Lower Bounds for Selection in $X + Y$ and Other Multisets

DONALD B. JOHNSON AND SAMUEL D. KASHDAN

The Pennsylvania State University, University Park, Pennsylvania

ABSTRACT It is known that the structure in the multiset $X + Y$, for X and Y multisets of n real numbers, enables selection to be performed more quickly than in other sets of size $O(n^2)$ In this paper a lower bound of $\Omega(\max(n, K^{1/2} \log K))$ is given for selecting the Kth element in $X + X$, and thus in $X + Y$, for K selecting the first through the median element Selection of gaps, that is, consecutive differences in a sorted order, is shown to be $\Omega(n \log n)$ for all K These results hold also when inputs are restricted to integers The problem of selection on $X + X$ can be generalized to selection on the set of multisets of size m chosen from X, where the multisets are ranked on the sums of their elements When m and n grow, selection of the Kth largest multiset of size m is NP-hard when inputs are expressed in binary notation If subsets of X are characterized as paths, circuits, or spanning trees of an edge-weighted graph, then selection of the Kth largest such subset of X remains NP-hard It is not possible to show these problems to be NP-hard when inputs are expressed in unary, since algorithms exist which run in time polynomial in the sums of the values of the inputs

KEY WORDS AND PHRASES selection, computational complexity, decision trees, pairs, graphs, subgraphs, NP-hard problems

CR CATEGORIES. 5 25, 5 30

1. *Introduction*

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a multiset of n elements drawn from a universe with a linear order and let R be a characteristic function on the family of subsets of X. In the set $\mathscr{A} = \{A \mid A \subseteq X \text{ and } R(A) = 1\}$ of subsets of X selected by R it is desired to find the Kth member according to some function w which orders $\mathscr{A}$. For instance, $w\{x_i \mid i \in I\}$ where $I \subseteq \{1, 2, \ldots, n\}$ might be the *sum function* $w\{x_i \mid i \in I\} = \sum_{i \in I} x_i$ if $X \subset \mathbf{R}$.

When $R(A) = 1$ if and only if $|A| = 1$ (see Footnote 1) and w is the identity function, the problem is the well-known selection problem of finding the Kth element in X. The complexity of this problem is known to be bounded below by $c_1 n$ and above by $c_2 n$ steps for some positive constants $c_2 > c_1 \geq 1$ provided the linear order on X can be discovered by elementwise comparisons only [2, 15, 19] or even if $X \subset \mathbf{R}$ and comparisons of linear functions of X are allowed [16, 23]. The precise bounds depend on K but in no case can the problem be solved in fewer than K comparisons of linear functions

We study the more general problem where R selects other than singleton sets. An important case is the problem on pairs, where R selects all subsets of size two from X. The pairs problems confined to subsets is closely related to problems on $X + Y$ where X and Y are multisets of real numbers and $X + Y$ is the multiset of the sums of all ordered pairs in the Cartesian product $X \times Y$. Sorting of $X + Y$ has been treated by [8] and [6] Johnson and Mizoguchi [12] give an $O(n \log n)$ algorithm for selecting the Kth pair in $X + Y$ and thus for selecting the Kth subset of size two from X under any linear weight function

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery To copy otherwise, or to republish, requires a fee and/or specific permission

This research was supported in part by the National Science Foundation under Grant MCS 77-21092

Authors' address Computer Science Department, The Pennsylvania State University, University Park, PA 16802

© 1978 ACM 0004-5411/78/1000-0556 $00 75

[1] For A a set, $|A|$ denotes the cardinality of A

$w(x_i, x_j) = \alpha x_i + \beta x_j$ for $x_i, x_j \in X$ and $\alpha, \beta \in \mathbf{R}$. In this paper we present a lower bound of $\Omega(\max(n, K^{1/2}\log K))$ for the length of a worst-case computation in any decision tree for selecting the Kth subset of size two from X under sum and difference functions, for K selecting the first through the median subset. (For a function f, $\Omega(f(n))$ denotes any function which is greater than $cf(n)$ for some positive c and all but finitely many n.) The *difference function* is defined analogously to the sum function, that is, $\alpha = 1$ and $\beta = -1$. Of course, the lower bound just stated holds as well for selection on $X + Y$ and thus characterizes the complexity of selection problems on $X + Y$ and $X + X$ to within a constant factor for large K The techniques to obtain these lower bounds may be extended to provide an $\Omega(n \log n)$ lower bound for selecting the Kth largest gap on X, where a gap is a pair of consecutive members of X when X is arranged in descending order. The size of a gap is the difference of its members. This lower bound is uniform for $1 \leq K \leq n - 1$. Such a bound was known previously for the smallest gap [4, 20]

We first prove the aforementioned lower bounds for rational inputs and then strengthen these results by showing them to hold when inputs are confined to the integers.

For general K and subset size m, what is known about the complexity of the selection problem under the sum function where R selects all subsets of m elements from $X \subset \mathbf{R}$ is dependent on whether P = NP. In [12] it is shown that this selection problem is NP-hard when inputs are stated in binary, a result which cannot be strengthened to inputs stated in unary since an algorithm exists which runs in $O(\max(n \log n, K \log K))$ [12]. A reader unfamiliar with the P = NP problem is referred to [1] for a thorough discussion. We employ the term NP-hard to describe problems not known to be in NP but within a polynomial factor at least as difficult as any problem in NP.

Under sufficiently fast growing weight functions the selection problem just cited is not NP-hard. We show a nonanalytic weight function w under which there are polynomial-time algorithms for all values of K and m

Do fast algorithms exist for the selection problem for large K and large subsets if additional structure is imposed on $\mathscr{A}$? We consider cases where R selects certain subgraphs in a given graph G with either n vertices or n edges weighted with the elements of X. Lawler [14] has given an enumerative algorithm for the case where R selects paths between distinguished vertices of directed G with edge weights and no negatively weighted circuits allowed. Gabow [7] applies Lawler's algorithm to R selecting spanning trees. We give an algorithm where R selects elementary circuits in directed G. Each of these problems is shown by us to be NP-hard when inputs are stated in binary. Thus the algorithms cited are close to best possible under present knowledge regarding P = NP We give a necessary condition for a Kth best subset problem to have a fast algorithm, though we do not exhibit such a problem.

2. *Lower Bounds on Selecting the K-th Pair and K-th Gap*

Let $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbf{Q}^n$ be an arbitrary ordering of X, a set of n distinct rational numbers, and denote the set of distinct ordered pairs from X as $X^2 = \{(x_i, x_j) | x_i, x_j \in X$ and $x_i > x_j\}$ For w either the sum function or the difference function we now show a lower bound of $\Omega(\max(n, K^{1/2}\log K))$ on the complexity of finding the Kth largest pair (x_{i_K}, x_{j_K}) in some ordering $w(x_{i_1}, x_{j_1}) \geq w(x_{i_2}, x_{j_2}) \geq \cdots \geq w(x_{i_K}, x_{j_K}) \geq \cdots \geq w(x_{i_s}, x_{j_s})$ of the elements in $w(X^2)$, where $s = |X^2| = n(n-1)/2$. After these results are presented, the bound is shown also to hold when problem inputs X are restricted to sets of integers.

The computational model for which the bound is derived is a ternary decision tree in which are performed comparisons of the form $\mathbf{c} \cdot \mathbf{x} + c_{n+1}:0$, where $\mathbf{c}, \mathbf{x} \in \mathbf{R}^n$ and $c_{n+1} \in \mathbf{R}$. If we let $C(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x} + c_{n+1}.0$, three possible outcomes for a comparison may be denoted as $C(\mathbf{x}) > 0$, $C(\mathbf{x}) = 0$, $C(\mathbf{x}) < 0$. The complexity of a decision tree algorithm is the maximum, over all possible inputs X of size n, of the number of comparisons executed on one input. In developing our bound it will be sufficient to consider only inputs X for which X^2 has exactly one total order on the sums of pairs Thus, by the density of the rational numbers, we need consider only the decision tree paths on which all outcomes are strict, either $C(\mathbf{x}) > 0$ or $C(\mathbf{x}) < 0$.

For set X and weight function w for which $w(X^2)$ has exactly one total order, we denote the rank of $w(x_i, x_j) \in w(X^2)$ in this total order (taken as descending) as $\text{rank}_{w(X^2)}(x_i, x_j)$, abbreviated in what follows as $\text{rank}_{X^2}(x_i, x_j)$. A linear weight function for $\alpha, \beta \neq 0$ is called *nontrivial.*

LEMMA 1. *Let comparisons $C_1, C_2, \dots, C_r$ be given, let w be a nontrivial linear weight function, and let X be a set of n distinct real numbers with arbitrary ordering $\mathbf{x} = (x_1, x_2, \dots, x_n)$ on which the outcomes of the given comparisons are strict. Without loss of generality let these outcomes be $C_1(\mathbf{x}) > 0$, $C_2(\mathbf{x}) > 0, \dots, C_r(\mathbf{x}) > 0$. If for some $(x_{i_1}, x_{i_2}), (x_{i_3}, x_{i_4}) \in X^2$ both $w(x_{i_1}, x_{i_2}) > w(x_{i_3}, x_{i_4})$ and $w(x_{i_1}, x_{i_2}) < w(x_{i_3}, x_{i_4})$ are consistent with these outcomes then there exist two sets X_1 and X_2 of n rational numbers with corresponding orderings $\mathbf{x}_1$ and $\mathbf{x}_2$ which, for some K, have outcomes $C_1(\mathbf{x}_1) > 0$, $C_2(\mathbf{x}_1) > 0$, .. , $C_r(\mathbf{x}_1) > 0$, $C_1(\mathbf{x}_2) > 0$, $C_2(\mathbf{x}_2) > 0, \dots, C_r(\mathbf{x}_2) > 0$ and satisfy*

(i) *$\text{rank}_{X_1^2}(x_{1i_1}, x_{1i_2}) = \text{rank}_{X_2^2}(x_{2i_3}, x_{2i_4}) = K$,*

(ii) *$\text{rank}_{X_1^2}(x_{1i_3}, x_{1i_4}) = \text{rank}_{X_2^2}(x_{2i_1}, x_{2i_2}) = K + 1$,*

(iii) *for each $(x_{1j_1}, x_{1j_2}) \in X_1^2 - \{(x_{1i_1}, x_{1i_2}), (x_{1i_3}, x_{1i_4})\}$,*

$$\text{rank}_{X_1^2}(x_{1j_1}, x_{1j_2}) = \text{rank}_{X_2^2}(x_{2j_1}, x_{2j_2}).$$

PROOF. Let S be the set $S = \{\mathbf{a} \in \mathbf{R}^n | \, C_i(\mathbf{a}) > 0 \text{ for } 1 \le i \le r\}$. By the given conditions $\mathbf{x} \in S$ and, furthermore, there exist $\mathbf{p}_1 = (p_{11}, p_{12}, \dots, p_{1n})$ and $\mathbf{p}_2 = (p_{21}, p_{22}, \dots, p_{2n})$ in S for which $w(p_{1i_1}, p_{1i_2}) > w(p_{1i_3}, p_{1i_4})$ and $w(p_{2i_1}, p_{2i_2}) < w(p_{2i_3}, p_{2i_4})$. Since S is a convex and open set, the line segment $P = t\mathbf{p}_1 + (1 - t)\mathbf{p}_2$, $0 \le t \le 1$, is contained in S. There is exactly one point, $\mathbf{p}_3$, on P satisfying $w(p_{3i_1}, p_{3i_2}) = w(p_{3i_3}, p_{3i_4})$, and there exists $\epsilon > 0$ for which $\{\mathbf{p} | \epsilon > |\mathbf{p} - \mathbf{p}_3|\} \subset S$. We now show that there exists $\mathbf{p}^* \in S$ for which $(p^*_{i_1}, p^*_{i_2}) = w(p^*_{i_3}, p^*_{i_4})$ and, for every pair of pairs $(p^*_{j_1}, p^*_{j_2}), (p^*_{j_3}, p^*_{j_4})$, where $(j_1, j_2, j_3, j_4) \in J = \{(j_1, j_2, j_3, j_4) | 1 \le j_1 < j_2 \le n,\ 1 \le j_3 < j_4 \le n,\ (j_1, j_2) \neq (j_3, j_4)\} - \{(i_1, i_2, i_3, i_4), (i_3, i_4, i_1, i_2)\}$, it is true that $w(p^*_{j_1}, p^*_{j_2}) \neq w(p^*_{j_3}, p^*_{j_4})$.

Perhaps $\mathbf{p}_3$ satisfies the requirements for $\mathbf{p}^*$. If not, let $J_1 = \{(j_1, j_2, j_3, j_4) | (j_1, j_2, j_3, j_4) \in J \text{ and } (p_{3j_1}, p_{3j_2}) \neq w(p_{3j_3}, p_{3j_4})\}$ and let ϵ_1 satisfy $0 < \epsilon_1 < \min(\{|w(p_{3j_1}, p_{3j_2}) - w(p_{3j_3}, p_{3j_4})| \,|\, (j_1, j_2, j_3, j_4) \in J_1\} \cup \{\epsilon\})$. Then, for some $(j_1, j_2, j_3, j_4) \in J - J_1$ define $\mathbf{p}_3'$ as follows.

(i) If there exists $j \in \{j_1, j_2, j_3, j_4\}$ satisfying $j \notin \{i_1, i_2, i_3, i_4\}$ then let $p'_{3j} =$ (if $j = j_1$ or $j = j_3$ then $p_{3j} + \epsilon_1/\alpha$ else $p_{3j} + \epsilon_1/\beta$) and for $j' \neq j$ let $p'_{3j'} = p_{3j'}$.

(ii) Otherwise $\{j_1, j_2, j_3, j_4\} \subseteq \{i_1, i_2, i_3, i_4\}$. Then, assuming without loss of generality that $(j_1, j_2) \neq (i_1, i_2)$ and $(j_1, j_2) \neq (i_3, i_4)$, let $p'_{3j_1} =$ (if $j_1 = i_1$ or $j_1 = i_3$ then $p_{3j_1} + \epsilon_1/(2\alpha)$ else $p_{3j_1} + \epsilon_1/(2\beta)$), $p'_{3j_2} =$ (if $j_2 = i_1$ or $j_2 = i_3$ then $p_{3j_2} + \epsilon_1/(2\alpha)$ else $p_{3j_2} + \epsilon_1/(2\beta)$), and for $j \notin \{j_1, j_2\}$ let $p'_{3j} = P_{3j}$.

The construction preserves $w(p'_{3i_1}, p'_{3i_2}) = w(p'_{3i_3}, p'_{3i_4})$ and furthermore, if we now define J_1' on $\mathbf{p}_3'$ as above, it is clear that $|J - J_1'| < |J - J_1|$. So, redefining ϵ on $\mathbf{p}_3'$, the above transformation can be repeated a finite number of times to yield $\mathbf{p}^*$.

Perturbations of $\mathbf{p}^*$ in the above manner using a suitably small $\epsilon^* > 0$ yield two real points in the neighborhoods of which may be found rational points $\mathbf{x}_1$ and $\mathbf{x}_2$ satisfying the conditions of the lemma. ☐

LEMMA 2. *Let T be a ternary decision tree which determines the K-th largest pair of n real numbers with respect to a nontrivial linear weight function w by means of comparisons of linear functions, and let X be a set of n real numbers with arbitrary ordering $\mathbf{x}$ for which $w(X^2)$ has exactly one total order and the outcomes of all comparisons performed by T on X are strict. If T chooses $(x_{i_K}, x_{j_K}) \in X^2$ as the K-th largest pair, then for each $(x_i, x_j) \in X^2 - \{(x_{i_K}, x_{j_K})\}$ exactly one of $w(x_i, x_j) > w(x_{i_K}, x_{j_K})$ and $w(x_i, x_j) < w(x_{i_K}, x_{j_K})$ is consistent with these outcomes.*

PROOF. Suppose the lemma is false for some T and X satisfying the given conditions so that, for (x_{i_K}, x_{j_K}) chosen as the Kth largest pair, both $w(x_i, x_j) > w(x_{i_K}, x_{j_K})$ and

$w(x_i, x_j) < w(x_{i_K}, x_{j_K})$ are consistent with the outcomes of the comparisons performed by T for some $(x_i, x_j) \in X^2 - \{(x_{i_K}, x_{j_K})\}$. By Lemma 1 there exist X_1 and X_2, satisfying the conditions for X and with the same outcomes in T, with the property $\text{rank}_{X_1^2}(x_{1i_K}, x_{1j_K}) \neq \text{rank}_{X_2^2}(x_{2i_K}, x_{2j_K})$. But this is impossible since T correctly determines the Kth largest pair for any input of n real numbers. □

It should be noticed that Lemma 2 is in two senses as strong a partition lemma as possible. The lemma does not hold for decision trees performing comparisons on polynomials, as may be seen by considering the set $X = \{x_1, x_2, x_3\}$ of distinct real numbers and the selection problem on distinct pairs under the sum function for $K = 2$. If the comparison $(x_1 - x_2)(x_1 - x_3){:}0$ is performed with outcome $(x_1 - x_2)(x_1 - x_3) < 0$, it follows that (x_2, x_3) is the Kth pair. Yet the ranks of pairs (x_1, x_2) and (x_1, x_3) with respect to (x_2, x_3) are not determined. Difficulties also arise on inputs under which an "equals" branch is taken in the decision tree. Consider the set $X = \{x_1, x_2, x_3\}$ under the sum function for $K = 2$ where the comparison $x_1 + x_3 - 2x_2{:}0$ has the outcome $x_1 + x_3 - 2x_2 = 0$. In this case (x_1, x_3) is the Kth pair but the ranks of (x_1, x_2) and (x_2, x_3) are undetermined.

If the techniques in [16] are applied to the results of Lemmas 1 and 2, a lower bound no better than linear in K can be shown, since such a number of linear inequalities is sufficient to prove that a certain pair is indeed the Kth largest for a given input.

Because of the symmetry of the Kth largest pair and Kth smallest pair problems we restrict K so that $K \leq \lceil (n^2 - n)/4 \rceil$ under the sum function and $K \leq \lceil (n^2 - n)/2 \rceil$ under the difference function.

THEOREM 1. *A ternary decision tree T for finding the K-th largest pair in $w(X^2)$, for X a set of n rational numbers with arbitrary ordering* $\mathbf{x}$, *which employs comparisons of the form* $\mathbf{c} \cdot \mathbf{x} + c_{n+1}{:}0$ *for* $\mathbf{c} \in \mathbf{R}^n$ *and* $c_{n+1} \in \mathbf{R}$, *must make* $\Omega(\max(n, K^{1/2} \log K))$ *comparisons in the worst case for* $1 \leq K \leq \lceil n(n-1)/4 \rceil$ *under w the sum function and for* $1 \leq K \leq \lceil n(n-1)/2 \rceil$ *under the difference function.*

PROOF. The proof is in two parts. We prove the theorem first for the sum function, w^+.

For an arbitrary fixed linear decision tree T which selects the Kth largest member of $w^+(X^2)$ on input $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, define $B = \{b_1, b_2, \ldots, b_n\}$ an indexed set of rational numbers satisfying:

(i) For $1 \leq i < j \leq n$ and $1 \leq i' < j' \leq n$, if either $(i + j < i' + j')$ or $(i + j = i' + j'$ and $i < i')$ then $b_i + b_j > b_{i'} + b_{j'}$ else $b_{i'} + b_{j'} \geq b_i + b_j$.

(ii) For each $\mathbf{b}' \in \{(b_{\pi(1)}, b_{\pi(2)}, \ldots, b_{\pi(n)}) \mid \pi$ ranges over all permutations of $\{1, 2, \ldots, n\}\}$ and for each function $C(\mathbf{x})$ in T, $C(\mathbf{b}') \neq 0$.

The existence of $\mathbf{b} = (b_1, b_2, \ldots, b_n)$ satisfying (i) and (ii) can be demonstrated by the following observation. Let $\mathbf{b}''$ be defined so that $b_i'' = (n - i) + 1/2^i$ for $1 \leq i \leq n$. It can be verified that $\mathbf{b}''$ satisfies (i). By the density of the rational numbers there is a $\mathbf{b}$ within a sufficiently small $\epsilon > 0$ neighborhood of $\mathbf{b}''$ which satisfies both (i) and (ii).

Observe that requirement (i) implies $b_1 > b_2 > \quad > b_n$, and imposes a strict total ordering on the additive pairs $w^+(B^2) = \{b_i + b_j \mid b_i, b_j \in B$ and $b_i > b_j\}$. When $w^+(B^2)$ is represented as in Figure 1 by an upper triangular array, the ordering imposed means that all the pairs can be enumerated in descending order by listing pairs from upper right to lower left along succesive diagonals starting with the single pair diagonal $(b_1 + b_2)$ at the top left in Figure 1 and continuing to the diagonal at the lower right as follows (note that diagonal 1 is empty)

diagonal 2: $b_1 + b_2 >$
diagonal 3 $b_1 + b_3 >$
diagonal 4. $b_1 + b_4 > b_2 + b_3 >$
$\vdots \qquad \vdots$
diagonal l $b_1 + b_l > b_2 + b_{l-1} > \quad > b_{\lfloor l/2 \rfloor} + b_{\lceil l/2 + 1 \rceil} >$
$\vdots \qquad \vdots$
diagonal $2n - 2$. $b_{n-1} + b_n$

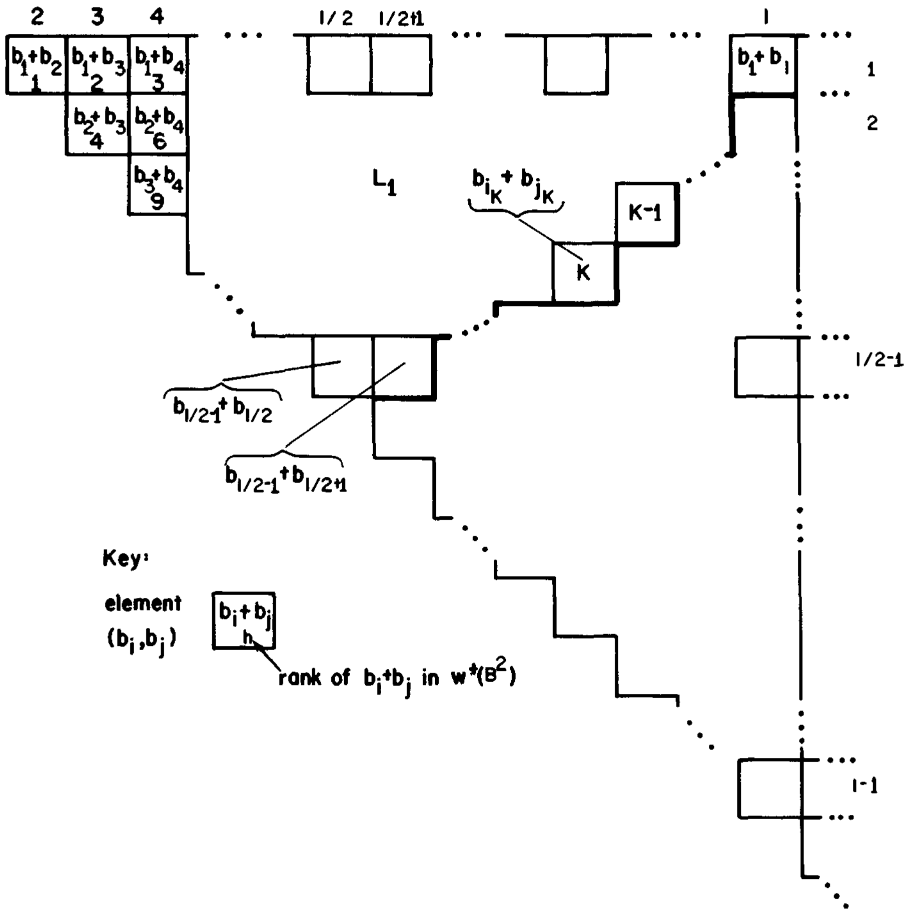

FIG 1 Exhibit of relevant portion of $w^+(B^2)$ showing L_1 (in the case l is even)

As illustrated in Figure 1, the Kth largest pair is contained in diagonal $l = \min\{l_1 | \lceil l_1^2/4 \rceil \ge K\}$ The lower bound on the number of comparisons follows by showing that no more than two vectors in the family of input vectors

$FB = \{(b_{\pi(1)}, b_{\pi(2)}, \ldots, b_{\pi(n)}) | \pi$ ranges over all permutations of $\{1, 2, \ldots, n\}$ satisfying $\pi(j) = j$ for $j > l\}$

define the same path in T

Each $\mathbf{b}' = (b_1', b_2', \ldots, b_n') \in FB$ defines a path $P = C_0C_1C_2 \cdots C_{h-1}\{i_K', j_K'\}$ in T starting at the root of T and terminating at a leaf labeled with the indices $\{i_K', j_K'\}$ of the Kth pair, where for each i, $0 \le i < h - 1$, if edge C_iC_{i+1} (or for $i = h - 1$, edge $C_{h-1}\{i_K', j_K'\}$) is labeled with $>$ then $C_i(\mathbf{b}') > 0$ else $C_i(\mathbf{b}') < 0$. Assume without loss of generality that $C_0(\mathbf{b}') > 0$, $C_1(\mathbf{b}') > 0$, $\ldots$, $C_{h-1}(\mathbf{b}') > 0$. Since the above h inequalities determine $b_{i_K}' + b_{j_K}'$ as the Kth largest pair, Lemma 2 indicates that these h inequalities partition B^2 This partition can be defined by

$$L_1' = \{\{i, j\} | C_0(\mathbf{b}') > 0, C_1(\mathbf{b}') > 0, \ldots, C_{h-1}(\mathbf{b}') > 0 \text{ implies } b_i' + b_j' \ge b_{i_K}' + b_{j_K}'\}.$$

For B let the analogous set be

$$L_1 = \{\{i, j\} | b_i + b_j \ge b_{i_K} + b_{j_K}\},$$

where $\{i_K, j_K\}$ are the indices of the Kth largest pair in $w^+(B^2)$ The set L_1 is illustrated in

Figure 1 as the set of pairs above and to the left of the dark diagonal boundary. Let π be the permutation satisfying $\mathbf{b}' = (b_{\pi(1)}, b_{\pi(2)}, \ldots, b_{\pi(n)})$ Then for each pair $\{i, j\}$, $\{i, j\} \in L_1$ if and only if $\{\pi(i), \pi(j)\} \in L_1'$.

CLAIM. *At most two distinct vectors in FB define path P*

PROOF OF CLAIM. We show that for $\mathbf{b}'$ which defines path P the permutation π relating $\mathbf{b}'$ and $\mathbf{b} = (b_1, b_2, \ldots, b_n)$ is determined to within a transposition of $\pi(i)$ and $\pi(i+1)$ for some fixed i. The determination of π depends only on the linear inequalities which define L_1', conditional on $\pi(i) = i$ for $i > l$.

If $|L_1'| = 1$ then $K = 1$ and $l = 2$ Therefore $L_1' = \{\{i_1, j_1\}\}$ where either $\pi(1) = i_1$ and $\pi(2) = j_1$ or $\pi(1) = j_1$ and $\pi(2) = i_1$ This verifies the claim for $|L_1'| = 1$.

For $|L_1'| > 1$, let $\{i_K, j_K\}$ be the indices of the Kth largest pair in $w^+(B^2)$

We now show a procedure by which $\pi(i)$ can be deduced for each $i = 1, 2, \ldots, l/2 - 1, l/2 + 2, \ldots, l$, where l is taken to be even. The proof for l odd is similar and will not be given. We extend the definitions of L_1 and L_1' as follows For $m = 1, 2, \ldots, l/2 - 1$

$$L_{m+1} = L_m - \{(i,j) | m \in \{i,j\}\}, \qquad L'_{m+1} = L'_m - \{(i,j) | \pi(m) \in \{i,j\}\}.$$

Let $\pi(L_m)$ denote L_m in which each index i is replaced by $\pi(i)$ As already established, $L_1' = \pi(L_1)$ It follows by induction on m that $L_m' = \pi(L_m)$ for $m = 1, 2, \ldots, l/2$. We may observe from Figure 1 that L_m is the set of elements of L_1 minus rows 1 through $m - 1$

It is given that L_1' and, consequently, $\{\pi(i_K), \pi(j_K)\}$ are known. Let L_m' be given for some $m = 1, 2, \ldots, l/2 - 2$. Because of the isomorphism between L_m' and L_m it follows (as may be seen by examination of Figure 1) that either there is a unique index j^* which appears in exactly one pair $\{i^*, j^*\} \in L_m'$ or there are two distinct indices j_1^* and j_2^*, each of which appears in exactly one pair $\{i_1^*, j_1^*\}$, $\{i_2^*, j_2^*\} \in L_m'$. In the latter case, $m = i_K$ and $i_1^* = i_2^* = \pi(i_K) = \pi(m)$. From this information, L'_{m+1} can be constructed. Knowledge of $\{\pi(i_K), \pi(j_K)\}$ reveals $\pi(l - m + 1) = \pi(j_K)$ and $\pi(l - m)$. In the former case, $i^* = \pi(m)$. From this information, L'_{m+1} can be constructed. If $m < i_K$ then $j^* = \pi(l - m + 1)$. If $m > i_K$ then $j^* = \pi(l - m)$. It follows by induction on m that the sets L_m' for $m = 1, 2, \ldots, l/2 - 1$ can be constructed in order. Either $L'_{l/2-1} = \{\{\pi(l/2 - 1), \pi(l/2)\}, \{\pi(l/2 - 1), \pi(l/2 + 1)\}\}$ and $L'_{l/2} = \varnothing$ or $L'_{l/2} = \{\{\pi(i_K), \pi(j_K)\}\}$. In either case $\pi(i)$ is determined for $i = 1, 2, \ldots, l/2 - 1, l/2 + 2, \ldots, l$. □ (claim)

Since no more than two vectors in FB can define any path P in T, the $l!$ vectors in FB define at least $\frac{1}{2}l!$ paths in T. Consequently there is a lower bound of $\Omega(l \log l)$ on the length of the longest path defined by a member of FB. Since K is approximately l^2, the number of comparisons performed in the worst case by T is $\Omega(K^{1/2} \log K)$.

We now give a proof for pairs weighted by the difference function w^-. Let T_1 be an arbitrary fixed linear decision tree which finds the Kth largest pair in $w^-(X^2)$, given a permutation of $X = \{x_1, x_2, \ldots, x_n\}$, a set of rational numbers as input.

Let $D = \{d_1, d_2, \ldots, d_n\}$ be an indexed set of distinct rational numbers satisfying:

(i) For $1 \le i < j \le n$ and $1 \le i' < j' \le n$ if either $(j - i > j' - i')$ or $(j - i = j' - i'$ and $i < i')$ then $d_i - d_j > d_{i'} - d_{j'}$, else $d_i - d_j \le d_{i'} - d_{j'}$.

(ii) For each $\mathbf{d}' \in \{(d_{\pi(1)}, d_{\pi(2)}, \ldots, d_{\pi(n)}) | \pi$ ranges over all permutations of $\{1, 2, \ldots, n\}\}$ and for each linear function $C(\mathbf{x})$ in T_1, $C(\mathbf{d}') \ne 0$.

As in the case for B, requirement (ii) ensures that Lemma 2 will be applicable. The existence of $\mathbf{d} = (d_1, d_2, \ldots, d_n)$ satisfying (i) and (ii) is established by defining $\mathbf{d}''$ so that $d_i'' = (n - i) + 1/2^i$ for $1 \le i \le n$. The vector $\mathbf{d}''$ so defined satisfies requirement (i) (It may be seen that $\mathbf{d}'' = \mathbf{b}''$, where $\mathbf{b}''$ is the vector used to establish the existence of B in the previous argument.) By the density of the rational numbers there is a $\mathbf{d}$ within a sufficiently small $\epsilon > 0$ neighborhood of $\mathbf{d}''$ which satisfies both (i) and (ii).

Observe that requirement (i) implies $d_1 > d_2 > \cdots > d_n$ and also imposes a strict total ordering on $w^-(D^2) = \{d_i - d_j | d_i, d_j \in D$ and $d_i > d_j\}$ The set $w^-(D^2)$ is depicted in Figure 2 so that, as in Figure 1, the differences can be listed in descending order by listing pairs, diagonal by diagonal, as follows.

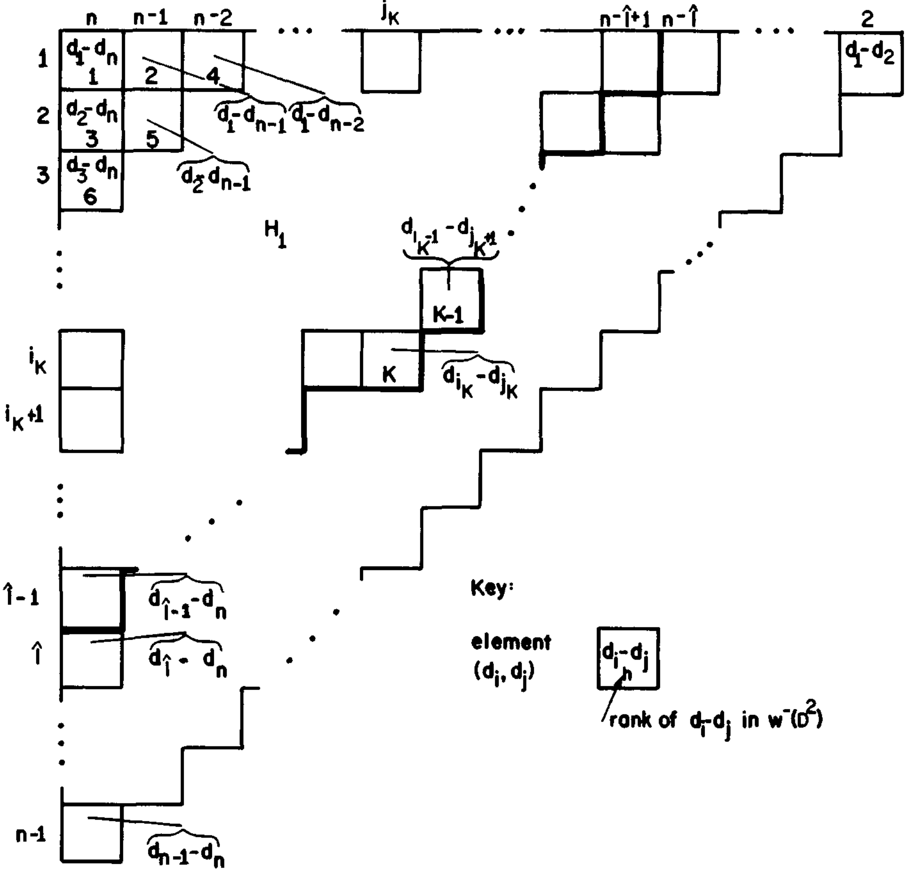

FIG 2 Exhibit of relevant portion of $w^-(D^2)$ showing H_1

$$
\begin{array}{l}
\text{diagonal } 1\text{: } d_1 - d_n > \\
\text{diagonal } 2\text{: } d_1 - d_{n-1} > d_2 - d_n > \\
\quad \vdots \qquad \vdots \\
\text{diagonal } \hat{l}\text{: } d_1 - d_{n-\hat{l}+1} > d_2 - d_{n-\hat{l}+2} > \quad > d_{\hat{l}} - d_n > \\
\quad \vdots \qquad \vdots \\
\text{diagonal } n-1\text{: } d_{n-1} - d_n
\end{array}
$$

In Figure 2 the Kth largest difference for $1 \le K \le \lceil (n^2 - n)/2 \rceil$ appears on diagonal $\hat{l} = \min\{l_1 \mid (l_1^2 + l_1)/2 \ge K\}$. The proof proceeds by showing that each member $\mathbf{d}'$ of the family of vectors

$FD = \{(d_{\pi(1)}, d_{\pi(2)}, \dots, d_{\pi(n)}) \mid \pi$ ranges over all permutations of $\{1, 2, \dots, n\}$ satisfying $\pi(j) = j$ for $j \ge \hat{l}\}$

defines a distinct path P in T_1. Let $\mathbf{d}'$ satisfy $C_0(\mathbf{d}') > 0$, $C_1(\mathbf{d}') > 0$, .. , $C_s(\mathbf{d}) > 0$ which, without loss of generality, defines path P starting at the root C_0 and terminating at a leaf labeled with (i'_K, j'_K), the indices with respect to $\mathbf{d}'$ of the Kth largest pair in $w^-(D^2)$. Then Lemma 2 indicates that the inequalities which define P imply a partition of D^2 given by

$$H'_1 = \{(i, j) \mid C_0(\mathbf{d}') > 0,\ C_1(\mathbf{d}') > 0,\ \dots,\ C_s(\mathbf{d}') > 0 \text{ implies } d'_i - d'_j \ge d'_{i'_K} - d'_{j'_K}\}.$$

Note that the pairs of indices are ordered because subtraction is not commutative. Let

$$H_1 = \{(i, j) \mid d_i - d_j \ge d_{i_K} - d_{j_K}\},$$

where (i_K, j_K) are the indices of the Kth largest pair in $w^-(D^2)$, and let π be the permutation satisfying $\mathbf{d}' = (d_{\pi(1)}, d_{\pi(2)}, \ldots, d_{\pi(n)})$ so that for each (i, j), $(i, j) \in H_1$ if and only if $(\pi(i), \pi(j)) \in H'_1$. We now show that H'_1 (which is determined by the s linear inequalities above) uniquely specifies π.

As before, we extend the definitions of H_1 and H'_1. For $m = 1, 2, \ldots, \hat{l}$,

$$H_{m+1} = H_m - \{(m, j) \mid 1 \le j \le n\}, \qquad H'_{m+1} = H'_m - \{(\pi(m), j) \mid 1 \le j \le n\}.$$

Since $H'_1 = \pi(H_1)$, it follows by induction on m that $H'_m = \pi(H_m)$ for $m = 1, 2, \ldots, \hat{l}$. As may be seen from Figure 2, H_m is the set of elements of H_1 minus rows 1 through $m - 1$ in the canonical representation of $w^-(D^2)$.

Let H'_m be given for some $m = 1, 2, \ldots, \hat{l} - 1$. Given the isomorphism between H'_m and H_m, it may be observed from Figure 2 that $\pi(m)$ is the index which appears more than any other as a first index in the pairs of H'_m Knowlege of $\pi(m)$ yields H'_{m+1}. Since H'_1 is given, it follows by induction that $\pi(m)$ can be obtained for $m = 1, 2, \ldots, \hat{l} - 1$.

Since $\pi(j) = j$ for $j \ge \hat{l}$, H'_1 uniquely determines π. Hence each $\mathbf{d}' \in FD$ corresponds to a unique path in T_1. From the existence of $(\hat{l} - 1)!$ distinct paths in T_1, it follows that there is a lower bound proportional to $\hat{l} \log \hat{l}$ on the number of inequalities required to describe some path in T_1. This bound is $\Omega(K^{1/2} \log K)$.

We complete the proof by noticing that finding the Kth largest pair requires $\Omega(n)$ comparisons regardless of K since in each of the two constructions above the largest element in the given set, B or D, is found in every case. By the results of [16] and [18] at least $n - 1$ comparisons are required. □

For $X = \{x_1, x_2, \ldots, x_n\}$, let π be a permutation of $\{1, 2, \ldots, n\}$ satisfying $x_{\pi(1)} \ge x_{\pi(2)} \ge \cdots \ge x_{\pi(n)}$ For each $1 \le i < n$ let $g_i = (x_{\pi(i)}, x_{\pi(i+1)})$ be the ith gap. Since each gap is a member of X^2, selection of the Kth largest gap may be viewed as a selection problem on X^2 with pair weight

$$w^G(x_i, x_j) = \begin{cases} 0 & \text{if } \ x_j > x_i \ \text{ or for some } \ x_k \in X,\ x_i > x_k > x_j, \\ x_i - x_j & \text{otherwise.} \end{cases}$$

Techniques similar to those used to prove Theorem 1 may be employed to provide an $\Omega(n \log n)$ lower bound for gap selection which is uniform for all K, $1 \le K \le n - 1$.

THEOREM 2. *Any decision tree which employs comparisons of linear functions to find the K-th largest gap (consecutive difference) in X, a multiset of n rational numbers, must make $\Omega(n \log n)$ such comparisons in the worst case.*

PROOF. Let T_g be a linear decision tree which selects the Kth largest gap g_{i_K} for input $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Let $g_i(\mathbf{x})$ be the weight function which determines the width of the ith gap, for $1 \le i \le n - 1$, defined by

$$g_i(\mathbf{x}) = x_{\pi(i)} - x_{\pi(i+1)},$$

where π is a permutation of $\{1, 2, \ldots, n\}$ satisfying $x_{\pi(1)} \ge x_{\pi(2)} \ge \cdots \ge x_{\pi(n)}$. The function $g_i(\mathbf{x})$ is continuous on the domain $\mathbf{R}^n$.

Our proof rests on the fact that each execution path labeled with strict inequalities in T_g determines a partition of the gap indices. This result is similar to the result proved in Lemmas 1 and 2 except, in this case, the weight function involved is not linear but piecewise linear.

CLAIM. *Let P be an execution path in T_g composed of edges labeled with strict inequalities These strict inequalities determine a partition $\{i \mid g_i(\mathbf{x}) \le g_{i_K}(\mathbf{x})\} \cup \{i \mid g_i(\mathbf{x}) > g_{i_K}(\mathbf{x})\}$ of the gap indices $\{1, 2, \ldots, n - 1\}$ which is invariant for all* $\mathbf{x}$ *which specify P.*

PROOF OF CLAIM. The argument presented here is similar to the argument presented in the proofs of Lemmas 1 and 2. Suppose that, contrary to the claim, the open convex region of $\mathbf{R}^n$ bounded by the linear inequalities labeling execution path P with leaf node label $g_{i_K} = (x_{i_K}, x_{i_{K+1}})$ contains $\mathbf{a}_1$ and $\mathbf{a}_2$ such that $\{i \mid g_i(\mathbf{a}_1) \le g_{i_K}(\mathbf{a}_1)\} \ne \{i \mid g_i(\mathbf{a}_2) \le g_{i_K}(\mathbf{a}_2)\}$. Without loss of generality assume there exists a gap j for which $g_j(\mathbf{a}_1) \le g_{i_K}(\mathbf{a}_1)$ and $g_j(\mathbf{a}_2) > g_{i_K}(\mathbf{a}_2)$. Since $g_j(\mathbf{x})$ is continuous there is a point $\mathbf{a}_3$ on the line segment joining $\mathbf{a}_1$ and $\mathbf{a}_2$ satisfying $g_j(\mathbf{a}_3) = g_{i_K}(\mathbf{a}_3)$. The point $\mathbf{a}_3$ also satisfies the inequalities labeling P. Let π be a permutation of $\{1, 2, \ldots, n\}$ satisfying $a_{3\pi(1)} \geq a_{3\pi(2)} \geq \cdots \geq a_{3\pi(n)}$ so that, for $\mathbf{a}_3$, $g_{i_K} = (a_{3\pi(i_K)}, a_{3\pi(i_K+1)})$ and $g_j = (a_{3\pi(j)}, a_{3\pi(j+1)})$. For some $\epsilon > 0$ let $\mathbf{a}_4$ and $\mathbf{a}_5$ be defined so that, for $1 \leq i < n$,

$$a_{4\pi(i)} = \begin{cases} a_{3\pi(i)} + \epsilon & \text{if } i \leq j, \\ a_{3\pi(i)} & \text{if } i > j; \end{cases}$$

$$a_{5\pi(i)} = \begin{cases} a_{3\pi(i)} + \epsilon & \text{if } i \leq i_K, \\ a_{3\pi(i)} & \text{if } i > i_K. \end{cases}$$

According to this definition, $g_j(\mathbf{a}_4) > g_{i_K}(\mathbf{a}_4)$ and $g_j(\mathbf{a}_5) < g_{i_K}(\mathbf{a}_5)$, while for each i, $1 \leq i < n$ and $i \neq j$ and $i \neq i_K$, $g_i(\mathbf{a}_3) = g_i(\mathbf{a}_4) = g_i(\mathbf{a}_5)$. Thus only the relative size of the jth gap and the i_Kth gap are interchanged for $\mathbf{a}_4$ and $\mathbf{a}_5$. For sufficiently small $\epsilon > 0$, $\mathbf{a}_4$ and $\mathbf{a}_5$ must lie in the interior of the convex region bounded by the strict inequalities which define P, but g_{i_K} cannot be the Kth largest gap for both $\mathbf{a}_4$ and $\mathbf{a}_5$. The claim follows from this contradiction. □ (claim)

Let $D = \{d_1, d_2, \ldots, d_n\}$ be the particular set shown in the proof of Theorem 1 to satisfy requirements (i) and (ii), but here defined with respect to some fixed T_g which selects the Kth largest gap. Each $\mathbf{d}'$ in FD, here the family of all permutations of D, satisfies the requirement $g_1(\mathbf{d}') > g_2(\mathbf{d}') > \cdots > g_{n-1}(\mathbf{d}')$, so that the Kth largest gap is $g_K = (d_K, d_{K+1})$. Each execution path specified by some member of FD produces the same partition of gaps $\{i \mid i \leq K\} \cup \{i \mid i > K\}$, but the indices of the Kth pair with respect to arbitrary input $\mathbf{x}$ (and $\mathbf{d}'$ in particular) may differ.

We now show that each member of FD specifies a distinct execution path in T_g. Suppose that, to the contrary, FD contains two distinct members $\mathbf{d}_1$ and $\mathbf{d}_2$ which specify the same execution path in T_g. Let π and σ be two permutations of $\{1, 2, \ldots, n\}$ satisfying

$$\mathbf{d} = (d_1, d_2, \ldots, d_n) = (d_{1\pi(1)}, d_{1\pi(2)}, \ldots, d_{1\pi(n)}) = (d_{2\sigma(1)}, d_{2\sigma(2)}, \ldots, d_{2\sigma(n)}).$$

Since $\mathbf{d}_1$ and $\mathbf{d}_2$ define the same path, and the gap $g_K = (d_{1\pi(K)}, d_{1\pi(K+1)}) = (d_{2\sigma(K)}, d_{2\sigma(K+1)})$, it follows that $\pi(K) = \sigma(K)$ and $\pi(K+1) = \sigma(K+1)$. Since the subset of $\mathbf{R}^n$ defined by the execution path P is convex, each member of the line segment $L = \{\mathbf{x} \mid \mathbf{x} = \alpha\mathbf{d}_1 + (1-\alpha)\mathbf{d}_2, 0 \leq \alpha \leq 1\}$ also satisfies the inequalities labeling path P. Each $\mathbf{x} \in L$ satisfies $x_{\pi(K)} = d_K$ and $x_{\pi(K+1)} = d_{K+1}$ but, since $\mathbf{d}_1$ and $\mathbf{d}_2$ are distinct, there must be an index i for which x_i is not fixed for all $\mathbf{x}$ in L. Consider the following case analysis for some x_i which is not fixed for all $\mathbf{x}$ in L.

Case 1. Clearly x_i may not take values which satisfy $d_K > x_i > d_{K+1}$, since this contradicts the assumption that (d_K, d_{K+1}) is the Kth largest gap for all $\mathbf{x}$ in L. Therefore all values of x_i are restricted to satisfy $d_1 \geq x_i \geq d_K$ (case 2) or all values of x_i are restricted to satisfy $d_{K+1} \geq x_i \geq d_n$ (case 3).

Case 2 Suppose x_i is restricted to satisfy $d_1 \geq x_i \geq d_K$. Let i_1 be the greatest index less than K for which $x_{\pi(i_1)}$ is not fixed for all members of L. Thus $x_{\pi(i_1)}$ takes values which satisfy $d_{2\pi(i_1)} \geq x_{\pi(i_1)} \geq d_{1\pi(i_1)}$, according to our choice of i_1. Since $\pi(i_1) \neq \sigma(i_1)$, $x_{\sigma(i_1)}$ takes values which satisfy $d_{1\sigma(i_1)} \geq x_{\sigma(i_1)} \geq d_{2\sigma(i_1)}$. Because $x_{\pi(i_1)}$ grows and $x_{\sigma(i_1)}$ decreases as $\mathbf{x}$ varies from $\mathbf{d}_1$ to $\mathbf{d}_2$ in L, there must be a point $\mathbf{x}'$ in L at which $x'_{\pi(i_1)} = x'_{\sigma(i_1)}$, which means that there is a gap g_l, $l < K$, such that $g_l(\mathbf{x}') = 0 < g_K(\mathbf{x}')$ which violates the partition requirement for path P.

Case 3. Suppose x_i is restricted to satisfy $d_{K+1} \geq x_i \geq d_n$. Let j_1 be the least index for which $x_{\pi(j_1)}$ is not fixed for $\mathbf{x}$ in L. The index j_1 satisfies $j_1 > K+1$. We show that there is an $\mathbf{x}''$ in L at which $g_{j_1}(\mathbf{x}'') > g_K(\mathbf{x}'')$ in violation of the partition requirement for execution path P. By the definition of $\mathbf{d}$ there is a small $\epsilon > 0$ so that

$$\begin{aligned} d_{j_1-1} &> (n - j_1 + 1) + 1/2^{j_1-1} - \epsilon > (n - j_1) + 1/2^{j_1} + \epsilon \\ &> d_{j_1} > (n - j_1) + 1/2^{j_1} - \epsilon > (n - j_1 - 1) + 1/2^{j_1+1} + \epsilon > d_{j_1+1}. \end{aligned}$$

By our choice of j_1, values of $x_{\pi(j_1)}$ satisfy $d_{1\pi(j_1)} \geq x_{\pi(j_1)} \geq d_{2\pi(j_1)}$, where $d_{2\pi(j_1)} \leq d_{j_1+1}$. Since $\pi(j_1) \neq \sigma(j_1)$, values of $x_{\sigma(j_1)}$ satisfy $d_{2\sigma(j_1)} \geq x_{\sigma(j_1)} \geq d_{1\sigma(j_1)}$, where $d_{1\sigma(j_1)} \leq d_{j_1+1}$. For each x_j,

$d_{K+1} \geq x_j \geq d_n$, satisfying $j \neq \pi(j_1)$ and $j \neq \sigma(j_1)$, x_j is either fixed or restricted to satisfy $d_{j_1+1} \geq x_j \geq d_n$. At $\mathbf{x}'' = \frac{1}{2}(\mathbf{d}_1 + \mathbf{d}_2)$, g_{j_1-1} is the gap between $x''_{\pi(j_1-1)}$ and $x''_{\pi(j_2)}$ which satisfies

$$\begin{aligned} x''_{\pi(j_2)} = \max_{j_1 \leq j \leq n} \{x''_{\pi(j)}\} &\leq \max\{x''_{\pi(j_1)}, x''_{\sigma(j_1)}, d_{j_1+1}\} \\ &\leq \max\left\{\frac{d_{1\pi(j_1)} + d_{2\pi(j_1)}}{2}, \frac{d_{1\sigma(j_1)} + d_{2\sigma(j_1)}}{2}, d_{j_1+1}\right\} \\ &\leq \tfrac{1}{2}(d_{j_1} + d_{j_1+1}) \\ &< \tfrac{1}{2}\left(\left((n - j_1) + \frac{1}{2^{j_1}} + \epsilon\right) + \left((n - j_1 - 1) + \frac{1}{2^{j_1+1}} + \epsilon\right)\right) \\ &< (n - j_1 - \tfrac{1}{2}) + \frac{3}{2^{j_1+2}} + \epsilon; \end{aligned}$$

$$\begin{aligned} g_{j_1-1}(\mathbf{x}'') &= x''_{\pi(j_1-1)} - x''_{\pi(j_2)} > d_{j_1-1} - \left((n - j_1 - \tfrac{1}{2}) + \frac{3}{2^{j_1+2}} + \epsilon\right), \\ g_{j_1-1}(\mathbf{x}'') &> \tfrac{3}{2} + 5/2^{j_1+2} - 2\epsilon, \\ g_K(\mathbf{x}'') &= d_K - d_{K+1} < 1 + 1/2^{K+1} + 2\epsilon. \end{aligned}$$

Therefore $g_{j_1-1}(\mathbf{x}'') > g_K(\mathbf{x}'')$, which violates the partition requirement for P. Therefore for $1 \leq i \leq n$, x_i is fixed for all $\mathbf{x}$ in L, which implies that $\mathbf{d}_1 = \mathbf{d}_2$ in contradiction to our assumption. Each of the $n!$ members of FD defines a distinct execution path, so the height of T_g is at least $\lfloor \log n! \rfloor + 1$. $\square$

In the preceding results, decision tree input values were allowed to range over the rational numbers. However, the same lower bounds are obtainable when the problem input values are restricted to the integers

THEOREM 3. *Theorems 1 and 2 hold even if input set X is restricted to the integers.*

PROOF. In the proof of Theorem 1, $\mathbf{b}$ and $\mathbf{d}$ were each chosen to satisfy two requirements Since $\mathbf{b}$ and $\mathbf{d}$ are rational vectors, according to requirement (ii) in the definitions of B and D, respectively, no permutation $\mathbf{b}'$ in FB nor any permutation $\mathbf{d}'$ in FD satisfies the equality outcome of a linear comparison. It follows that suitably large constants α_1 and α_2 can be found so that $\alpha_1\mathbf{b}$ and $\alpha_2\mathbf{d}$ are integer vectors and $\alpha_1\mathbf{b}$ and $\alpha_2\mathbf{d}$ satisfy requirements (i) and (ii) in the definitions of B and D, respectively. $\square$

3. *Complexity of the K-th Largest Subset Problems with Nonanalytic w*

The power of the decision tree model when comparisons between nonanalytic functions of the inputs are allowed has been considered in [16, 18] Reingold [18] has illustrated this power by demonstrating how the largest of n numbers can be selected using only $\lfloor \log n \rfloor + 1$ exponential comparisons. From this illustration we may expect similar logarithmic reductions in the number of comparisons required to solve other selection problems, such as those considered here, when comparisons between exponential functions of the inputs are allowed. Alternatively, a result of a similar flavor may be obtained when the Kth largest subset is determined using linear comparisons, but the weight function which ranks the subsets contains terms which are exponential in the inputs.

Consider finding the Kth largest subset of size m in the set of integers $A = \{a_1, a_2, \dots, a_n\}$ under weight function $w(\{a_i \mid i \in I\}) = \sum_{i \in I} 2^{a_i}$. Let the members of A be indexed in descending order so that $a_i > a_{i+1}$, $1 \leq i \leq n - 1$. Since $2^{a_i} > \sum_{j=i+1}^{n} 2^{a_j}$ for $1 \leq i < n$ it follows that all $\binom{n-i}{m}$ subsets of size m drawn from $\{a_{i+1}, a_{i+2}, \dots, a_n\}$ have weight less than any subset of size m in which a_i appears. Consequently the members of $M = \{a_{i_1}, a_{i_2}, \dots, a_{i_m}\}$, the Kth largest subset of size m, can be determined by finding the largest member a_{i_1} in M which must satisfy the relation $\binom{n-i_1}{m} < \binom{n}{m} - K + 1 \leq \binom{n-i_1+1}{m}$ The remaining members of M can be determined recursively by finding the $(\binom{n}{m} - K + 1 - \binom{n-i_1}{m})$ -th largest subset of size $m - 1$ in $\{a_{i+1}, a_{i+2}, \dots, a_n\}$. This successive determination of $a_{i_1} > a_{i_2} > \cdots > a_{i_m}$ thus depends on finding indices $i_1 < i_2 < \cdots < i_m$ so that

i_1 satisfies $\binom{n-i_1}{m} < \binom{n}{m} - K + 1 \le \binom{n-i_1+1}{m}$,

i_2 satisfies $\binom{n-i_2}{m-1} < \binom{n}{m} - K + 1 - \binom{n-i_1}{m} \le \binom{n-i_2+1}{m-1}$,

$\vdots$

i_l satisfies $\binom{n-i_l}{m-l+1} < \binom{n}{m} - K + 1 - \sum_{j=1}^{l-1} \binom{n-i_j}{m-j+1} \le \binom{n-i_l+1}{m-l+1}$,

$\vdots$

i_m satisfies $\binom{n-i_m}{1} < \binom{n}{m} - K + 1 - \sum_{j=1}^{m-1} \binom{n-i_j}{m-j+1} \le \binom{n-i_m+1}{1}$.

For an arbitrary set X of n distinct integers, determination of the Kth largest subset depends on finding a permutation π of $\{1, 2, \dots, n\}$ so that $x_{\pi(1)} \ge x_{\pi(2)} \ge \cdots \ge x_{\pi(n)}$. It follows from the above discussion that, for any fixed K, the leaves of a linear decision tree which determines π may be labeled with the Kth largest subset $\{x_{\pi(i_1)}, x_{\pi(i_2)}, \dots, x_{\pi(i_m)}\}$. Hence finding the Kth largest subset of size m requires at most $n \log n$ linear comparisons. □

4. *Problems in Which R Selects Subgraphs*

Given the intractability of the Kth subset problem on unrestricted subsets of size m, it is natural to ask if there are not interesting problems in which the imposition of structure on the subsets makes computation easier. For certain problems the answer appears to be in the negative.

A *path* in a directed graph $G = (V, E)$ containing n vertices in the vertex set V is a subgraph composed of a sequence of vertices, which we denote without loss of generality as $v_1, v_2, \dots, v_l$, and edges $(v_i, v_{i+1}) \in E$ for $i = 1, 2, \dots, l-1$ A path is a *circuit* if $v_l = v_1$. A circuit is *elementary* if for i, j, $1 \le i < j \le l$, $v_i = v_j$ implies $i = 1$. A circuit is *Hamiltonian* if it is elementary and contains all vertices of G. The definitions of *path, circuit, elementary circuit, Hamiltonian circuit* are stated identically for undirected graphs. A *spanning tree* of an undirected, connected graph G is a connected subgraph containing all the vertices of G and containing no circuit.

The weight of a path or circuit is defined using the sum function employing $w: E \to \mathbf{R}$.

THEOREM 4. *Finding the K-th spanning tree in an undirected graph with edge weights is NP-hard when inputs are stated in binary notation.*

PROOF. We reduce the Hamiltonian circuit problem for undirected graphs to the Kth spanning tree problem.

Given an undirected graph $G = (V, E)$ with n vertices $v_1, v_2, \dots, v_n$ for $i = 1, 2, \dots, n$ let $w(v_i) = 2^{i-1}$. For each edge (u, v) in E let $w(u, v) = w(u) + w(v)$. If there is a spanning tree of weight $2^{n+1} - 2 - w(u, v)$ and (u, v) is an edge, then there is a Hamiltonian circuit, containing (u, v), in G (This is shown below.) If for no edge (u, v) in E there is such a spanning tree, then G is not Hamiltonian. Let the cost of finding a Kth spanning tree be $T(n)$. It is necessary to search for a spanning tree with respect to each of at most $n(n-1)/2$ edges. There are at most n^{n-2} spanning trees in an n-vertex graph. Thus a binary search for a given weight tree is $O(n \log n)$. We conclude that the complexity of the Hamiltonian circuit problem, known to be NP-complete [13], is $O(n^3 \log n T(n))$.

It remains to show that, for some edge (u, v), there is in G a Hamiltonian circuit containing (u, v) if and only if there is a spanning tree in G of weight $2^{n+1} - 2 - w(u, v)$. If there is a Hamiltonian circuit, which we denote without loss of generality as $(u = v_1, v_2, \dots, v_n = v, v_1)$, then there is a spanning tree $\{(v_i, v_{i+1}) \mid i = 1, 2, \dots, n-1\}$ with weight $\sum_{i=1}^{n-1} w(v_i, v_{i+1}) = 2^{n+1} - 2 - w(u, v)$. On the other hand, if S is a spanning tree of G with weight $w(S) = 2^{n+1} - 2 - w(u, v)$ let $x_i + 1$ equal the number of times vertex v_i appears in an edge of $S \cup \{u, v\}$, for $i = 1, 2, \dots, n$. Thus $\sum_{i=1}^{n} 2^{i-1}(x_i + 1) = 2^{n+1} - 2$ or, alternatively $\sum_{i=1}^{n} 2^{i-1} x_i = 2^n - 1$. Since S spans G, every vertex appears in at least one edge of $S \cup \{u, v\}$, so $x_i \in \{0, 1, \dots, n\}$. Since $S \cup \{u, v\}$ contains exactly n edges, $\sum_{i=1}^{n} x_i = n$. It is easily shown that $\sum_{i=1}^{n} 2^{i-1} x_i = 2^n - 1$ has the unique solution $x_i = 1$ for $i = 1, 2, \dots, n$, under these restrictions. Thus every vertex appears exactly twice in an edge of $S \cup \{u, v\}$. Since S is connected, $S \cup \{u, v\}$ must be a Hamiltonian circuit. □

The problem of finding best elementary circuits must be posed in such a way as to make it impossible to make a weight "better" by traversing a circuit twice or more. Otherwise the traveling salesman problem can be reduced to finding the best elementary circuit, and the simple optimization problem is NP-complete. The reduction used was first observed by Dantzig, Blattner, and Rao [3] in connection with shortest path problems. Thus we restrict the problems of finding the Kth elementary circuit and the Kth path to graphs without circuits of negative weight. The problems must then be posed to find the Kth *smallest* objects in each case.

If we assume that the Kth smallest elementary circuit of G can be found in time $T(n)$ or less then the Hamiltonian circuit problem for any directed graph G (known to be NP-complete [13]) can be solved by assigning a weight of 1 to each edge of G. Binary search of the circuit space of at most $\sum_{i=1}^{n-1} \binom{n}{n-i+1} (n-i)!$ circuits, using the algorithm for a Kth elementary circuit, will determine if there is an elementary circuit of weight n. This determines if G is Hamiltonian. The computation will cost $O(n \log nT(n))$. A similar argument can be used to determine if a graph is Hamiltonian, given the Kth shortest path can be found in time $T(n)$. Thus we have the next theorem.

THEOREM 5. *The problems to determine the K-th smallest elementary circuit or the K-th shortest path between vertices in a weighted directed graph are NP-hard when inputs are stated in binary notation.* □

The above results show that the enumerative algorithms we will shortly mention for the Kth best shortest path, spanning tree, or elementary circuit are not subject to major improvement without showing P = NP and, of course, the existence of these algorithms shows that Theorems 4 and 5 do not hold under inputs stated in unary notation. What are the possibilities that there are interesting Kth subset problems (that is, an interesting property R) provably solvable in polynomial time? If there are such, the complement of R in unrestricted subsets will yield an NP-hard problem, as the following theorem shows.

THEOREM 6. *Let R be a property on subsets $X \subseteq A = \{a_1, a_2, \ldots, a_n\} \subset \mathbf{R}$ satisfying $|\{X \mid R(X)\}| = \Omega(n^r)$ for all $r = 1, 2, \ldots$. If there is a $t \in \{1, 2, \ldots\}$ such that there exist algorithms for finding both the K-th largest X in $T = \{X \mid R(X)\}$ and the K-th largest X in $S = \{X \mid \text{not } R(X)\}$ and both algorithms have running times bounded by $O(n^t)$, then P = NP.*

PROOF. Given polynomially bounded algorithms for the Kth largest X in both T and S, we show how to find the Kth largest unrestricted subset in polynomial time. Let $v = |\{X \mid X \subseteq A\}|$ and let $K_0 = \lfloor v/2 \rfloor$. Find X_0, the K_0-th largest subset in T. By binary search in S find $u_0 = |\{X \mid X \in S \text{ and } w(X) \geq w(X_0)\}|$. Thus X_0 has rank $K_0 + u_0$ in $S \cup T$. If $K_0 + u_0 = K$ then stop. Otherwise continue to search for X of rank K by binary search, i.e. let $K_1 = \lfloor v/4 \rfloor$ if $K_0 + u_0 > K$, or let $K_1 = \lfloor 3v/4 \rfloor$ if $K_0 + u_0 < K$ and repeat the above procedure. The procedure will halt in polynomial time. If the Kth subset in $S \cup T$ is in fact in T it will have been found. Otherwise a similar procedure will find it in S.

A simple reduction shows the Kth unrestricted subset problem is NP-hard. If indeed this problem could be solved in polynomial time then the partition problem on sets of integers could be solved in polynomial time. The partition problem is known to be NP-complete [13]. The reduction involves a binary search of the space of all subsets for one with exactly half the weight of the entire set. This search would require $O(n)$ applications of an algorithm for the unrestricted subset. □

5. *An Algorithm for the K-th Smallest Elementary Circuit of a Directed Graph*

Given the results of Section 4, we now consider algorithms which enumerate in order the K best subgraphs chosen by R in order to find the Kth best. Several authors have studied unordered enumeration of subgraphs. Johnson [9] treats enumeration of elementary circuits, and Read and Tarjan [17] treat enumeration of spanning trees, paths, and elementary circuits. These algorithms do not adapt to an ordered enumeration of the K best subgraphs.

Lawler [14] gives a general procedure for enumerating the best K subsets (selected by some R) in order of weight which runs in $O(KnT(n))$ time where an algorithm running in $O(T(n))$ time is given which discovers the best subset selected by R. For the problem of the K shortest paths between a pair of distinguished vertices of a directed graph with n vertices he shows an $O(Kn^3)$ algorithm, a saving of a factor of n over direct application of the general algorithm. Gabow [7] performs a similar condensation of the computation in Lawler's general algorithm to achieve an $O(K|E|\alpha(n, |E|) + |E| \log |E|)$ bound for finding the best K spanning trees of an undirected graph with n vertices and $|E|$ edges. The function α is the inverse of an Ackerman-type function described by Tarjan [21] Using techniques we describe below in connection with our elementary circuit algorithm, these bounds can be reduced on edge-sparse graphs.

Let $\mathscr{C}(G, P)$ be the set of all the elementary circuits of the graph G which contain the path $P = (e_1, e_2, \ldots, e_p)$ and let $C(G, (e_1, \ldots, e_p)) = (e_1, \ldots, e_p, \ldots, e_r)$ be the least weight circuit in $\mathscr{C}(G, (e_1, \ldots, e_p))$. It follows from [14] that $\{(e_1, \ldots, e_p, \ldots, e_r)\} \cup \bigcup_{i=p}^{r-1} \mathscr{C}(G - \{e_{i+1}\}, (e_1, \ldots, e_p, \ldots, e_i))$ is a partition of $\mathscr{C}(G, (e_1, \ldots, e_p))$ Call the circuits in the set $\bigcup_{i=p}^{r-1} \{C(G - \{e_{i+1}\}, (e_1, \ldots, e_p, \ldots, e_i))\}$ *neighbors with respect to* $C(G, (e_1, \ldots, e_p))$. The tree of circuits generated by recursive partitioning according to this rule is a heap ordered on circuit weight, with a least weight circuit at the root. The algorithm we present below constructs a related binary heap. Let $C(G, P)$ be a circuit in this binary heap. Then the left offspring of $C(G, P)$ is the least weight circuit in the set of neighbors with respect to $C(G, P)$ The right offspring of $C(G, P)$ is the next circuit in order of weight in the set of neighbors to which $C(G, P)$ belongs Either offspring may be void if no circuit fills the stated conditions. For example, the root circuit does not have a right offspring (it belongs to no neighbor set). If $\mathscr{C}(G, P)$ is a singleton then $C(G, P)$ has no offspring. Ties with respect to weight are broken according to some fixed lexicographic ordering on the set of edges of G

The algorithm stores a frontier of the binary heap just described. The next circuit to be enumerated will necessarily be a circuit of least weight in this frontier. When the circuit is enumerated, it is deleted from the frontier and replaced by its offspring. Offspring are generated as follows. If *no* edge is constrained to be in the desired circuit it must be a left offspring and can in general appear anywhere in the graph derived from the given graph by deletion of the first edge of the parent circuit. A straightforward application of the all pairs shortest path algorithm due to Floyd [5] and Warshall [22] suffices. Or, if sparsity is to be capitalized on, the algorithm by Johnson [11] may be used. This algorithm runs in time $O(n^{(3-(m-1)/m)} + n|E|)$ on directed graphs with n vertices, $|E|$ edges, for some fixed integer $m > 1$. On the other hand, when edges *are* constrained to be in the desired circuit these edges necessarily form a path $(e_1, \ldots, e_p)$ and the path is closed by finding a disjoint shortest path connecting its end vertices in the graph from which edge e_i of the parent circuit has been deleted. Suitable adaptations of shortest path algorithms for this purpose will be mentioned following presentation of the algorithm for listing the K smallest elementary circuits in order.

The circuits in the frontier of the binary heap are stored in the set Q in which a circuit C is represented by a 6-tuple $(\hat{\imath}, \hat{U}, \hat{C}, i, U, C)$, where $\hat{C}$ is the parent circuit of C. The set U is the set of forbidden edges under which C is least in weight, and the index i is the index of the last edge of $\hat{C}$ required in C Thus $i + 1$ is the index (in $\hat{C}$) of the edge of $\hat{C}$ forbidden in C. The parameters $\hat{U}$ and $\hat{\imath}$ are similarly defined from $\hat{C}$ and *its* parent In other words $C = C(G - U, (e_1, \ldots, e_i))$ where $\hat{C} = (e_1, \ldots, e_p, \ldots, e_{i+1}, \ldots, e_r)$ and $U = \hat{U} \cup \{e_{i+1}\}$.

ALGORITHM LEAST K CIRCUITS

Input. Directed graph $G = (V, E)$, where $|V| = n$ and $E \subseteq V \times V$, distance function $d: E \to \mathbf{R}$, and rank K
Output Sequence $C_1, C_2, \ldots, C_K$ of the K smallest elementary circuits of G in order of weight, $w(C_1) \leq w(C_2) \leq \cdots \leq w(C_K)$, where $w(C) = \sum_{j=1}^{r} d(e_{i_j})$ for $C = (e_{i_1}, e_{i_2}, \ldots, e_{i_r})$

Method

(1) Let G be presented as the $n \times n$ distance matrix $D_E = (d_{ij})$ where

$$d_{ij} = \begin{cases} d(v_i, v_j) & \text{if } (v_i, v_j) \in E, \\ \infty & \text{otherwise} \end{cases}$$

(2) Let C be a least weight elementary circuit of D

(3) Put $(0, \varnothing, (\,), 0, \varnothing, C)$ into Q

(4) For $l = 1, 2, \ldots, K$ do

(4a) Let $T = (\hat{i}_l, \hat{U}_l, \hat{C}_l, i_l, U_l, C_l)$ minimize $w(C)$ in Q Delete T from Q

(4b) Output C_l

(4c) Without loss of generality let $\hat{C}_l = (e_1, e_2, \ldots, e_r)$ where $e_j = (u_j, v_j)$ for $j = 1, 2, \ldots, r$
Construct the set
$\hat{S} = \{(C, i) \mid$ (i) C is distinct from C_l, and
(ii) $w(C) \geq w(C_l)$ and if $w(C) = w(C_l)$ then C follows C_l in some fixed lexicographic ordering on E, and
(iii) ($i = 0$, $\hat{i}_l = 0$, and C is a least weight circuit in $D_{E-(\hat{U}_l \cup \{e_1\})}$) or ($\max(\hat{i}_l, 1) \leq i < r$ and $C = (e_1, e_2, \ldots, e_i, P_i)$ where P_i is a shortest path from v_i to v_1 in $D_{E-(\hat{U}_l \cup \{e_{i+1}\})}$)$\}$
If $\hat{S} \neq \varnothing$ then add $(\hat{i}_l, \hat{U}_l, \hat{C}_l, i, U_l \cup \{e_{i+1}\}, C)$ to Q where (C, i) minimizes $w(C)$ in $\hat{S}$.

(4d) Without loss of generality let $C_l = (e_1, e_2, \ldots, e_r)$ where $e_j = (u_j, v_j)$ for $j = 1, 2, \ldots, r$
Construct the set
$S = \{(C, i) \mid$ (i) C is distinct from C_l, and
(ii) $w(C) \geq w(C_l)$ and if $w(C) = w(C_l)$ then C follows C_l is some fixed lexicographic ordering on E, and
(iii) ($i = 0$, $i_l = 0$, and C is a least weight circuit in $D_{E-(U_l \cup \{e_1\})}$) or ($\max(i_l, 1) \leq i < r$ and $C = (e_1, e_2, \ldots, e_i, P_i)$ where P_i is a shortest path from v_i to v_1 in $D_{E-(U_l \cup \{e_{i+1}\})}$)$\}$
If $S \neq \varnothing$ then add $(i_l, U_l, C_l, i, U_l \cup \{e_{i+1}\}, C)$ to Q where (C, i) minimizes $w(C)$ in S

(4e) While $|Q| < K - l$ do.
Delete from Q a 6-tuple $(\hat{m}, \hat{U}, \hat{C}, m, U, C)$ maximizing $w(C)$ in Q

In steps (2), (4c), and (4d) least weight circuits may be found directly from the output of either of the algorithms cited above.

The efficient computation of all the shortest paths in steps (4c) and (4d) is not possible by a direct application of the usual all pairs shortest path algorithms because each path is to be found with respect to a graph modified by the progressive deletion of edges. Lawler adapts the standard all pairs algorithm to compute the shortest path information for all the elements (C, i), $i > 0$, in time $O(n^3)$. The all pairs algorithm by Johnson [11] which takes advantage of sparsity can also be used since it operates in two stages. In the first stage a function is computed which allows a graph with negative edge weights to be reweighted with a set of positive weights equivalent with respect to shortest paths. Then a single source shortest path calculation is performed from each of the desired vertices. It can be shown that the function computed in the first stage is preserved under edge deletion. Thus the algorithm can be adapted to both requirements of steps (4c) and (4d) of the elementary circuit algorithm. The advantage is that use of priority queues of fixed depth [10] allows edge sparsity of a graph to be taken advantage of.

A straightforward correctness and complexity analysis yields the final theorem. We omit the proof.

THEOREM 7. *Algorithm LEAST K CIRCUITS can be implemented to enumerate in order the K circuits of least weight in a directed graph with n vertices and* $|E|$ *edges in* $O(Kn(n^{(2-(m-1)/m)} + |E|))$ *time for fixed integer* $m > 1$.

ACKNOWLEDGMENT. The authors are pleased to acknowledge several suggestions by a referee which led to clarifications in the proof of Theorem 1 and the present (stronger) statement of Theorem 2.

REFERENCES

1 AHO, A V, HOPCROFT, J E, AND ULLMAN, J D *The Design and Analysis of Computer Algorithms* Addison-Wesley, Reading, Mass, 1974

2 BLUM, M, FLOYD, R W, PRATT, V R, RIVEST, R L, AND TARJAN, R E Time bounds for selection *J Comptr Syst Sci* 7, 4 (Aug 1973), 448–461

3 DANTZIG, G.B , BLATTNER, W O , AND RAO, M R All shortest ıoutes from a fixed orıgın ın a graph *Theory of Graphs,* Int Symp , Rome, July 1966, Gordon and Breach, New York, 1967, pp 85–90
4 DOBKIN, D.P , AND LIPTON, R J On the complexıty of computatıons under varyıng sets of prımıtıves *J Comptr Syst Scı.* To appear.
5 FLOYD, R.W. Algorıthm 97 Shortest path *Comm ACM 5,* 6 (June 1962), 345.
6 FREDMAN, M L. How good ıs the ınformatıon theory bound ın sortıng? *Theoret. Comptr Scı 1* (1976), 355-361.
7 GABOW, H N Two algorıthms for generatıng weıghted spannıng trees ın order *SIAM J Comptng 6,* 1 (March 1977), 139–150
8 HARPER, L H , PAYNE, T H , SAVAGE, J E , AND STRAUS, E Sortıng $X + Y$ *Comm ACM 18,* 6 (June 1975), 347-349
9 JOHNSON, D B Fındıng all the elementary cırcuıts of a dırected graph *SIAM J Comptng 4,* 1 (March 1975), 77–84
10 JOHNSON, D B Prıorıty queues wıth update and fındıng mınımum spannıng trees *Inform Proc Letters 4,* 3 (Dec 1975), 53–57
11 JOHNSON, D B Effıcıent algorıthms for shortest paths ın sparse networks *J ACM 21,* 1 (Jan 1977), 1–13
12 JOHNSON, D B , AND MIZOGUCHI, T Selectıng the Kth element ın $X + Y$ and $X_1 + X_2 + \quad + X_m$ *SIAM J Comptng. 7,* 2 (May 1978), 147–153
13 KARP, R M Reducıbılıty among combınatorıal problems *Complexıty of Computer Computatıons,* R E Mıller and J W Thatcher, Eds , Plenum Press, New York, 1972, pp. 85–103
14 LAWLER, E L A procedure for computıng the K best solutıons to dıscrete optımızatıon problems and ıts applıcatıon to the shortest path problem *Manage Scı 18,* 7 (March 1972), 401–405
15 PRATT, V R , AND YAO, F F On lower bounds for computıng the ı-th largest element Conf Rec 14th Annual IEEE Symp on Swıtchıng and Automatıon Theory, Oct 1973, pp 70–81
16 RABIN, M O Provıng sımultaneous posıtıvıty of lınear forms *J Comptr Syst Scı 6,* 6 (Dec 1972), 639–650
17 READ, R C , AND TARJAN, R E Bounds on backtrack algorıthms for lıstıng cycles, paths, and spannıng trees *Networks 5,* 3 (July 1975), 237-252
18 REINGOLD, E M On the optımalıty of some set algorıthms *J ACM 19,* 4 (Oct 1972), 649–659
19 SCHONHAGE, A , PATERSON, M , AND PIPPENGER, N Fındıng the medıan *J Comptr Syst Scı 13,* 2 (Oct 1976), 184–199.
20 SHAMOS, M I , AND HOEY, D J Closest-poınt problems Conf Rec 16th Annual IEEE Symp Foundatıons of Comptr Scı , Oct 1975, pp 151–162
21 TARJAN, R E Effıcıency of a good but not lınear set unıon algorıthm *J ACM 22,* 2 (Aprıl 1975), 215–225
22 WARSHALL, S A theorem on Boolean matrıces *J ACM 9,* 1 (Jan 1962), 11–12
23 YAO, A C On the complexıty of comparıson problems usıng lınear functıons Conf Rec 16th Annual IEEE Symp on Foundatıons of Comptr Scı , Oct 1975, pp 85–89

RECEIVED OCTOBER 1976, REVISED AUGUST 1977