JOURNAL OF ALGORITHMS 6, 265–274 (1985)

An Optimal Algorithm for Finding All the Jumps of a Monotone Step-Function

REFAEL HASSIN AND NIMROD MEGIDDO*

Statistics Department, Tel Aviv University, Tel Aviv, Israel 69978

Received July 26, 1983

The idea of binary search is generalized as follows. Given f: $\{0,1,\dots,N\} \to \{0,\dots,K\}$ such that $f(0)=0$, $f(N)=K$, and $f(i) \leqslant f(j)$ for $i<j$, all the "jumps" of f, i.e., all is such that $f(i) > f(i-1)$ together with the difference $f(i)-f(i-1)$ are recognized within $K\lfloor \log_2(N/K)\rfloor + \lfloor (N-1)2^{-\lfloor \log_2(N/K)\rfloor}\rfloor$ f-evaluations. This is proved to be the exact bound in the non-trivial case when $K \leqslant N$. An optimal strategy is as follows: The first query will be at $i = 2^m$, where $m = \lfloor \log_2(N/K)\rfloor$. An adversary will then respond either $f(i)=0$ or $f(i)=1$ as explained in the paper. © 1985 Academic Press, Inc.

If f: $\{0,1,\dots,N\} \to \{0,1\}$ is monotone and $f(0)=0$, $f(N)=1$, then f has one "jump" which can be recognized by a binary search within $\lceil \log_2 N \rceil$ f-evaluations. In this paper we generalize this situation to the case of a monotone nondecreasing step-function, i.e., f: $\{0,1,\dots,N\} \to \{0,1,\dots,K\}$, where $f(0)=0$, $f(N)=K$, and $f(i) \leqslant f(j)$ for $i<j$. Obviously, by performing K binary searches one may recognize $f(i)$ for all i, so that $K\lceil \log_2 N \rceil$ f-evaluations should suffice. Also, trivially, if $K \geqslant N$ then $N-1$ f-evaluations are sufficient and may also be necessary in the worst-case.

We prove in this paper that the exact upper bound on the number of f-evaluations required for the recognition of all the jumps is $K\lfloor \log(N/K)\rfloor + \lfloor (N-1)2^{-\lfloor \log(N/K)\rfloor}\rfloor$, where $\log X = \mathrm{Max}(0, \log_2 X)$ (and hence our bound is equal to $N-1$ if $K \geqslant N/2$). An optimal strategy may be described as follows. The first query will be at $i = 2^m$, where $m = \lfloor \log(N/K)\rfloor$. Let the response be $f(i) = K_1$ $(0 \leqslant K_1 \leqslant K)$. We now proceed, recursively, with

*Current address: IBM Research Laboratory, San Jose, Calif. 95193. The work of this author was supported in part by the National Science Foundation Grants ECS-8121741 and ECS-8218181.

0196-6774/85 $3.00

the two resulting problems, namely, finding all the jumps of f over the sets $\{0, 1, \ldots, i\}$ and $\{i, \ldots, N\}$.

One typical application of our problem is the following: Suppose that we need to recognize the exact boundaries of blocks of words whose first letters are identical. Such a problem arises when we wish to find a given vector in a preprocessed set of vectors (see [1–4]).

We now turn to a description of the problem in the form of a two-person zero-sum game. For the fundamental concepts of game theory see Owen [5].

A Game Representation

The following game is obviously equivalent to our problem of identifying all the jumps of a monotone step-function. Given a sequence of N boxes and a set of K identical balls, an adversary distributes the balls among the boxes and we have to discover how many balls are contained in each box. In other words, an adversary chooses N integers $n_1, \ldots, n_N$ ($n_i \geqslant 0, \Sigma n_i = K$) and we may ask for any number N_1 ($1 \leqslant N_1 \leqslant N - 1$) what is the *total* number of balls contained within the first N_1 boxes. Our goal is to minimize the number of queries while finding all the numbers n_i.

We denote by (N, K) the game which starts with N boxes and K balls. The different games are recursively related as follows. The first query splits the sequence $(1, \ldots, N)$ into two disjoint sequences of N_1 and N_2 boxes ($N_1 + N_2 = N$). The adversary responds by splitting the set of K balls into two sets of K_1 and K_2 elements ($K_1 + K_2 = K$). Thus, after the first query and the adversary's response our game actually reduces to the (independent) playing of two games: (N_1, K_1) and (N_2, K_2). Suppose that at the end of the play we have to pay to the adversary an amount equal to the number of queries. Let $V(N, K)$ denote the value of the resulting game. It follows that $V(N, 0) = 0$ and for $K \geqslant 1$, $V(N, K) = 1 + \mathrm{Min}_{N_1 + N_2 = N} \mathrm{Max}_{K_1 + K_2 = K} \{V(N_1, K_1) + V(N_2, K_2)\}$. It is also obvious that for $K \geqslant N$, $V(N, K) = N - 1$. The main result of this paper is

THEOREM. *For* $K \geqslant 1$,

$$(1)\quad V(N, K) = K\lfloor \log(N/K) \rfloor + \lfloor (N - 1) 2^{-\lfloor \log(N/K) \rfloor} \rfloor$$

($\log X = \mathrm{Max}(0, \log_2 X)$).

The proof will be established as follows. Let us denote the right-hand side of (1) by $F(N, K)$. In what follows, Lemma 1 is merely technical. Lemma 2 demonstrates the optimal strategy of the adversary, that is, provides a response scheme that guarantees the value of the game. In other words, Lemma 2 proves that $F(N, K)$ is a lower bound on the value of the game. Lemma 3 describes a strategy of queries which terminates the search within $F(N, K)$ queries against any adversary. In other words, Lemma 3 shows that $F(N, K)$ is also an upper bound.

To simplify notation we henceforth denote

(2) $m = \lfloor \log(N/K) \rfloor$ and define

$$\delta(X) = \begin{cases} 0 & \text{if } X > 0 \\ 1 & \text{if } X = 0. \end{cases}$$

LEMMA 1. *Every pair* (N, K) $(1 \leqslant K \leqslant N)$ *defines a unique triple of integers* (α, β, r) *such that* $0 < \alpha \leqslant K$, $\alpha + \beta = K$, $0 \leqslant r < 2^m$ *and* $N = \alpha 2^m + \beta 2^{m+1} + r$. *We can express* $F(N, K)$ *in terms of* α, β, *and* r *by* $F(N, K) = (m+1)(\alpha + \beta) + \beta - \delta(r)$.

Proof. It follows that there exists a unique pair of integers (β, r) such that $0 \leqslant \beta < K$, $0 \leqslant r < 2^m$, and $N = (K + \beta)2^m + r$. Let $\alpha = K - \beta$. It follows that $\alpha > 0$ and $N = \alpha 2^m + \beta 2^{m+1} + r$. This implies

$$\begin{aligned} F(N, K) &= Km + \lfloor (N-1)2^{-m} \rfloor \\ &= Km + \lfloor (\alpha 2^m + \beta 2^{m+1} + r - 1)2^{-m} \rfloor \\ &= Km + \alpha + 2\beta - \delta(r) \\ &= (m+1)(\alpha + \beta) + \beta - \delta(r). \end{aligned}$$

We are now ready to describe a strategy for the adversary which guarantees at least $F(N, K)$ queries. Notice that a strategy for the adversary is well defined if we specify for every N_1 $(1 \leqslant N_1 \leqslant N - 1)$ a number K_1 $(0 \leqslant K_1 \leqslant K)$ which is the adversary's response to our first query (in the game (N, K)) as to the number of balls contained in the first N_1 boxes.

The strategy and the proof of what it guarantees are contained in the following lemma.

LEMMA 2. *For every* N, K, *and* N_1 *such that* $1 \leqslant N_1 \leqslant N - 1$, *there exists a number* K_1 *such that* $0 \leqslant K_1 \leqslant K$ *for which*

$$F(N_1, K_1) + F(N_2, K_2) + 1 \geqslant F(N, K)$$

where $N_2 = N - N_1$ *and* $K_2 = K - K_1$.

Proof. We distinguish different cases for the definition of an optimal response K_1.

Case 1. $N_1 < \alpha 2^m$. Let $N_1 = i2^m + r_1$, where $i < \alpha$ and $0 \leqslant r_1 < 2^m$.

Subcase 1.1. $r \geqslant r_1$ or $\beta > 0$. In this case, the adversary chooses $K_1 = i$. If $i = 0$ then $F(N_1, K_1) = 0$. Otherwise (if $i \geqslant 1$),

$$\left\lfloor \log \frac{N_1}{K_1} \right\rfloor = \left\lfloor \log \frac{i2^m + r_1}{i} \right\rfloor = m$$

so that

$$F(N_1, K_1) = im + \left\lfloor \frac{i2^m + r_1 - 1}{2^m} \right\rfloor = i(m+1) - \delta(r_1).$$

In both cases, $F(N_1, K_1) \geqslant i(m+1) - \delta(r_1)$.

By the assumptions of this subcase,

$$\left\lfloor \log \frac{N_2}{K_2} \right\rfloor = \left\lfloor \log \frac{(\alpha - i)2^m + \beta 2^{m+1} + r - r_1}{\alpha + \beta - i} \right\rfloor = m$$

and

$$\begin{aligned} F(N_2, K_2) &= (\alpha + \beta - i)m + \left\lfloor \frac{(\alpha - i)2^m + \beta 2^{m+1} + r - r_1 - 1}{2^m} \right\rfloor \\ &= (m+1)(\alpha + \beta - i) + \beta + \left\lfloor \frac{r - r_1 - 1}{2^m} \right\rfloor \\ &\geqslant (m+1)(\alpha + \beta - i) + \beta - 1. \end{aligned}$$

Thus,

$$\begin{aligned} &F(N_1, K_1) + F(N_2, K_2) + 1 \\ &\quad\geqslant i(m+1) - \delta(r_1) + (m+1)(\alpha + \beta - i) + \beta \\ &\quad\geqslant (m+1)(\alpha + \beta) + \beta - \delta(r) \\ &\quad= F(N, K) \qquad \text{(by Lemma 1)}. \end{aligned}$$

Subcase 1.2. $\beta = 0$ and $r < r_1 < r + 2^{m-1}$. Again, the adversary chooses $K_1 = i$ and $F(N_1, K_1) \geqslant i(m+1) - \delta(r_1)$. If $i = K$ then $F(N_2, K_2) = 0$; otherwise (if $i < \alpha$),

$$\left\lfloor \log \frac{N_2}{K_2} \right\rfloor = \left\lfloor \log \frac{(\alpha - i)2^m + r - r_1}{\alpha - i} \right\rfloor = m - 1$$

so that

$$\begin{aligned} F(N_2, K_2) &= (\alpha - i)(m-1) + \left\lfloor \frac{(\alpha - i)2^m + r - r_1 - 1}{2^{m-1}} \right\rfloor \\ &= (\alpha - i)(m-1) + 2(\alpha - i) - 1. \end{aligned}$$

In both cases, $F(N_2, K_2) \geqslant (\alpha - i)(m+1) - 1$. Thus,

$$\begin{aligned} &F(N_1, K_1) + F(N_2, K_2) + 1 \\ &\quad\geqslant i(m+1) - \delta(r_1) + (\alpha - i)(m+1) \\ &\quad= (m+1)\alpha - \delta(r_1) \geqslant (m+1)\alpha - \delta(r) = F(N, K). \end{aligned}$$

Subcase 1.3. $\beta = 0$ and $r_1 \geqslant r + 2^{m-1}$. Recall that $r_1 < 2^m$. In this case the adversary chooses $K_1 = i + 1$. (The definition of i implies $i < \alpha$ so that $K = \alpha + \beta \geqslant i + 1$ and $K_1 = i + 1$ is feasible.) Here

$$\left\lfloor \log \frac{N_1}{K_1} \right\rfloor = \left\lfloor \log \frac{i2^m + r_1}{i + 1} \right\rfloor = m - 1$$

so that

$$\begin{aligned} F(N_1, K_1) &= (i + 1)(m - 1) + \left\lfloor \frac{i2^m + r_1 - 1}{2^{m-1}} \right\rfloor \\ &= (i + 1)(m - 1) + 2i + \left\lfloor \frac{r_1 - 1}{2^{m-1}} \right\rfloor. \end{aligned}$$

Note that if $\delta(r) = 0$ then $r > 0$ so that $\lfloor (r_1 - 1)/2^{m-1} \rfloor = 1$ and hence in any case $\lfloor (r - 1)/2^{m-1} \rfloor \geqslant 1 - \delta(r)$. Note that in the present subcase $N_1 = i2^m + r_1 < \alpha 2^m + r = N$ so that $i < \alpha$. If $i + 1 = K$ then $F(N_2, K_2) = 0$; otherwise ($i + 1 < \alpha$),

$$\left\lfloor \log \frac{N_2}{K_2} \right\rfloor = \left\lfloor \log \frac{(\alpha - i)2^m + r - r_1}{\alpha - i - 1} \right\rfloor = m$$

so that

$$\begin{aligned} F(N_2, K_2) &= (\alpha - i - 1)m + \left\lfloor \frac{(\alpha - i)2^m + r - r_1 - 1}{2^m} \right\rfloor \\ &= (\alpha - i - 1)m + \alpha - i - 1 \end{aligned}$$

thus,

$$\begin{aligned} &F(N_1, K_1) + F(N_2, K_2) + 1 \\ &\quad \geqslant (i + 1)(m - 1) + 2i + 1 - \delta(r) + (\alpha - i - 1)(m + 1) + 1 \\ &\quad = \alpha(m + 1) - \delta(r) \\ &\quad = F(N, K). \end{aligned}$$

Case 2. $N_1 \geqslant \alpha 2^m$. Let $N_1 = \alpha 2^m + i2^{m+1} + r_1$, where $0 \leqslant i \leqslant \beta$ and $0 \leqslant r_1 < 2^{m+1}$.

Subcase 2.1. $r_1 \leqslant r$ or $\beta = i$. Here, the adversary chooses $K_1 = \alpha + i$. It follows that

$$\left\lfloor \log \frac{N_1}{K_1} \right\rfloor = \left\lfloor \log \frac{\alpha 2^m + i2^{m+1} + r_1}{\alpha + i} \right\rfloor = \left\lfloor \log\left(2^m + \frac{i2^m + r_1}{\alpha + i}\right) \right\rfloor$$

so that if either $\alpha > 1$ or $r_1 < 2^m$ then $\lfloor \log(N_1/K_1) \rfloor = m$ and

$$F(N_1, K_1) = (\alpha + i)m + \left\lfloor \frac{\alpha 2^m + i2^{m+1} + r_1 - 1}{2^m} \right\rfloor$$
$$= (\alpha + i)(m + 1) + i + \left\lfloor \frac{r_1 - 1}{2^m} \right\rfloor.$$

If $\alpha = 1$ and $r_1 \geqslant 2^m$ then $\lfloor \log(N_1/K_1) \rfloor = m + 1$ so that

$$F(N_1, K_1) = (\alpha + i)(m + 1) + \left\lfloor \frac{\alpha 2^m + i2^{m+1} + r_1 - 1}{2^{m+1}} \right\rfloor$$
$$\geqslant (\alpha + i)(m = 1) + i,$$

If $i = \beta$ then $F(N_2, K_2) = 0$; otherwise (if $i < \beta$),

$$\left\lfloor \log \frac{N_2}{K_2} \right\rfloor = \left\lfloor \log \frac{(\beta - i)2^{m+1} + r - r_1}{\beta - i} \right\rfloor = m + 1$$

so that

$$F(N_2, K_2) = (\beta - i)(m + 1) + \left\lfloor \frac{(\beta - i)2^{m+1} + r - r_1 - 1}{2^{m+1}} \right\rfloor$$
$$= (\beta - i)(m + 1) = \beta - i - \delta(r - r_1).$$

In both cases $F(N_2, K_2) \geqslant (\beta - i)(m + 2) - \delta(r - r_1)$. Thus,

$$\begin{aligned} &F(N_1, K_1) + F(N_2, K_2) + 1 \\ &\quad \geqslant (\alpha + i)(m + 1) + i - \delta(r_1) + (\beta - i)(m + 2) \\ &\qquad + \delta(r_1) - \delta(r) - 1 + 1 \\ &\quad = (\alpha + \beta)(m + 1) + \beta - \delta(r) \\ &\quad = F(N, K). \end{aligned}$$

Subcase 2.2. $r < r_1 \leqslant r + 2^m$ and $\beta > i$. Again, the adversary chooses $K_1 = \alpha + i$ and $F(N_1, K_1) \geqslant (\alpha + i)(m + 1) + i - \delta(r_1)$. Here,

$$\left\lfloor \log \frac{N_2}{K_2} \right\rfloor = \left\lfloor \log \frac{(\beta - i)2^{m+1} + r - r_1}{\beta - i} \right\rfloor = m$$

so that

$$F(N_2, K_2) = (\beta - i)m + \left\lfloor \frac{(\beta - i)2^{m+1} + r - r_1 - 1}{2^m} \right\rfloor$$
$$= (\beta - i)m + 2(\beta - i) + \left\lfloor \frac{r - r_1 - 1}{2^m} \right\rfloor.$$

Since $r_1 > 0$, $\delta(r_1) = 0$ so that

$$\begin{aligned}
&F(N_1, K_1) + F(N_2, K_2) + 1 \\
&\quad \geqslant (\alpha + i)(m + 1) + i + (\beta - i)m + 2(\beta - i) + \left\lfloor \frac{r - r_1 - 1}{2^m} \right\rfloor + 1 \\
&\quad = (\alpha + \beta)(m + 1) + \beta + \left\lfloor \frac{r - r_1 - 1}{2^m} \right\rfloor + 1 \\
&\quad \geqslant (\alpha + \beta)(m + 1) + \beta - \delta(r) \\
&\quad = F(N, K)
\end{aligned}$$

Subcase 2.3. $r_1 > r + 2^m$ and $\beta > i$. Here the adversary chooses $K_1 = \alpha + i + 1$. If $K_2 = \beta - i - 1 = 0$ then $F(N_2, K_2) = 0$. Otherwise,

$$\left\lfloor \log \frac{N_2}{K_2} \right\rfloor = \left\lfloor \log \frac{(\beta - i)2^{m+1} + r - r_1}{\beta - i - 1} \right\rfloor = m + 1,$$

so that

$$\begin{aligned}
F(N_2, K_2) &= (\beta - i - 1)(m + 1) + \left\lfloor \frac{(\beta - i)2^{m+1} + r - r_1 - 1}{2^{m+1}} \right\rfloor \\
&= (\beta - i - 1)(m + 1) + (\beta - i - 1).
\end{aligned}$$

Also,

$$\begin{aligned}
\left\lfloor \log \frac{N_1}{K_1} \right\rfloor &= \left\lfloor \log \frac{\alpha 2^m + i 2^{m+1} + r_1}{\alpha + i + 1} \right\rfloor \\
&= \left\lfloor \log \frac{(\alpha + i + 1)2^m + i2^m + r_1 - 2^m}{\alpha + i + 1} \right\rfloor = m,
\end{aligned}$$

so that

$$\begin{aligned}
F(N_1, K_1) &= (\alpha + i + 1)m + \left\lfloor \frac{\alpha 2^m + i 2^{m+1} + r_1 - 1}{2^m} \right\rfloor \\
&= (\alpha + i + 1)m + (\alpha + 2i + 1) \\
&= (\alpha + i + 1)(m + 1) + i
\end{aligned}$$

and

$$\begin{aligned}1 + F(N_1, K_1) + F(N_2, K_2) &= 1 + (\alpha + i + 1)(m + 1) + i \\ &\quad + (\beta - i - 1)(m + 1) + \beta - i - 1 \\ &= (\alpha + \beta)(m + 1) + \beta \\ &\geqslant (\alpha + \beta)(m + 1) + \beta - \delta(r) \\ &= F(N, K).\end{aligned}$$

This completes the proof that the adversary can force at least $F(N, K)$ queries. We shall now prove that $F(N, K)$ queries always suffice. Essentially, an optimal query at the state of N cells and K balls is $N_1 = 2^m$ ($m = \lfloor \log(N/K) \rfloor$).

LEMMA 3. *For every N and $K \geqslant 2$, for all possible responses K_1 ($0 \leqslant K_1 \leqslant K$), $F(2^m, K_1) + F(N - 2^m, K - K_1) + 1 \leqslant F(N, K)$.*

Proof. (a) We first prove the lemma for $K_1 = 0$.

Case (a1). $N - 2^m \geqslant K2^m$. In this case $\lfloor \log(N - 2^m)/K \rfloor = m$ and

$$\begin{aligned}&F(2^m, 0) + F(N - 2^m, K) + 1 \\ &\quad = 0 + Km + \lfloor (N - 2^m - 1)/2^m \rfloor + 1 \\ &\quad = Km + \lfloor (N - 1)/2^m \rfloor = F(N, K).\end{aligned}$$

Case (a2). $N - 2^m < K2^m$. Since $N \geqslant K2^m$ it follows that $N - 2^m \geqslant (K - 1)2^m < K2^{m-1}$. Hence, in this case, $\lfloor \log(N - 2^m)/K \rfloor = m - 1$. Since we assume $N - 2^m < K2^m$ then $K > N/2^m - 1$ and thus $K + 1 - (N - 1)/2^m > 0$. Therefore,

$$\begin{aligned}&F(2^m, 0) + F(N - 2^m, K) + 1 \\ &\quad = K(m - 1) + \left\lfloor \frac{N - 2^m - 1}{2^{m-1}} \right\rfloor + 1 \\ &\quad = \lfloor (Km + (N - 1)/2^m) - (K + 1 - (N - 1)/2^m) \rfloor \\ &\quad \leqslant \lfloor Km + (N - 1)/2^m \rfloor = F(N, K).\end{aligned}$$

(b) Second, we prove the lemma for $K_1 = 1$. First, $F(2^m, 1) = m$. Also, since $m = \lfloor \log(N/K) \rfloor$, it follows that $2^m \leqslant N/K < 2^{m+1}$, so that

$$(K - 1)2^m \leqslant N - 2^m < (K - 1)2^{m+1} + 2^m.$$

Case (b1). $N - 2^m < (K-1)2^{m+1}$.
In this case $\lfloor \log(N - 2^m)/(K-1) \rfloor = m$ so that

$$\begin{aligned} &F(2^m, 1) + F(N - 2^m, K - 1) + 1 \\ &\quad = m + (K-1)m + \left\lfloor \frac{N - 2^m - 1}{2^m} \right\rfloor + 1 \\ &\quad = Km + \lfloor (N-1)/2^m \rfloor = F(N, K). \end{aligned}$$

Case (b2). $N - 2^m \geqslant (K-1)2^{m+1}$.
In this case $\lfloor \log(N - 2^m)/(K-1) \rfloor = m + 1$ and

$$\begin{aligned} &F(2^m, 1) + F(N - 2^m, K - 1) + 1 \\ &\quad = m + (K-1)(m+1) + \lfloor (N - 2^m - 1)/2^{m+1} \rfloor + 1 \\ &\quad = Km + K + \lfloor (N - 2^m - 1)/2^{m+1} \rfloor \\ &\quad = Km + \left\lfloor \frac{2^{m+1}K + N - 2^m - 1}{2^{m+1}} \right\rfloor \\ &\quad \leqslant Km + \left\lfloor \frac{(N + 2^m) + N - 2^m - 1}{2^{m+1}} \right\rfloor \\ &\quad = Km + \lfloor (2N-1)/2^{m+1} \rfloor \\ &\quad = Km + \lfloor (2N-2)/2^{m+1} \rfloor \\ &\quad = Km + \lfloor (N-1)/2^m \rfloor = F(N, K), \end{aligned}$$

where the inequality follows from our assumption that $K2^{m+1} \leqslant N + 2^m$.

(c) We now complete the proof assuming that $K_1 \geqslant 2$. Let $K_1 = 2^{k_1} + r_1$, where $0 \leqslant r_1 < 2^{k_1}$. We note that for $K_1 \geqslant 1$,

$$\left\lfloor \log \frac{2^m}{K_1} \right\rfloor = m - k_1 + \delta(r_1) - 1,$$

so that

$$\begin{aligned} F(2^m, K_1) &= K_1(m - k_1 + \delta(r_1) - 1) + \left\lfloor \frac{2^m - 1}{2^m - k_1 + \delta(r_1) - 1} \right\rfloor \\ &= K_1(m - k_1 - 1) + 2^{k_1+1} - 1. \end{aligned}$$

Let $K_2 = K - K_1$ and $m_2 = \lfloor \log(N - 2^m)/K_2 \rfloor$. Then

$$\begin{aligned} a &\equiv 1 + F(2^m, K_1) + F(N - 2^m, K_2) \\ &= K_1(m - k_1 - 1) + 2^{k_1+1} + K_2 m_2 + \left\lfloor \frac{N - 2^m - 1}{2^{m_2}} \right\rfloor. \end{aligned}$$

Since for $K_1 \geqslant 2$, $K_1(k_1+1) \geqslant 2^{k_1}(k_1+1) \geqslant 2^{k_1+1}$, it follows that

$$\begin{aligned} a &\leqslant K_1 m + K_2 m_2 + \left\lfloor \frac{N-2^m-1}{2^{m_2}} \right\rfloor \\ &= Km + K_2(m_2 - m) + \left\lfloor \frac{N-2^m-1}{2^{m_2}} \right\rfloor \\ &\leqslant Km + K_2(2^{m_2-m}-1) + \left\lfloor \frac{N-2^m-1}{2^{m_2}} \right\rfloor. \end{aligned}$$

It follows from the definition of m_2 that $2^{m_2} \leqslant (N-2^m)/K_2$ and thus $K_2 \leqslant (N-2^m)/2^{m_2}$. Therefore

$$\begin{aligned} a &\leqslant Km + \frac{N-2^m}{2^{m_2}} \cdot (2^{m_2-m}-1) + \left\lfloor \frac{N-2^m-1}{2^{m_2}} \right\rfloor \\ &= Km + \frac{N}{2^m} - \frac{N-2^m}{2^{m_2}} - 1 + \left\lfloor \frac{N-2^m-1}{2^{m_2}} \right\rfloor. \end{aligned}$$

Since

$$\frac{N}{2^m} - 1 \leqslant \left\lfloor \frac{N-1}{2^m} \right\rfloor \quad \text{and} \quad \left\lfloor \frac{N-2^m-1}{2^{m_2}} \right\rfloor - \frac{N-2^m}{2^{m_2}} \leqslant 0,$$

it follows that $a \leqslant Km + \lfloor (N-1)/2^m \rfloor = F(N,K)$. This completes the proof that $F(N,K)$ queries suffice, so that $V(N,K) = F(N,K)$.

References

1. J. L. Bentley and J. B. Saxe, Algorithms on vector sets, *SIGACT News* **11** (1979), 36–39.
2. R. Hassin and N. Megiddo, "Searching a Presorted Set of Vectors," Department of Statistics, Tel Aviv University, June 1980.
3. D. S. Hirschberg, On the complexity of searching a set of vectors, *SIAM J. Comput.* **9** (1980), 126–129.
4. S. Rao Kosaraju, On a multidimensional search problem, *SIGACT News* **11** (1979).
5. G. Owen, "Game Theory," Saunders, Philadelphia, 1968.