Discrete Applied Mathematics 46 (1993) 221–234
North-Holland

Monotonicity and efficient computation of optimal dichotomous search

R. Hassin and M. Henig

Tel-Aviv University, Tel-Aviv 69978, Israel

Received 19 December 1989
Revised 3 January 1991

Abstract

We consider the problem of designing an efficient dichotomous search in order to locate an object which lies on a given interval. A query at a point of the interval reveals whether the object is to its "left" or to its "right". By successively placing queries at points of the interval it narrows down until the searcher can identify a unit interval containing the object. The objective is to minimize the expected cost of the search. We analyze the problem for a wide range of cost structures, generalizing several known results. In particular we extend a monotonicity theorem of Knuth showing that it also holds under weaker assumptions. Consequently, the computation effort needed to solve the problem is reduced.

Keywords. Dichotomous search, dynamic programming, monotone policy.

1. Introduction

We consider the problem of designing an efficient dichotomous search in order to locate an object which is known to lie on a given interval, $\{1, \ldots, N\}$, called *the interval of uncertainty*. We denote the (a priori) probability that the object lies in i, $i \in \{1, \ldots, N\}$, by p_i. A query at k reveals whether the object is in $\{1, \ldots, k\}$ or in $\{k+1, \ldots, N\}$. We denote by Problem (m, n) the instance where the interval of uncertainty is $\{\min(m, n), \ldots, \max(m, n)\}$ with the following convention: $m < n$ implies that the most recent query was placed at $m-1$, while $m > n$ means it was placed at n. By successively placing a query at a point of the interval of uncertainty this interval narrows down until the searcher can identify the object's location. The objective is to minimize the expected cost of the search.

Correspondence to: Professor R. Hassin, Department of Statistics and Operations Research, Tel Aviv University. Tel Aviv 69978, Israel.

This problem has been well analyzed when the only cost involved in the search process is a fixed cost per query. Gilbert and Moore [8] proposed an algorithm requiring $O(N^3)$ computational steps. This algorithm was later improved by Knuth [19] reducing its complexity to $O(N^2)$. Algorithms of time $O(N \log N)$ were described by Hu and Tucker [16] and Garsia and Wachs [7]. Both algorithms are significantly more complex than their predecessors and this is probably one of the reasons why they were not extended to more general problems.

In this paper we generalize and improve several search models discussed in the literature. In Section 2 we will present some preliminaries and a theorem which constitutes the main result of this paper. The discussion about the models extended by this theorem is deferred to Section 3. The proof of the theorem will be given in the final section.

2. Properties of the solution

We consider two types of costs involved in the search:

- $D^l(m, n, k)$ for placing the lth query at k in Problem (m, n);
- C_{li} if the object is found in i after l queries.

The problem can be solved by applying the following recursive equations: Denote $p_{ij} = p_i + \cdots + p_j$. For $l = 0, \ldots, N-1$, $m = 1, \ldots, N$

$$F^l(m, m) = C_{lm}, \tag{1a}$$

for $m < n$ and $l = 1, \ldots, N-(n-m)$

$$F^{l-1}(m, n) = \min_{m \le k < n} \{D^l(m, n, k) + \frac{p_{mk}}{p_{mn}} F^l(k, m) + \frac{p_{k+1,n}}{p_{mn}} F^l(k+1, n)\}, \tag{1b}$$

and for $m > n$ and $l = 1, \ldots, N-(m-n)$

$$F^{l-1}(m, n) = \min_{n \le k < m} \{D^l(m, n, k) + \frac{p_{nk}}{p_{nm}} F^l(k, n) + \frac{p_{k+1,m}}{p_{nm}} F^l(k+1, m)\}. \tag{1c}$$

$F^l(m, n)$ is the minimum expected cost involved with locating the object in Problem (m, n) when l queries have already been placed. Let k be a minimizing value for the right-hand side of (1b) or (1c), then we say that F is *attained* at k. The minimum total expected cost of the search is $F^0(1, N)$.

The complexity of the algorithm is $O(N^4)$. It can be reduced to $O(N^3)$ by applying ideas from Knuth [19] in cases where the following "monotonicity" property holds:

Property 2.1. *If $F^l(m, n-1)$ is attained at k', then for some $k \geq k'$, $F^l(m, n)$ is attained at k.*

Theorem 2.2 (Knuth [19]). *Property* 2.1 *holds when C is linear in l and constant in i, and D is constant.*

To prove Theorem 2.2 Knuth first proved that the following property holds when C is linear:

Property 2.3. *If $p_N = 0$, the optimal policy to Problem $(1, N)$ is as for Problem $(1, N-1)$, except for the case where the remaining interval is $\{N-1, N\}$ in which an additional query is needed.*

Property 2.3 does not necessarily hold for nonlinear costs even when C is convex and D is constant, as shown by the following example:

Example 2.4. Let $p_1 = 0.5$, $p_2 = p_3 = 0.25$, and $p_4 = 0$. Let $C_{1i} = C_{2i} = 1$, $C_{3i} = 10$ for every i. The optimal policy for Problem (1, 3) has $k = 1$. For Problem (1, 4), $k = 2$.

We show next that Property 2.1 does not necessarily hold even when D is constant.

Example 2.5. Let C_{li} be 1, 4, 5 for $l = 1, 2, 3$ and every i, and $p_i = 0.2, 0.3, 0.4, 0.1$ for $i = 1, \ldots, 4$, respectively. Let $D \equiv 1$. The optimal policy for Problem (1, 3) starts with a query at $k = 2$ while for Problem (1, 4) one starts with $k = 1$ and if the object is in $\{2, \ldots, 4\}$ the resulting Problem (2, 4) is solved by a first query in $k = 3$.

The case of a uniform distribution and convex costs is easy as seen from the next result which is similar to a theorem by Markowsky [21] for nonalphabetic trees:

Theorem 2.6. *Suppose that C is convex in l and constant in i, and that p is constant, then the optimal policy is as for linear C.*

Proof. See Markowsky [21]. □

Our main result is a proof that Property 2.1 holds for the general model introduced above, under certain assumptions on the costs. As seen from Example 2.4 a different approach from the one used by Knuth is needed. The approach we use is of proving that under certain assumptions a function associated with F is submodular. Related principles for general dynamic programming are described by Topkis [25] and Heyman and Sobel [12], and applied to a special case of our problem by Yao [28] (see also Yao [29]).

We next study some properties of the function $G^l(m, n) = p_{mn} F^l(m, n)$, if $m \le n$ and $G^l(m, n) = p_{nm} F^l(m, n)$, if $m > n$. As already noticed by Morris [22] for a very restricted case, this function possesses some important properties that F does not have. Let $d^l(m, n, k)$ be equal to $D^l(m, n, k)p_{mn}$ for $m \le n$, and to $D^l(m, n, k)p_{nm}$ for $n < m$. Let $c_{lm} = C_{lm}p_m$. Then $G^l(m, n)$ is defined as follows: For $l = 0, \dots, N - 1$, $m = 1, \dots, N$

$$G^l(m,m) = c_{lm}, \tag{2a}$$

for $m < n$ and $l = 1, \dots, N - (n - m)$

$$G^{l-1}(m, n) = \min_{m \le k < n} \{d^l(m, n, k) + G^l(k, m) + G^l(k + 1, n)\}, \tag{2b}$$

and for $m > n$ and $l = 1, \dots, N - (m - n)$

$$G^{l-1}(m, n) = \min_{n \le k < m} \{d^l(m, n, k) + G^l(k, n) + G^l(k + 1, m)\}. \tag{2c}$$

Obviously, $G^l(m, n)$ is attained by the same values as $F^l(m, n)$.

The proof to the following theorem will be given in Section 4:

Theorem 2.7. For $l = 0, \dots, N - 1$, *let* d^l *be defined over the lattice*

$$\mathscr{L} = \{(m, n, k): 1 \le m, n \le N, \min\{m, n\} \le k < \max\{m, n\}\}.$$

Let c_{ln} *be defined over* $\{(l, n): l = 1, \dots, N - 1, n = 1, \dots, N\}$. *We make the following assumptions*:

(A1) d^l *is submodular for every fixed* l, *i.e., for every pair of points* $(m_i, n_i, k_i) \in \mathscr{L}$, $i = 1, 2$

$$\begin{aligned} &d^l(m_1, n_1, k_1) + d^l(m_2, n_2, k_2) \\ &\quad \ge d^l(\min\{m_1, m_2\}, \min\{n_1, n_2\}, \min\{k_1, k_2\}) \\ &\qquad + d^l(\max\{m_1, m_2\}, \max\{n_1, n_2\}, \max\{k_1, k_2\}). \end{aligned}$$

(A2.1)

$$\begin{aligned} &d^l(n - 1, n, n - 1) + d^l(n, n + 1, n) \\ &\quad \le \min\{d^l(n - 1, n + 1, n - 1) + d^{l+1}(n, n + 1, n), \\ &\qquad\qquad d^l(n - 1, n + 1, n) + d^{l+1}(n, n - 1, n - 1)\}. \end{aligned}$$

(A2.2)

$$\begin{aligned} &d^l(n, n - 1, n - 1) + d^l(n + 1, n, n) \\ &\quad \le \min\{d^l(n + 1, n - 1, n - 1) + d^{l+1}(n + 1, n, n), \\ &\qquad\qquad d^l(n + 1, n - 1, n) + d^{l+1}(n - 1, n, n - 1)\}. \end{aligned}$$

(A3) *c is convex in l for every fixed n, i.e.,*

$$2c_{ln} \le c_{l+1,n} + c_{l-1,n}, \quad l = 2, \ldots, N-2.$$

(A4) *c is nonnegative and nondecreasing in l for every fixed n.*
Under Assumptions (A1)–(A4), G^l *is submodular, i.e.,*

$$G^l(m,n) + G^l(m+a, n+b) \le G^l(m, n+b) + G^l(m+a, n) \tag{3}$$

for $1 \le m \le m+a \le N$, $1 \le n \le n+b \le N$, *and* $2 \le l \le N - \max\{|m-n-b|, |m-n+a|\} - 1$.

Remark 2.8. Conditions (A2) connect the search costs incurred in periods l and $l+1$, and seem to lack the intuitive reasoning associated with the other conditions. In the next section we will show that they hold in several cases of interest at least whenever we can assume that D^l, and therefore also d^l, is nondecreasing in l. In this case, it is sufficient to demonstrate that
(A5.1)

$$\begin{aligned} &d^l(n-1, n, n-1) + d^l(n, n+1, n) \\ &\quad \le \min\{d^l(n-1, n+1, n-1) + d^l(n, n+1, n), \\ &\qquad\qquad d^l(n-1, n+1, n) + d^l(n, n-1, n-1)\}, \end{aligned}$$

and
(A5.2)

$$\begin{aligned} &d^l(n, n-1, n-1) + d^l(n+1, n, n) \\ &\quad \le \min\{d^l(n+1, n-1, n-1) + d^l(n+1, n, n), \\ &\qquad\qquad d^l(n+1, n-1, n) + d^l(n-1, n, n-1)\}. \end{aligned}$$

Theorem 2.9 (Monotonicity Theorem). *Property* 2.1 *holds under the conditions of Theorem* 2.7.

Proof. We prove the claim for $m < n$. The proof for $m > n$ is similar. Suppose that $G^l(m, n-1)$ is attained at k'. By (2b)

$$G^l(m,n) \le d^{l+1}(m,n,k') + G^{l+1}(k',m) + G^{l+1}(k'+1, n). \tag{4}$$

Adding and subtracting identical terms we obtain

$$\begin{aligned} G^l(m,n) \le {} & [d^{l+1}(m, n-1, k') + G^{l+1}(k', m) + G^{l+1}(k'+1, n-1)] \\ & + [G^{l+1}(k'+1, n) - G^{l+1}(k'+1, n-1)] \\ & + [d^{l+1}(m,n,k') - d^{l+1}(m, n-1, k')]. \end{aligned}$$

Using (2b) and the definition of k' for the first expression, (3) for the second, and (A1) for the third, we obtain for every $k \in \{m, \dots, k'\}$

$$\begin{aligned} G^l(m,n) \le {} & [d^{l+1}(m,n-1,k) + G^{l+1}(k,m) + G^{l+1}(k+1,n-1)] \\ & + [G^{l+1}(k+1,n) - G^{l+1}(k+1,n-1)] \\ & + [d^{l+1}(m,n,k) - d^{l+1}(m,n-1,k)] \\ = {} & d^{l+1}(m,n,k) + G^{l+1}(k,m) + G^{l+1}(k+1,n). \end{aligned}$$

If strict inequality holds for every $k \in \{m, \dots, k'\}$ then $G^l(m,n)$ must be attained at some $k > k'$. If equality holds for some $k \le k'$ then equality must hold in (4) and $G^l(m,n)$ is attained at k'. □

Corollary 2.10. *For fixed $l \ge 1$ there exist values $k(m,n)$, $m, n = 1, \dots, N$ such that $G^l(m,n)$ is attained at $k(m,n)$ and*

$$k(m,n-1) \le k(m,n) \le k(m+1,n).$$

Proof. Follows from Theorem 2.9 by left-right symmetry. □

For completeness we will describe how Corollary 2.10 serves to reduce the amount of computations from $O(N^4)$ to $O(N^3)$, using Knuth's observation. The corollary makes it possible to restrict the search for $k(m,n)$ to the interval $\{k(m,n-1), \dots, k(m+1,n)\}$. Thus, for l fixed the minimum in (2) is computed in order of increasing $|n-m|$, and the effort is proportional to

$$\begin{aligned} & \sum_{m,n} (k(m+1,n) - k(m,n-1) + 1) \\ & \quad = \sum_m \sum_r [k(m+1, m+1+r) - k(m, m+r) + 1]. \end{aligned}$$

This is a telescopic series and for any fixed m all terms except for two can be cancelled to obtain an overall complexity of $O(N^2)$. Summing for l an $O(N^3)$ time is obtained. In those applications where C is linear, the state variable l is redundant and the complexity is reduced from $O(N^3)$ to $O(N^2)$.

3. Applications

Many of the interesting applications to the search problem discussed in here are in computer science where the search policy defines a binary tree associated with a code. We will try to present more potential applications from other fields, and refer to the computer science literature when a related problem is discussed there. We will mention each case separately, but the conclusions are also valid for combinations of the assumptions.

Restricted number of queries. In this case there is an upper bound, L, on the number of queries allowed in the search process. This bound may result from budget limitations or from the fact that the object becomes useless after a while. This is a special case of a convex function where C_{ln}, and therefore also c_{ln}, is "infinite" for $l > L$. Knuth's algorithm was extended to this case by Itai [12], Wessner [27], and Yao [28]. (Similar results for the related problem where the intervals can be re-ordered by the searcher are also given by Garey [6], Hu and Tan [15] and Larmore [20].)

Travel costs (see, Murakami [24], Hu [14], Hu and Wachs [17], Hassin and Hotovly [11]). In many cases of interest the searcher actually has to move to the point in order to place a query. For example, Murakami [24] models in this way the search for the recrystallization temperature of a metal, Hu [14] and Hu and Wachs [17] use such a model for search for a record on a tape. Thus, in addition to search costs which depend on the number of queries and tend to favor queries placed about the median of the distribution, the travel costs favor a policy that places queries close to the searcher's present location. Let τ_i^+ (τ_i^-) denote a nonnegative cost of traveling from $i-1$ to i (from i to $i-1$). Let D_{ij} denote the cost associated with the travel from i to j. Then, $D_{ii}=0$, for $i<j$, $D_{ij}=\sum_{k=i+1}^{j}\tau_k^+$, and for $i>j$, $D_{ij}=\sum_{k=j+1}^{i}\tau_k^-$. Suppose that the object is known to lie in $\{m,\ldots,n\}$ for $m<n$. If the last query was at $m-1$ the cost to place the next one at k is $D_{m-1,k}$. If the last query was in n then the travel cost is D_{nk}. Defining $D^l(m,n,k)$ as the above travel cost function and multiplying by p_{mn} results in the associated function $d^l(m,n,k)$ as in (2). Unfortunately, this is not a submodular function and (A1) is violated. To overcome this difficulty we reformulate the problem. We observe that to locate an object lying at i one has to place queries both at $i-1$ and i. Suppose $m<n$. Problem (m,n) assumes that the answers to queries at $m-1$ and n are known. Any solution to the problem includes the unavoidable travel cost associated with the travel from the present location $m-1$ to $i-1$, if $i\neq m$, and to i, if $i\neq n$. This cost is independent of the search policy. We now define the problem excluding this constant from F. $D(m,n,k)$ will now describe the *expected additional* travel cost caused by placing the next query in k. Suppose first that $m<n$. Such a cost is incurred only if $k>i$, where the object is in i, and is composed of the expected extra travel including the return trip to the new far end of the interval containing the object. Thus, $D(m,n,k)=p_m/p_{mn}(D_{mk}+D_{km})+\sum_{i=m+1}^{k}p_i/p_{mn}\cdot(D_{ik}+D_{k,i-1})$. Applying the transformation to (2) we obtain for $m<n$, $d(m,n,k)=p_m(D_{mk}+D_{km})+\sum_{i=m+1}^{k}p_i(D_{ik}+D_{k,i-1})$, and for $m>n$, $d(m,n,k)=p_m(D_{m-1,k}+D_{k,m-1})+\sum_{i=k+1}^{m-1}p_i(D_{i-1,k}+D_{ki})$.

Lemma 3.1. (A1) *and* (A5) *hold for the reformulated model with travel costs.*

Proof. (A5) follows since the left-hand sides of (A5) are zero, while the right-hand sides are nonnegative. Noting that $d(m,n,k)$ is independent of n, submodularity of d amounts to

$$d(m,n,k)-d(m+1,n,k)\leq d(m,n,k+1)-d(m+1,n,k+1),$$

for $m < n$ (and similarly for $m > n$). Substituting d and cancelling terms the inequality reduces to $d_{mk} + d_{km} \leq d_{m,k+1} + d_{k+1,m}$. This inequality is satisfied since $\tau_{k+1}^{+} \geq 0$ implies that $d_{mk} \leq d_{m,k+1}$, while $\tau_k^{-} \geq 0$ implies that $d_{km} \leq d_{k+1,m}$. $\square$

Deviation dependent costs. In many applications the cost of a query depends on whether the "object" lies to its "right" or "left". For example, a "query" above the searched value may destroy the machinery used in an experiment (Cameron and Narayanamurthy [3], Murakami [23], Hinderer [13]). In other cases the cost is proportional to the sum of absolute deviations of the query points from the object's location (Baston and Bostock [2], Alpern [1]). More complex cost functions may be involved in determining the optimal dosage of some medicine (see, Eichhorn and Zacks [5], Eichhorn [4]). Hinderer [13] proved the monotonicity theorem for the direction-dependent costs. We now show its validation for the more general case.

Let $R(i, k) \geq 0$ be the cost associated with a query placed in k while the object is located in i. Then $D^l(m, n, k) = \sum_{i=m,\ldots,n} R(i, k)p_i/p_{mn}$ for $m \leq n$, (and similarly for $m > n$). Thus $d^l(m, n, k) = \sum_{i=m,\ldots,n} r(i, k)$, where $r(i, k) = R(i, k)p_i$. We make the reasonable assumption that for i fixed $R(i, k)$ (and therefore also $r(i, k)$) is unimodal in k with a minimum $R(i, i) = 0$. This corresponds to a monotonicity of the costs with respect to the deviation of the query from the actual location of the object.

Lemma 3.2. *In the model with deviation dependent costs, d^l satisfies* (A1).

Proof. Let $(m_i, n_i, k_i) \in \mathscr{L}$, $i = 1, 2$ be given. Let $d \equiv d^l$, and assume, without loss of generality, that $k_1 \geq k_2$. By the left-right symmetry of the assumptions we also assume, without loss of generality, that $n_i \geq m_i$, $i = 1, 2$. Assumption (A1) reduces to

$$\sum_{i=\max\{m_1, m_2\}}^{\max\{n_1, n_2\}} r(i, k_1) - \sum_{i=m_1}^{n_1} r(i, k_1) \leq \sum_{i=m_2}^{n_2} r(i, k_2) - \sum_{i=\min\{m_1, m_2\}}^{\min\{n_1, n_2\}} r(i, k_2). \tag{5}$$

Suppose first that $m_1 \geq m_2$ and $n_1 \geq n_2$. (5) holds since both sides are equal to zero. Suppose now that $m_1 \geq m_2$ while $n_1 < n_2$. (5) becomes

$$\sum_{i=n_1+1}^{n_2} r(i, k_1) \leq \sum_{i=n_1+1}^{n_2} r(i, k_2).$$

(5) holds also in this case since the unimodality assumption of R implies $r(i, k_1) \leq r(i, k_2)$ for every $i \in \{n_1 + 1, \ldots, n_2\}$.

Suppose next that $m_1 < m_2$. (5) becomes

$$\sum_{i=m_2}^{\max\{n_1, n_2\}} r(i, k_1) - \sum_{i=m_1}^{n_1} r(i, k_1) \leq \sum_{i=m_2}^{n_2} r(i, k_2) - \sum_{i=m_1}^{\min\{n_1, n_2\}} r(i, k_2).$$

If $n_1 \geq n_2$ it becomes

$$-\sum_{i=m_1}^{m_2-1} r(i, k_1) \leq -\sum_{i=m_1}^{m_2-1} r(i, k_2).$$

In this case we have $k_1 \geq k_2 \geq m_2 > m_1$ and the inequality holds by the unimodality assumption of R. If $n_1 < n_2$ the inequality becomes

$$\sum_{i=m_2}^{n_2} r(i, k_1) - \sum_{i=m_1}^{n_1} r(i, k_1) \leq \sum_{i=m_2}^{n_2} r(i, k_2) - \sum_{i=m_1}^{n_1} r(i, k_2),$$

or

$$\sum_{i=n_1+1}^{n_2} r(i, k_1) - \sum_{i=m_1}^{m_2-1} r(i, k_1) \leq \sum_{i=n_1+1}^{n_2} r(i, k_2) - \sum_{i=m_1}^{m_2-1} r(i, k_2).$$

In this case our assumptions amount to $m_1 < m_2 \leq k_2 \leq k_1 \leq n_1 < n_2$, and it follows from the unimodality assumption that $r(i, k_1) \leq r(i, k_2)$ for $i = n_1 + 1, \ldots, n_2$ and $r(i, k_2) \leq r(i, k_1)$ for $i = m_1, \ldots, m_2 - 1$. □

Lemma 3.3. *In the model with deviation dependent costs, d^l satisfies* (A5).

Proof. In this model $d(m, n, k) = d(n, m, k)$. Hence, (A5.1) reduces to $d(n-1, n, n-1) \leq d(n-1, n+1, n-1)$, and $d(n, n+1, n) \leq d(n-1, n+1, n)$. These inequalities follow since for $m < n$, $d(m, n, k)$ is nondecreasing in n. (A5.2) follows from similar arguments. □

Position-dependent query costs (Wachs [26]). Here the cost of placing a query varies according to the position of this query. For example, a query may require drilling in the specific position, and the terrain's hardness may be different in each point. Wachs [26] deals with such a situation, motivated by the problem arising in search on a sequential access file or tape. The monotonicity property is proved there for a restricted case.

The application of Theorem 2.7 to this case is straightforward. Let $d^l(m, n, k) = d^l(k)$. Then (A1) and (A5) hold with equality.

Geometric distribution (Hassin [10]). Suppose that $D^l(m, n, k) = D^l(m+1, n+1, k+1)$ whenever these terms are defined. This is the case in most of the above-mentioned applications, where the costs depend only on the location of the query relative to the current interval of uncertainty. Assume further that $\{p_i\}$ is a truncated geometric distribution, i.e., p_i is proportional to p^i. (Hassin [10] describes several applications where this assumption holds.) Then, Problem $(m, m+j)$ is independent of m, and

$$k(m, m+j) \leq k(m, m+j+1) = 1 + k(m-1, m+j) \leq 1 + k(m, m+j),$$

where the inequalities follow from Corollary 2.10 and the equality is immediate from the above assumptions. Denoting by $k(j)$ a value where Problem $(m, m+j)$ is attained, we conclude that

$$k(j+1) \in \{k(j), k(j)+1\}.$$

The result is an $O(n^2)$ algorithm for (1), which becomes linear if D^l is independent of l. Hassin [10] proved an analogous result for a restricted case.

Parallel search (Gotlieb [9]). In this case there are L searchers and each of them may place a query at each stage of the search process. These queries are evaluated simultaneously and as a result a new interval of uncertainty is obtained and a new set of up to L queries is placed. We assume that each stage takes one unit of time and that the cost associated with locating the object is C per stage. One possible way to solve the problem is by defining $F(m, n)$ to be the cost of the (m, n) problem and then

$$F(m, m) = 0,$$

$$F(m, n) = C, \quad m + L \geq n,$$

and for $m < n - L$

$$F(m, n) = \min_{k_0 = m \leq k_1 \leq \cdots < k_{L+1} = n} \left\{ C + \sum_{i=0}^{L} \frac{p_{k_i+1, k_{i+1}}}{p_{mn}} F(k_i + 1, k_{i+1}) \right\}.$$

It is more efficient however to compute the optimal assignment of searchers in a given stage sequentially, deciding at each stage on the assignment of the next leftmost searcher. Let l be an index denoting the number of searchers that were already assigned. Let $F^l(m, n)$ $(0 \leq l \leq L)$ denote the expected cost needed when l queries were already placed outside the interval of uncertainty $\{m, \ldots, n\}$. Then $F^l(m, m) = 0$, and for $m < n$

$$F^{l-1}(m, n) = \delta_l + \min_{m \leq k < n} \left\{ \frac{p_{mk}}{p_{mn}} F^{g(l)}(m, k) + \frac{p_{k+1,n}}{p_{mn}} F^{h(l)}(k+1, n) \right\}, \quad (6)$$

where $\delta_l = C$ if $l = 1 \pmod L$, and $\delta_l = 0$ otherwise, $g(l) = \lceil l/L \rceil L$, and $h(l) = l$.

As stated, the complexity of the algorithm is $O(n^4)$. However, realizing that $F^l = F^{l \pmod L}$, it is sufficient to solve for $l = 1, \ldots, L$, and the complexity is $O(n^3 L)$ (where it is assumed that $L < n$). One can follow the inductive proof of Theorem 2.7 to see that the theorem holds also for this case (with $d^l(m, n, k) = \delta_l p_{mn}$ being modular, and $c \equiv 0$). We did not present the theorem in a more general way because it requires a substantial complication of the notation. The complexity reduces consequently from $O(n^3 L)$ to $O(n^2 L)$. We note that the decision in (6) is where to locate the next leftmost query in the present stage. It can be replaced by a decision of where to locate the middle one among the remaining queries in that stage, substituting $g(l)$ and $h(l)$ by $\lfloor (g(l) + l)/2 \rfloor$ and $\lceil (g(l) + l)/2 \rceil$, respectively. (To see this, note that the remaining number of queries in the present stage is $g(l) - l$, and dividing this number equally for the intervals on the left and right of k leaves for each interval the above-mentioned number of additional queries in the present stage.) Consequently, the complexity is reduced to $O(n^2 \log L)$ (see Gotlieb [9] for details).

4. Proof of the main theorem

Notice that for a given $l \in \{0, \ldots, N-1\}$, G^l is defined over a lattice $\mathscr{L}^1 = \{(m, n): m, n \in \{1, \ldots, N\}, |m - n| \leq N - l\}$.

Lemma 4.1. *Let $l \in \{0, \ldots, N-1\}$. Suppose that for all m, n such that $(m+1, n)$ and $(m, n+1)$ are in $\mathscr{L}^l$*

$$G^l(m+1, n+1) - G^l(m+1, n) \leq G^l(m, n+1) - G^l(m, n).$$

Then for all $a, b \geq 0$ such that $(m+a, n)$ and $(m, n+b)$ are in $\mathscr{L}^l$,

$$G^l(m+a, n+b) - G^l(m+a, n) \leq G^l(m, n+b) - G^l(m, n).$$

Proof. By the assumption $G^l(m+1, n+j+1) - G^l(m+1, n+j) \leq G^l(m, n+j+1) - G^l(m, n+j)$ for $j = 0, \ldots, b-1$. By summing both sides of the inequality over all values of $j = 0, \ldots, b-1$ and after cancelling identical terms we get $G^l(m+1, n+b) - G^l(m+1, n) \leq G^l(m, n+b) - G^l(m, n)$. Thus, $G^l(m+i, n+b) - G^l(m+i, n) \leq G^l(m+i-1, n+b) - G^l(m+i-1, n)$ for $i = 1, \ldots, a$. By summing both sides of the inequality over $i = 1, \ldots, a$, and after cancelling identical terms we get $G^l(m+a, n+b) - G^l(m+a, n) \leq G^l(m, n+b) - G^l(m, n)$. $\square$

Proof of Theorem 2.7. Noting that the claim and assumptions are symmetric with respect to m and n, it is sufficient to prove the claim for $m \leq n$. If either $a = 0$ or $b = 0$ then (3) is an identity. By Lemma 4.1 it is sufficient therefore to prove (3) for $a = b = 1$, i.e.,

$$G^l(m, n) + G^l(m+1, n+1) \leq G^l(m, n+1) + G^l(m+1, n). \tag{7}$$

The claim is trivially valid for $l = N-1$ since G^{N-1} is defined for $m = n$ (and therefore, $a = b = 0$) only. We continue with induction on l. Suppose the claim is true for some l $(3 \leq l \leq N-1)$. We distinguish three cases.

Case 1: $m \leq n-2$. Let k_1 and k_2 solve $G^{l-1}(m, n+1)$ and $G^{l-1}(m+1, n)$ respectively. We prove this case for $k_1 \geq k_2$. A similar proof can be given for $k_1 < k_2$. From this assumption it follows that $k_1 \geq m+1$ is a feasible choice for the $(m+1, n+1)$ problem, while k_2 is a feasible choice for the (m, n) problem. By (2b)

$$\begin{aligned}
&G^{l-1}(m, n) + G^{l-1}(m+1, n+1) \\
&\quad \leq d^l(m, n, k_2) + G^l(k_2, m) + G^l(k_2+1, n) + d^l(m+1, n+1, k_1) \\
&\qquad + G^l(k_1, m+1) + G^l(k_1+1, n+1) \\
&\quad \leq d^l(m+1, n, k_2) + G^l(k_2, m) + G^l(k_2+1, n) + d^l(m, n+1, k_1) \\
&\qquad + G^l(k_1, m+1) + G^l(k_1+1, n+1),
\end{aligned}$$

where the last inequality follows (for $k_1 \geq k_2$) from (A1). By the induction assumption

$$G^l(k_2+1,n) + G^l(k_1+1,n+1) \leq G^l(k_2+1,n+1) + G^l(k_1+1,n),$$

hence

$$\begin{aligned} G^{l-1}(m,n) + G^{l-1}(m+1,n+1) &\leq d^l(m,n+1,k_1) + G^l(k_1,m+1) \\ &\quad + G^l(k_1+1,n) + d^l(m+1,n,k_2) \\ &\quad + G^l(k_2,m) + G^l(k_2+1,n+1) \\ &= G^{l-1}(m,n+1) + G^{l-1}(m+1,n), \end{aligned}$$

where the equality holds since we assume $m \leq n-2$.

Case 2: $m = n-1$. By (2a), $G^{l-1}(n,n) = c_{l-1,n}$. Denote $d^l(n) = d^l(n,n+1,n)$. By (2b),

$$\begin{aligned} G^{l-1}(n,n+1) &= d^l(n) + G^l(n,n) + G^l(n+1,n+1) \\ &= d^l(n) + c_{ln} + c_{l,n+1}, \end{aligned} \tag{8}$$

$$\begin{aligned} G^{l-1}(n-1,n) &= d^l(n-1) + G^l(n-1,n-1) + G^l(n,n) \\ &= d^l(n-1) + c_{l,n-1} + c_{ln} \end{aligned} \tag{9}$$

and

$$\begin{aligned} &G^{l-1}(n-1,n+1) \\ &\quad = \min\{d^l(n-1,n+1,n-1) + G^l(n-1,n-1) + G^l(n,n+1), \\ &\qquad\quad d^l(n-1,n+1,n) + G^l(n,n-1) + G^l(n+1,n+1)\} \\ &\quad = \min\{d^l(n-1,n+1,n-1) + d^{l+1}(n) + c_{l,n-1} + c_{l+1,n} + c_{l+1,n+1}, \\ &\qquad\quad d^l(n-1,n+1,n) + d^{l+1}(n,n-1,n-1) \\ &\qquad\quad + c_{l+1,n-1} + c_{l+1,n} + c_{l,n+1}\}. \end{aligned} \tag{10}$$

Adding (8) to (9) we have

$$\begin{aligned} G^{l-1}(n-1,n) + G^{l-1}(n,n+1) &= d^l(n) + d^l(n-1) + c_{l,n-1} \\ &\quad + 2c_{ln} + c_{l,n+1} \\ &\leq d^l(n) + d^l(n-1) + c_{l,n-1} + c_{l+1,n} \\ &\quad + c_{l-1,n} + c_{l,n+1}, \end{aligned} \tag{11}$$

where the last inequality follows from (A3). Using $c_{l,n+1} \leq c_{l+1,n+1}$ (by (A4)) and (A2) we obtain that

$$\begin{aligned} G^{l-1}(n-1,n) + G^{l-1}(n,n+1) &\leq d^l(n-1,n+1,n-1) + d^{l+1}(n) \\ &\quad + c_{l,n-1} + c_{l+1,n} \\ &\quad + c_{l+1,n+1} + c_{l-1,n}. \end{aligned} \tag{12}$$

Similarly, using $c_{l,n-1} \le c_{l+1,n-1}$ (by (A4)) and (A2) we obtain that

$$\begin{aligned} G^{l-1}(n-1,n) + G^{l-1}(n,n+1) \le d^{l+1}(n,n-1,n-1) \\ + d^{l}(n-1,n+1,n) + c_{l,n+1} \\ + c_{l+1,n} + c_{l+1,n-1} + c_{l-1,n}. \end{aligned} \tag{13}$$

By (12) and (13),

$$\begin{aligned} &G^{l-1}(n-1,n) + G^{l-1}(n,n+1) \\ &\le \min\{d^{l}(n-1,n+1,n-1) + d^{l+1}(n) + c_{l,n-1} + c_{l+1,n} + c_{l+1,n+1}, \\ &\qquad d^{l+1}(n,n-1,n-1) + d^{l}(n-1,n+1,n) \\ &\qquad + c_{l,n+1} + c_{l+1,n} + c_{l+1,n-1}\} + c_{l-1,n} \\ &= G^{l-1}(n-1,n+1) + G^{l-1}(n,n), \end{aligned}$$

where the equality follows from (10).

Case 3: $m = n$. It is obvious that (7) is valid since by (A4),

$$G^{l}(m,m) = c_{lm} \le c_{l+1,m} + c_{l+1,m+1} \le G^{l}(m,m+1),$$

and similarly

$$G^{l}(m+1,m+1) \le G^{l}(m+1,m). \qquad \square$$

References

[1] S. Alpern, Search for point in interval, with high-low feedback, Math. Proc. Cambridge Philos. Soc. 98 (1985) 569–578.

[2] V.J. Baston and F.A. Bostock, A high-low search game on the unit interval, Math. Proc. Cambridge Philos. Soc. 97 (1985) 345–348.

[3] S.H. Cameron and S.G. Narayanamurthy, A search problem, Oper. Res. 12 (1964) 623–629.

[4] B.H. Eichhorn, Sequential search of an optimal dosage–non-Bayesyan methods, Naval Res. Logist. Quart. 20 (1973) 729.

[5] B.H. Eichhorn and S. Zacks, Sequential search of an optimal dosage, J. Amer. Statist. Assoc. 68 (1973) 594–598.

[6] M.R. Garey, Optimal binary trees with restricted maximal depth, SIAM J. Comput. 3 (1974) 101–110.

[7] A.M. Garsia and M.L. Wachs, A new algorithm for minimum cost binary trees, SIAM J. Comput. 6 (1977) 622.

[8] E.N. Gilbert and E.F. Moore, Variable length binary encodings, Bell System Tech. J. 38 (1959) 933–968.

[9] L. Gotlieb, Optimal multi-way search trees, SIAM J. Comput. 10 (1981) 422–433.

[10] R. Hassin, A dichotomous search for a geometric random variable, Oper. Res. 32 (1984) 423–439.

[11] R. Hassin and R. Hotovly, Asymptotic results of dichotomous search with search and travel costs, European J. Oper. Res., to appear.

[12] D. Heyman and M. Sobel, Stochastic Models in Operations Research, Vol II (McGraw-Hill, New York, 1984).

[13] K. Hinderer, On dichotomous search with direction-dependent costs for a uniformly hidden object, Optimization 21 (1990) 215–229.

[14] A.J. Hu, Selection of the optimum partition search, Computing 37 (1986) 261–264.
[15] T.C. Hu and K.C. Tan, Path length of binary search trees, SIAM J. Appl. Math. 22 (1972) 225–234.
[16] T.C. Hu and A.C. Tucker, Optimal computer search trees and variable-length alphabetic codes, SIAM J. Appl. Math. 21 (1971) 514–532.
[17] T.C. Hu and M.L. Wachs, Binary search on a tape, SIAM J. Comput. 16 (1987) 573–590.
[18] A. Itai, Optimal alphabetic trees, SIAM J. Comput. 5 (1976) 9–18.
[19] D.E. Knuth, Optimum binary search trees, Acta Inform. 1 (1971) 14–25.
[20] L.L. Larmore, Height restricted binary trees, SIAM J. Comput. 16 (1987) 1115–1123.
[21] G. Markowsky, Best Huffman trees, Acta Inform. 16 (1981) 363–370.
[22] R. Morris, Some theorems on sorting, SIAM J. Appl. Math. 17 (1969) 1–6.
[23] S. Murakami, A dichotomous search, J. Oper. Res. Soc. Japan 14 (1971) 127–142.
[24] S. Murakami, A dichotomous search with travel cost, J. Oper. Res. Soc. Japan 19 (1976) 245–254.
[25] D. Topkis, Minimizing a submodular function on a lattice, Oper. Res. 26 (1978) 305–321.
[26] M.L. Wachs, On an efficient dynamic programming technique of F.F. Yao, J. Algorithms 10 (1989) 518–530.
[27] R.L. Wessner, Optimal alphabetic search trees with restricted maximal height, Inform. Process. Lett. 4 (1976) 90–94.
[28] F.F. Yao, Efficient dynamic programming using quadrangle inequalities, in: Proceedings 12th Annual ACM Symposium on the Theory of Computing (1980) 429–435.
[29] F.F. Yao, Speed-up in dynamic programming, SIAM J. Algebraic Discrete Methods 3 (1982) 532–540.