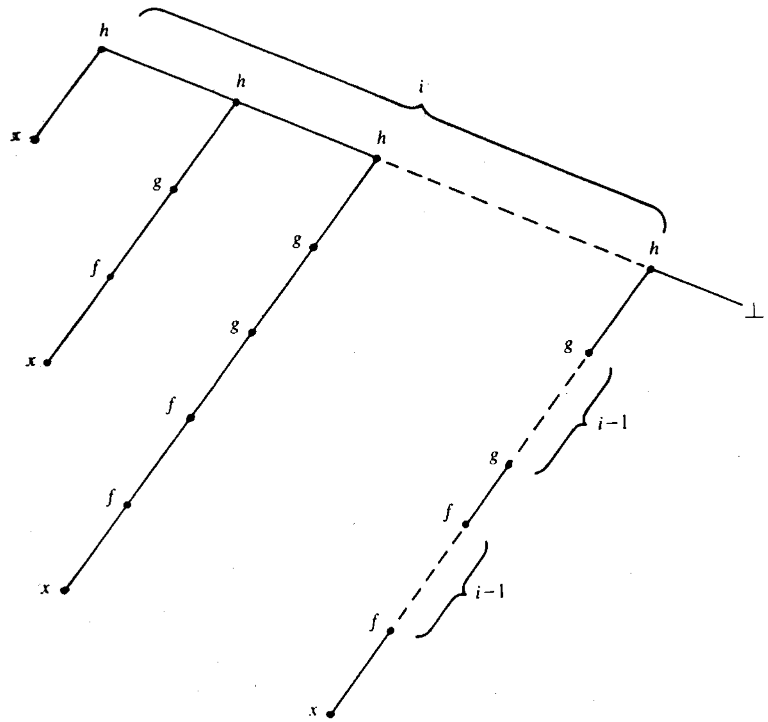

FIG. 2

This example shows a weakness of the approach using n-rational Σ-algebras. These investigations are left for further research.

Acknowledgments. I wish to thank Irene Guessarian, Saul Gorn, Scott Weinstein and especially Eric Wagner, for their helpful comments.

REFERENCES

[*See Part I, this issue, pp. 774–775.*]

SIAM J. COMPUT.
Vol. 13, No. 4, November 1984

THE INFORMATION-THEORETIC BOUND IS GOOD FOR MERGING*

NATHAN LINIAL†

Abstract. Let $A=(a_1>\cdots>a_m)$ and $B=(b_1>\cdots>b_n)$ be given ordered lists: also let there be given some order relations between a_i's and b_j's. Suppose that an unknown total order exists on $A\cup B$ which is consistent with all these relations (= *a* linear extension of the partial order) and we wish to find out this total order by comparing pairs of elements $a_r:b_s$. If the partial order has N linear extensions, then the Information Theoretic Bound says that $\log_2 N$ steps will be required in the worst case from any such algorithm. In this paper we show that there exists an algorithm which will take no more than $C\log_2 N$ comparisons where $C=(\log_2((\sqrt{5}+1)/2))^{-1}$. The computation required to determine the pair $a_r:b_s$ to be compared has length polynomial in $(m+n)$. The constant C is best possible. Many related results are reviewed.

Key words. theoretic bound, partially ordered sets, order ideals, lattice paths, convex polygons

1. Introduction and review. This paper is a part of an effort to answer the question "How good is the Information Theoretic Lower Bound." This question had already received considerable attention, e.g., [*Fr*][*GYY*1]. For many algorithmic problems, the quest of an answer is equivalent to searching a certain space whose elements are referred to as "compatible solutions" in the sense that they do not contradict the presently available information concerning the solution. Let us assume that our queries concerning the solution are such that they permit exactly two answers (the generalization to other cases is obvious). Thus the space of compatible solutions is split into two parts according to the answer. Assuming answers are given by an adversary, we may assume that the actual answers are always such that we are left with the majority of the compatible solutions after each query. The best one can do is to make such a query for which the space of compatible solutions is split into two equal parts. For this optimal strategy the number of steps will thus be $\log_2 N_0$ where N_0 is the initial number of compatible solutions. The problem is of course that in many situations such an efficient query which splits the compatible solutions into two sets of equal size does not exist. The purpose of this paper is to investigate the quality of the ITB under such circumstances.

This general model of a problem encompasses a great variety of search–sort problems and the situation varies from one problem to another. In many interesting families of problems which are included in this model the following situation occurs: although in general one cannot always find an optimal query which splits the space of compatible solutions into two equal parts, one can find a constant $\frac{1}{2}\geqq\alpha>0$ such that a query can always be found for which the smaller subspace has size at least α times the size of whole compatible solution space. (So the size of the large subspace is at most $(1-\alpha)$ times the size of the whole space.) In this case it is clear that the solution can be found in $\log N_0/\log(1-\alpha)$ steps (N_0 again being the initial size of the space of compatible solutions). In such cases the ITB gives the right order of magnitude for the optimal number of steps. Sometimes a somewhat more complicated result can be stated: there is an integer k and a constant $0<\beta<1$ so that one can always find k queries with the property that no matter what answers one receives the size of the remaining subspace is at most β times the size of the space before these queries were

* Received by the editors August 7, 1982, and in revised form July 14, 1983.

† Institute of Mathematics and Computer Science, The Hebrew University of Jerusalem, Jerusalem 91904 Israel.

made. Clearly the ITB gives here, too, the correct order of magnitude for the optimal number of steps.

Let us review some previous work in which this situation has been shown to occur;

(1) [LS] Let (T, r) be a tree rooted at r. The space of compatible solutions consists of all subtrees of T rooted at r. The queries are: a node x in T is picked and one asks whether x belongs to the chosen subtree. One proves here:

a) There is always a node which belongs to a fraction α of the compatible trees where $\frac{1}{3} \leqq \alpha \leqq \frac{2}{3}$.

b) One can always find $k \leqq 3$ vertices (=queries) so that after these queries are answered, the size of the compatible solution space drops to at most λ^k of its initial size where $\lambda = 5^{-1/3}$. The constant λ is best possible and the cases of equality are completely characterized.

2) Let $(P, \geqq)$ be a finite poset and consider the space of its nonempty ideals (=down sets; $A \subseteq P$ is an ideal if $x \in A$, $y < x$ implies $y \in A$). A query is made here by asking whether an element x of P belongs to the chosen ideal or not. This space is related to a large variety of search problems (see [LS]). Sands [Sa] has shown that if one restricts the attention to posets of height $\leqq k$, then there is a constant $\alpha_k < \frac{1}{2}$ so that one can always find an $x \in P$(=a query) such that the fraction of those order ideals containing x is between α_k and $1-\alpha_k$. A major open problem in this field is the following:

Problem 1 [Sa]. [LS]. Prove that there is a universal constant $0 < \alpha < \frac{1}{2}$ so that in any finite poset P there is an element x for which

$$1-\alpha > \frac{\text{no. of ideals in } P \text{ containing } x}{\text{no. of ideals in } P} > \alpha.$$

3) [KLS] Let $G = (V, E, r)$ be a connected graph roots at r. Let the space of compatible solutions be the collection of all connected subgraphs containing r. A query is made by picking a vertex $x \in V$ and asking if it belongs to the connected subgraph. If no further assumption is made on the graph G, then the ITB may totally fail. If for example one chooses G to be C_n—the circuit on n vertices—and r to be any designated vertex, then $N_0 = O(n^2)$ is the number of connected subgraphs of G containing r. However for certain connected subgraphs, like the whole graph minus one vertex, the search will require $n-1$ queries.

However, if one assumes that all vertices in G have degree at least three, then it can be shown [KLS] that $N_0 \geqq 2^{n/4}$ and so the ITB must be good (for example, make all n possible queries). Not much is known, though, about how to find the most efficient queries and how efficient they are.

2. The problem and the main theorem. In the standard sorting problem [Kn], as everyone knows, one is given n elements $x_1, \cdots, x_n$ and one has to find a total order on them by comparing pairs x_i: x_j. The ITB implies that at least $\log_2 N_0 = \log_2 n!$ steps are required and that this bound can be more-or-less achieved. Consider now the following more general problem:

The general sorting problem. The input consists of n elements $x_1, \cdots, x_n$ together with some order relations between them. One is to discover their total order which is known to be compatible with the input order relations.

Formal restatement of the problem. Let $(P, \geqq)$ be a finite poset. There is a linear order on P compatible with $\geqq$ (an extension of $\geqq$) which is unknown to us. This extension is to be discovered by querying the order relations between pairs of elements $x, y \in P$ where x, y are unrelated by $\geqq$.

The ITB implies that any algorithm which solves this problem requires at lea $\log_2 N_0$ steps where N_0 is the number of extensions of $\geqq$. We conjecture that the IT gives the right order of magnitude. Namely, we make the following

CONJECTURE 1. *There is a universal constant $c > 1$ such that the general sorti problem can be solved in $c \log_2 N_0$ steps where N_0 is the number of extensions of $(P, \geqq$*

We want to make an even sharper conjecture asserting that one can always fin an efficient query. To this end we make the following

DEFINITION. Let $(P, \geqq)$ be a poset, $x, y \in P$.

$$\Pr(x > y) := \frac{\text{no. of extensions of } (P, \geqq) \text{ in which } x > y}{\text{no. of extensions of } (P, \geqq)}.$$

The quantities $\Pr(x > y)$ received much attention recently [Gr], [Sh], [GYY2], [KS] We want to make:

CONJECTURE 2. *There is a universal constant $\frac{1}{2} > \alpha > 0$ such that if $(P, \geqq)$ is a finite poset in which the order $\geqq$ is not total, then there exists $x, y \in P$ such that*

$$1-\alpha \geqq \Pr(x > y) \geqq \alpha.$$

In fact we know of no counterexample even for $\alpha = \frac{1}{3}$.

Now we can state and prove our main results. We can show the validity of conjectures 1 and 2 in the case where $(P, \geqq)$ can be covered by two chains. This special case is well known as the *merging problem*, see [Kn]. One is given two linearly ordered lists $A = (\alpha_1 > \cdots > \alpha_m)$ and $B = (b_1 > \cdots > b_n)$ and some order relations between elements of A and elements of B. We want to merge A and B into one ordered list where the linear order on $A \cup B$ is an extension of the partial order just described. So we have:

THEOREM 1. *Any algorithm which can merge A and B will require $\log_2 N_0$ steps in the worst case where N_0 is the number of extensions of the partial order on $A \cup B$. An algorithm exists which merges A and B in no more than $C_1 \log_2 N_0$ where $C_1 = \log_2((1+\sqrt{5})/2))^{-1}$. This bound is best possible. The computation needed for finding the appropriate queries can be done in time polynomial in $|A \cup B|$.*

THEOREM 2. *With A, B as above one can always find $x \in A$, $y \in B$ for which*

$$\tfrac{2}{3} \geqq \Pr(x > y) \geqq \tfrac{1}{3}.$$

The constants $\frac{1}{3}$, $\frac{2}{3}$ are best possible. The elements x, y can be found in time polynomial in $|A \cup B|$.

Let us start with

Proof of Theorem 2. Let us show first why $\frac{1}{3}$, $\frac{2}{3}$ are best possible. Consider the case where $A = (a_1 > \cdots > a_m)$, $B = (b_1 > \cdots > b_{2m})$, $a_j > b_{2j+1}$ $(m-1 \geqq j \geqq 1)$ and $b_{2j-2} > a_j$ $(m \geqq j \geqq 2)$. It is easily verified that

$$\Pr(a_j > b_k) = \begin{cases} 0, & k \leqq 2j-2, \\ \frac{1}{3}, & k = 2j-1, \\ \frac{2}{3}, & k = 2j, \\ 1, & k \geqq 2j+1, \end{cases} \quad (m \geqq j \geqq 1,\ 2m \geqq k \geqq 1).$$

Now let us turn to the proof of the existence of $x \in A$, $y \in B$ for which $\frac{2}{3} \geqq \Pr(x > y) \geqq \frac{1}{3}$. We may assume w.l.o.g. that a_1 and b_1 are incomparable. If $a_1 > b_1$, then a_1 is the unique maximal element in $A \cup B$ and so it remains the maximal element in any extension of the partial order. Therefore, nothing will change if a_1 is deleted from the poset. We prove our claim by contradiction and we assume again w.l.o.g. that

$$\Pr(a_1 > b_1) < \tfrac{1}{3}.$$

Define now the following quantities

$$q_1 = \Pr(a_1 > b_1),$$
$$q_i = \Pr(b_{i-1} > a_1 > b_i)(n \geqq i \geqq 2),$$
$$q_{n+1} = \Pr(b_n > a_1).$$

We prove the following:

LEMMA. *The real numbers* $q_i(n+1 \geqq i \geqq 1)$ *satisfy*:

1) $\frac{1}{3} \geqq q_1 \geqq \cdots \geqq q_{n+1} \geqq 0$,
2) $\sum_{i=1}^{n+1} q_i = 1$.

Proof. Since $q_1, \cdots, q_{n+1}$ is a probability distribution, all we have to show is that $q_1 \geqq \cdots \geqq q_{n+1}$. To show this we exhibit a 1:1 mapping from the event whose probability is q_{i+1} into the event with probability $q_i(1 \geqq i \geqq n)$. Notice that in an extension for which $b_{i-1} > a_1 > b_i$ not only does a_1 come after b_{i-1} but it must immediately follow it: Of course none of the a_j can precede a_1 and none of the b_j can come between b_{i-1} and b_i. The mapping from those extensions in which a_1 immediately follows b_i to those where $b_{i-1} > a_1 > b_i$ is obtained by permuting a_1 and b_{i-1}. This mapping clearly is well defined and 1:1.

The theorem can be proved now: let r be defined by

$$\sum_{i=1}^{r-1} q_i \leqq \tfrac{1}{2} < \sum_{i=1}^{r} q_i.$$

Since $\sum_{i=1}^{r-1} q_i = \Pr(a_1 > b_{r-1}) \leqq \frac{1}{2}$, it follows that $\sum_{i=1}^{r-1} q_i < \frac{1}{3}$. Similarly $\sum_{i=1}^{r} q_i = \Pr(a_1 > b_r)$ must be $> \frac{2}{3}$. Therefore $q_r > \frac{1}{3}$, but this contradicts $\frac{1}{3} > q_1 \geqq q_r$.

Complexity. The last claim of the theorem reduces now to proving that the index r of the above proof can be found in time which is polynomial in $|A \cup B|$. The reader should be aware that two separate complexity measures are being considered: the main one is a count of the number of queries that have to be asked in order to solve the merging problem, and the other one, which we address now, is the time complexity of the computations which are required to design the queries. Given a partially ordered set on n elements $(P, \geqq)$ which can be covered by two chains, there is a determinant formula giving the number of extensions of $\geqq$, see [Mo, p. 32]. Since these determinants are computable in polynomial time and we need to compute polynomially many such determinants to implement our algorithm, this proves our assertion. For completeness, let us recite the determinant counting formula: Let $P = A \cup B$, where $A = (a_1 > \cdots > a_m)$, $B = (b_1 > \cdots > b_n)$, and assume $m \geqq n$. Define integers $\alpha_1, \cdots, \alpha_m, \beta_1, \cdots, \beta_m$ as follows: $\beta_i = \min\{t \mid a_i < b_t\}$, $\alpha_j = \max\{t \mid b_t > a_j\}$ and where the minimum and maximum of an empty set are taken to be $n+1$ and zero respectively. The number of extensions of $(P, \geqq)$ is given by

$$\det\left[\binom{\beta_i - \alpha_j + 1}{j - i + 1}\right]_{m \geqq i, j \geqq 1};$$

see [Mo] for the details. □

The following theorem is equivalent with Theorem 1 but states the result in a more convenient way. We remind the reader about the definition of Fibonacci numbers: This is the sequence defined by: $F_0 = 1$, $F_1 = 2$, $F_{n+1} = F_n + F_{n-1}(n \geqq 1)$. The following explicit formula also exists for these integers

$$F_n = A\lambda^n + B\cdot(-\lambda)^{-n},$$

where

$$\lambda = \frac{\sqrt{5}+1}{2}, \quad A = \frac{5+3\sqrt{5}}{10}, \quad B = \frac{5-3\sqrt{5}}{10}.$$

THEOREM 1.1. *A merging problem which cannot be solved by less than n querie must have at least F_n compatible solutions. For each $n \geqq 1$ there exists a unique mergin problem which requires n queries and has exactly F_n compatible solutions The appropriat queries can be found in time polynomial in the size of the poset.*

Proof. Let us start by exhibiting the extreme cases. We describe the merging problems which are referred to as the *special merging problems. For* $n = 2m-1$, *le* $A = (a_1 > \cdots > a_m)$, $B = (b_1 > \cdots > b_m)$ *and the relations* $a_j > b_{j+1}(m-1 \geqq j \geqq 1)$ $b_k > a_{k+2}(m-2 \geqq k \geqq 1)$. For $n = 2m$ let $A = (a_1 > \cdots > a_{m+1})$, $B = (b_1 > \cdots > b_m$ and $a_j > b_{j+1}(m-1 \geqq j \geqq 1)$, $b_k > a_{k+2}(m-1 \geqq k \geqq 1)$. In either case a_j is incomparabl with only b_{j-1} and b_j. Whenever $a_j : b_j$ are compared, the answer is $a_j > b_j$ and th answer on $a_j : b_{j-1}$ is $b_{j-1} > a_j$. These answers supply no further information on incomparable pairs: therefore all n queries have to be made to solve these merging problems To show that the number of compatible solutions in these merging problems are give by Fibonacci numbers, let us consider the case $n = 2m$. We split the compatible solution into two parts according to whether $a_1 > b_1$ or $b_1 > a_1$. If $a_1 > b_1$, then a_1 is the uniqu maximal element and so can be deleted altogether. For the rest of the elements w make the following renaming $b_i' = a_{i+1}(i = 1, \cdots, m)a_i' = b_i(i = 1, \cdots, m)$ which show that the remaining problem is the special problem for $n = 2m-1$. If $a_1 < b_1$, then a_1, b_1 are the maximal elements of the poset so they can be deleted. The remaining problem is again the special one for $n = 2m-2$. We have thus shown that $F_n = F_{n-1} + F_{n-2}$ fo even $n \geqq 2$. The rest of the details can be easily filled in by the reader.

Now we turn to the actual proof of the theorem and of the uniqueness of the special problems: We'll show that if a merging problem is given with $N_0 \leqq F_n$ compatible solution and n steps are needed to solve it, then the problem is special. For $n \leqq 3$ the cases are few and can be checked each in itself. The general case is done by induction on n. Without loss of generality we assume that $q_1 = \Pr(a_1 > b_1) \leqq \frac{1}{2}$. As in the lemma we define q_i to be $\Pr(b_{i-1} > a_1 > b_i)(m+1 \geqq i \geqq 1)$. Consider the index r for which

$$\Pr(a_1 > b_{r-1}) = \sum_{i=1}^{r-1} q_i \leqq \tfrac{1}{2} < \sum_{i=1}^{r} q_i = \Pr(a_1 > b_r).$$

If $\Pr(a_1 > b_r) < F_{n-1}/F_n$, then comparing $a_1 : b_r$ we remain with a problem which has less than F_{n-1} compatible solutions and so can be solved in $n-2$ steps, contradiction. Similarly if $\Pr(a_1 > b_{r-1}) > F_{n-2}/F_n$, then on comparing $a_1 : b_{r-1}$ we remain with a problem with less than $F_n - F_{n-2} = F_{n-1}$ compatible solutions and the same argument applies. If follows, therefore, that $q_r \geqq F_{n-1}/F_n - F_{n-2}/F_n = F_{n-3}/F_n$. This implies now that $r \leqq 2$, because otherwise

$$\frac{F_{n-2}}{F_n} \geqq \sum_{1}^{r-1} q_i \geqq q_1 + q_2 \geqq 2q_r \geqq \frac{2F_{n-3}}{F_n},$$

a contradiction if $n \geqq 4$. On the other hand $r \neq 1$ because, by assumption $q_1 = \Pr(a_1 > b_1) \leqq \frac{1}{2}$.

So $r = 2$, $q_1 \leqq F_{n-2}/F_n$, $q_1 + q_2 \geqq F_{n-1}/F_n$. Make the comparison $a_1 : b_2$, to which we may assume the answer is $a_1 > b_2$. This is followed by the comparison $a_1 : b_1$ to which we may assume a reply $a_1 > b_1$. The remaining problem has at most F_{n-2} compatible solutions and so can either be solved in $n-3$ queries making up a total of $n-1$ queries for the original problem, or else it is the special problem with F_{n-2} compatible solutions. One has to verify now that the problem we started with is special. This is an easy fact to verify and the details are omitted. The complexity argument is the same as in Theorem 2. □

3. Open problems. The major problem is, of course, to show that Theorems 1 and 2 hold for general sorting problems. These problems were stated above in conjectures 1, 2. To state other problems let us make the following definition: An extension of a partial order $(P, \geqq)$ can be described as 1:1 order-preserving map $\sigma: P \to \{1, \cdots, |P|\}$. For $x \in P$ we define $h(x)$ to be the average of $\sigma(x)$ over all extensions σ of $(P, \geqq)$. Let $|P| = n$ be the order of the poset, then $\sum_{x\in P} h(x) = n(n+1)/2$. We define the "second moment" of h as $V(P) = \sum_{x\in P} h^2(x)$. If $p, q \in P$ are incomparable elements, then denote by $P(p, q)$ the poset which is obtained by adding the relation $p > q$ to P (and, of course, taking transitive closure of the new relation). We have:

THEOREM 3. *Let P be a poset, $p, q \in P$ incomparable elements. Then*

$$V(P) \leqq \max\{V(P(p, q)), V(P(q, p))\}.$$

Proof. The most convenient way to view this inequality is geometrically: To any poset $(P, \geqq)$ we canonically assign an n-dimensional convex polyhedron $C(P)$ where $|P| = n$. The assignment is as follows: If P has no order relations, then $C(P)$ is the unit cube $\{(x_1, \cdots, x_n) | 1 \geqq x_i \geqq 0\}$. Let us say that $C(P)$ has been defined for posets with k order relations or less ($k \geqq 0$). Then on introducing the new relation $p_i > p_j$ the convex polytope of the new poset $P(p_i, p_j)$, namely $C(P(p_i, p_j))$, is obtained by taking that part of $C(P)$ which lies in the half-space $x_i > x_j$. Accordingly, for P which is totally ordered, $C(P)$ is a simplex $x_{\pi(1)} < x_{\pi(2)} < \cdots < x_{\pi(n)}$. Notice that these simplices have volume $1/n!$ each, and that if $(P, \geqq)$ is any partial order on $P = \{p_1, \cdots, p_n\}$, then there is a 1:1 correspondence between the extensions of $(P, \geqq)$ and the simplices that make up $C(P)$. In particular the volume of $C(P)$ equals $1/n!$ times the number of extensions of $(P, \geqq)$. Notice also that since all these simplices have equal volume, $\bar{h}(P) = 1/(n+1)(h(p_1), \cdots, h(p_n))$ is the center of gravity of $C(P)$. If follows that $V(P)$ is the square of the distance from the center of gravity of $C(P)$ to the origin. Now that we have established the geometric interpretation of $V(P)$, the validity of the theorem follows at once: $C(P)$ is the disjoint union of $C(P(p_i, p_j))$ and $C(P(p_j, p_i))$. Therefore the origin and the centers of gravity for $C(P(p_i, p_j))$ and $C(P(p_j, p_i))$ form a triangle and the center of gravity of $C(P)$ lies on the edge connecting the two centers of gravity. The theorem now follows from obvious facts of plane geometry. □

Now that we have established Theorem 3, we are ready to ask if a stronger statement holds.

CONJECTURE 3. *Let P be a poset and let $p, q \in P$ be incomparable. Then*

$$V(P(p, q)) \geqq V(P).$$

See the problem session of [OS, p. 806] for a related discussion.

Note added in proof. Problem 1 has been recently answered affirmatively by the author and M. Saks. The constant that was found is $\alpha = \frac{1}{4}(3 - \log_2 5)$.

An interesting problem in computational complexity is to show that it is hard to count the number of linear extensions of a finite poset. We conjecture that this problem is #P-complete. This conjecture has apparently been made also by R. Karp and by some other researchers. Using the construction made in the proof of Theorem 3, this

conjecture could be a concrete statement to the effect that evaluating the volume [of] polyhedra is a hard computational problem.

Also, counting the number of order ideals in posets can be shown to [be] #P-complete. This was shown also by R. Karp (private communication, March 19[...]

It has been brought to our attention that Conjecture 2 has been independe[ntly] made by a number of researchers, some time ago. In particular we know that M. Fr[ed]man and R. Stanley had thought about it.

REFERENCES

[Fr] M. FREDMAN, *How good is the information theory bound in sorting*, Theoret. Comput. Sc[i.] (1976), pp. 355–361.

[Gr] R. L. GRAHAM, *Linear extensions of partial orders and the FKG inequality*, in [OS], pp. 2[...] 236.

[GYY1] R. L. GRAHAM, A. C. YAO AND F. F. YAO, *Information bounds are weak in the sho[rtest] distance problem*, J. Assoc. Comput. Mach., 27 (1980), pp. 428–444.

[GYY2] ———, *Some monotonicity properties of partial orders*, SIAM J. Alg. Disc. Meth., 1 (19[80],) pp. 251–258.

[Kn] D. E. KNUTH, *The Art of Computer Programming, Vol. 3, Sorting and Searching*, Addis[on-]Wesley, Reading, MA, 1973.

[KS] D. J. KLEITMAN AND J. B. SHEARER, *A monotonicity property of partial order*, to appear.

[KLS] D. J. KLEITMAN, N. LINIAL AND D. STURTEVANT, *On extremal spanning trees*, unpublis[hed] manuscript.

[LS] N. LINIAL AND M. SAKS, *Searching ordered structures*, J. Algorithms, to appear.

[Mo] S. G. MOHANTY, *Lattice Path Counting and Applications*, Academic Press, New York, 19[...]

[OS] *Ordered Sets*, I. Rival, ed., NATO Advanced Study ser. C. vol. 83, 1981.

[Sa] B. SANDS, *Counting antichains in finite partially ordered sets*, Discrete Math., 35 (19[...]) pp. 213–228.

[Sh] L. A. SHEPP, *The FKG inequality and some monotonicity properties of partial orders*, SIA[M J.] Alg. Disc. Meth., 1 (1980), pp. 295–299.