the approximate and the exact solutions.

We first construct the Lagrangian

$$L = \sum_{k=1}^{N} l(k)\alpha^{k-1} - \lambda(1 - \sum_{k=1}^{N} (½)^{l(k)}).$$

Differentiating with respect to $l(k)$, we obtain

$$\alpha^{k-1} - \lambda(\ln 2)(½)^{l(k)} = 0,$$

or

$$l(k) = \log_2(\lambda \ln 2) - (k-1)\log_2\alpha. \tag{13}$$

From (8b),

$$1 = \sum_{k=1}^{N} (½)^{l(k)} = \sum_{k=1}^{N} \alpha^{k-1}(\lambda \ln 2)^{-1}$$
$$= (1-\alpha^N)(\lambda(1-\alpha)\ln 2)^{-1}, \quad \text{or}$$
$$\lambda \ln 2 = (1-\alpha^N)/(1-\alpha). \text{ Combining this expression with (13) gives} \tag{14}$$
$$l(k) = \log_2[(1-\alpha^N)/(1-\alpha)] - (k-1)\log_2\alpha$$
$$= \log_2[(1-\alpha^N)/(1-\alpha)\alpha^{k-1}].$$

Substituting $l(k)$ from (14) into (7), we have

$$\sum_{k=1}^{x_N^*} (½)^{l(k)} = [(1-\alpha)/(1-\alpha^N)] \sum_{k=1}^{x_N^*} \alpha^{k-1}$$
$$= [(1-\alpha^{x_N^*})/(1-\alpha^N)] = ½, \quad \text{or} \tag{15}$$
$$x_N^* = \log_\alpha((1+\alpha^N)/2).$$

Thus, if x_N^* in (15) is between $k - ½$ and $k + ½$ for an integer k, the approximate strategy is k.

To find the approximate value of $F(N)$, we introduce $l(k)$ from (14) into (7) giving

$$\hat{F}(N) = (1-\alpha) \sum_{k=1}^{N} \log_2((1-\alpha^N)/(1-\alpha)\alpha^{k-1})\alpha^{k-1}, \quad \text{or}$$
$$\hat{F}(N) = (1-\alpha^N)\log_2((\alpha-\alpha^{N+1})/(1-\alpha)) \tag{16}$$
$$- ((1-\alpha^{N+1})/(1-\alpha) - (N+1)\alpha^N)\log_2\alpha.$$

It is intuitively clear that x_N^* is nondecreasing in α, and, thus, there exist quantities $g_k(N)$ such that $x_N^* = k$ is optimal if, and only if, $g_{k-1}(N) \leq \alpha \leq g_k(N)$. The graphs of $g_k(N)$ can be depicted and used to find the optimal strategy for any values of N and α in practical problems (cf. Section 7). To find the approximate values for the functions $g_k(N)$, we look for the value of α which gives $x_N^* = k + ½$. Thus,

$$k + ½ = \log_\alpha((1+\alpha^N)/2) \quad \text{or}$$
$$2\alpha^{k+½} - \alpha^N = 1. \tag{17}$$

6. NUMERICAL RESULTS

Tables I and II compare the approximate solution with the exact one. We first compare $F(N)$ with its approximation $\hat{F}(N)$ and with the correct cost obtained when using the approximate solution $\hat{x}_N{}^*$, denoted by $FF(N)$. Table I gives some typical values. The approximate value $\hat{g}_k(N)$ is the solution of (17). Table II compares $g_k(N)$ and $\hat{g}_k(N)$ for $N = 100$. Since $100 = 2^6 + 36$, by Theorem 7 $x_{100}^* \leq 36$ and $g_k(N)$ exists for $k \leq 35$. Note that the difference is always smaller than 0.5%. This quality of the

TABLE I

APPROXIMATE AND EXACT SOLUTIONS

N	$F(N)$	x_N^*	$\hat{F}(N)$	$\hat{x}_N^*$	$FF(N)$	FF/F
$\alpha = 0.6$						
5	2.022	1	2.000	1	2.022	1.000
10	2.435	1	2.374	1	2.435	1.000
100	2.500	1	2.427	1	2.500	1.000
500	2.500	1	2.427	1	2.500	1.000
1000	2.500	1	2.427	1	2.500	1.000
$\alpha = 0.8$						
5	2.274	2	2.252	2	2.274	1.000
10	3.090	3	3.059	3	3.090	1.000
100	3.639	3	3.610	3	3.639	1.000
500	3.639	3	3.610	3	3.639	1.000
1000	3.639	3	3.610	3	3.639	1.000
$\alpha = 0.9$						
5	2.338	2	2.306	2	2.338	1.000
10	3.281	4	3.258	4	3.298	1.005
100	4.725	7	4.690	7	4.725	1.000
500	4.725	7	4.690	7	4.725	1.000
1000	4.725	7	4.690	7	4.725	1.000
$\alpha = 0.99$						
5	2.394	2	2.322	2	2.394	1.000
10	3.388	4	3.321	5	3.394	1.002
100	6.613	35	6.585	38	6.634	1.003
500	8.048	68	8.022	68	8.073	1.003
1000	8.104	69	8.079	69	8.125	1.003
$\alpha = 0.9999$						
5	2.399	2	2.322	2	2.399	1.000
10	3.399	4	3.322	5	3.399	1.000
100	6.710	36	6.643	49	6.718	1.001
500	8.970	244	8.951	219	8.998	1.003
1000	9.944	394	9.907	380	9.971	1.003

approximation was obtained for the other values of N which have been tested.

7. GRAPHIC PRESENTATION

Figure 5 graphically presents the solution for $N \leq 40$ and $0 < \alpha < 1$ by the graphs of $g_k(N)$. For example if $\alpha = 0.95$ then $-(\ln \alpha)^{-1} = 19.50$ and $x_{32}^* = 10$. In fact we learn from Figure 5 that $x_{32}^* = 10$ for $R(12) < \alpha < R(15)$.

8. SUMMARY AND CONCLUDING REMARKS

In this paper we considered a special case of the following problem: What is the strategy that minimizes the expected number of questions (of a particular type) required to identify an integer between 1 and N selected randomly with a given probability distribution? The questions are of the following form: Is the number greater than or equal to x? We considered the special case of the geometric/exponential probability distribution which has many applications.

TABLE II

EXACT AND APPROXIMATE VALUES OF THE g FUNCTION

k	g_k (100)	$\hat{g}_k$ (100)	$\hat{g}/g$
3	0.8192	0.8213	1.0026
4	0.8567	0.8582	1.0018
5	0.8812	0.8826	1.0016
6	0.8987	0.8999	1.0012
7	0.9116	0.9127	1.0012
8	0.9216	0.9227	1.0011
9	0.9296	0.9306	1.0011
10	0.9362	0.9371	1.0010
11	0.9416	0.9425	1.0010
12	0.9463	0.9471	1.0008
13	0.9503	0.9510	1.0007
14	0.9537	0.9543	1.0007
15	0.9568	0.9573	1.0005
16	0.9602	0.9599	0.9996
17	0.9627	0.9624	0.9997
18	0.9649	0.9646	0.9997
19	0.9670	0.9668	0.9998
20	0.9687	0.9687	1.0000
21	0.9704	0.9705	1.0000
22	0.9720	0.9721	1.0001
23	0.9735	0.9737	1.0002
24	0.9748	0.9751	1.0003
25	0.9760	0.9766	1.0006
26	0.9771	0.9779	1.0008
27	0.9782	0.9792	1.0010
28	0.9791	0.9804	1.0013
29	0.9801	0.9816	1.0015
30	0.9811	0.9827	1.0016
31	0.9820	0.9838	1.0018
32	0.9886	0.9848	0.9961
33	0.9893	0.9858	0.9965
34	0.9896	0.9868	0.9972
35	0.9900	0.9878	0.9977

We analyzed the problem in two stages. First, we stated and proved Theorems 1–7 which characterize some of the more important properties of the solution. The interested reader can, however, discover some additional properties by inspecting Figure 5.

In the second stage, we developed formulas for approximating the

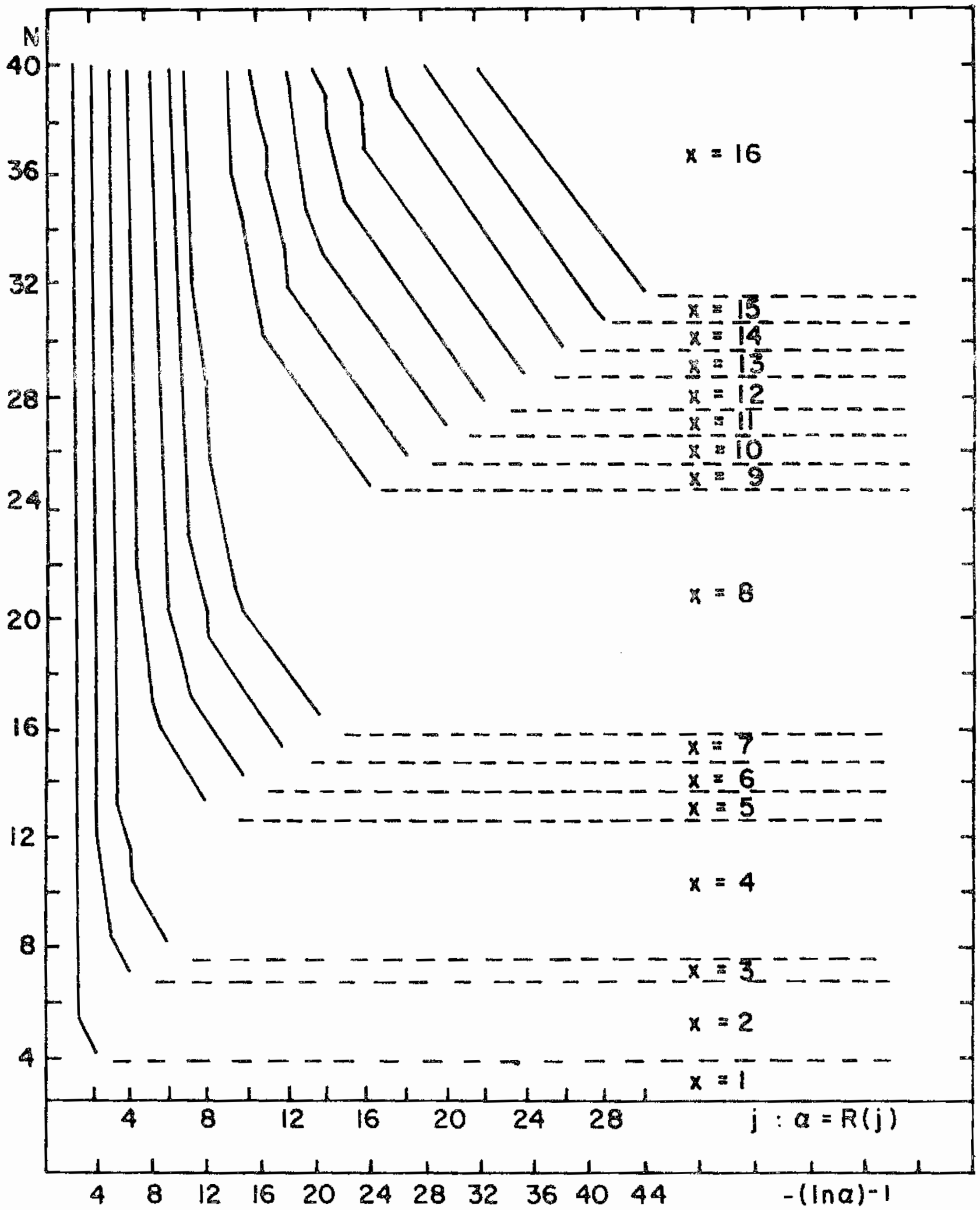

Figure 5

solutions. The approximation technique can also be applied to the following extension of the problem. Suppose that the cost of the search is equal to the expected value of some nonlinear function C of the number of inspections. There does not seem to be any direct way to modify the dynamic programming approach to model to this case. However, the modification of the tree formulation is straightforward, and we wish to minimize $\sum_{k=1}^{n} C(l(k))\alpha^{k-1}$ subject to (8b) and (8c). The monotonicity of (8c) assures that the optimal tree is nondecreasing and, therefore, an approximation to the solution can be found by dropping the integrality constraints. In the general case, however, we do not obtain a simple formula such as (15), but a set of equations whose variables are the levels of the optimal tree.

ACKNOWLEDGMENT

The author wishes to convey his gratitude to the referees for helping to improve this paper.

REFERENCES

CAMERON, S. H., AND S. G. NARAYANAMURTHY. 1964. A Search Problem. *Opns. Res.* **12,** 623–629.

GAL, S. 1974. A Discrete Search Game. *SIAM J. Appl. Math.* **27,** 641–648.

GILBERT, E. N., AND E. F. MOORE. 1959. Variable Length Binary Encodings. *Bell System Tech. J.* **38,** 933–968.

HASSIN, R., AND N. MEGIDDO. 1980. An Optimal Algorithm for Finding all the Jumps of a Monotone Step-Function, to appear in the *Journal of Algorithms.*

HU, T. C., AND A. C. TUCKER. 1971. Optimal Computer-Search Trees and Variable-Length Alphabetical Codes. *SIAM J. Appl. Math.* **21,** 514–532.

KNUTH, D. E. 1973. The Art of Computer Programming. In *Vol. 3. Sorting and Searching.* Addison-Wesley, Reading, Mass.

MORRIS, R. 1969. Some Theorems on Sorting. *SIAM J. Appl. Math.* **17,** 1–6.

MURAKAMI, S. 1971. A Dichotomous Search. *J. Opns. Res. Soc. Jpn.* **14,** 127–142.

MURAKAMI, S. 1976. A Dichotomous Search with Travel Cost. *J. Opns. Res. Soc. Jpn.* **19,** 245–254.