Progress in Selection[⋆]

Mike Paterson

Department of Computer Science, University of Warwick, Coventry CV4 7AL, UK

Abstract. There has been recent progress in the selection problem, and in median-finding in particular, after a lull of ten years. This paper reviews some ancient and modern results on this problem, and suggests possibilities for future research.

1 Introduction

The *selection problem*, determining the k^{th} largest out of a set of n elements, is a junior partner of the more fundamental sorting problem, but it has still been studied extensively over several decades.

Our focus will be on performing selection using a minimal number of comparisons in the worst case. Let $V_k(n)$ be the worst-case minimum number of pairwise comparisons required to find the k^{th} largest out of n distinct elements. Of particular interest is finding the *median*, the $\lceil n/2 \rceil^{\text{th}}$ largest say. We denote $V_{\lceil n/2 \rceil}(n)$ by $M(n)$.

The worst-case comparison complexity of sorting is $n \log_2 n + O(n)$, and even the coefficient of the linear term has been fairly closely estimated. However for $V_k(n)$ we do not yet have an asymptotic value, except when $k = o(n)$, (and symmetrically, for $k - n = o(n)$). For finding the median, we currently know only a broad interval for the value of $M(n)$.

In this survey, I shall review some of the classic results in the quest to determine $V_k(n)$ and $M(n)$, report on some recent progress, and conjecture an asymptotic value for $M(n)$.

2 History

Credit for first raising the selection problem is often accorded to Charles Dodgson, who considered the proper allocation of the second and third prizes in tennis tournaments. Steinhaus proposed the problem of finding $V_2(n)$. The upper bound of $n + \lceil \log_2 n \rceil - 2$ was given by Schreier [22], but this was not shown to be the exact value until the proof by Kislitsyn [15]. Hadian and Sobel [10] gave an upper bound: $V_k(n) \leq n - k + (k-1)\lceil \log_2(n-k+2) \rceil$. This bound is asymptotically optimal for fixed k. A good account of early work in this area can be found in [16]. Successive improvements, for various ranges of k with respect to n, were

[⋆] This research was supported in part by the EU under contract 20244 (ALCOM-IT).

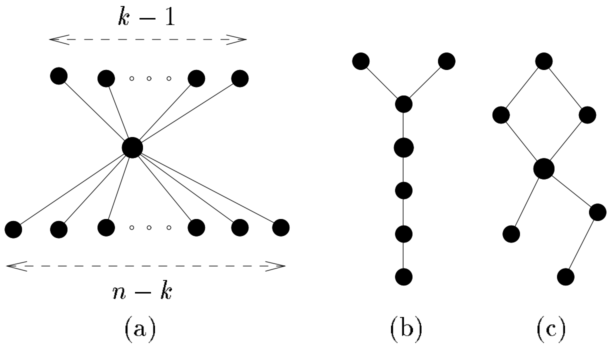

Fig. 1. (a) S^{k-1}_{n-k}; (b) and (c) both contain S^3_3.

made by Kirkpatrick [13, 14], Yap [25], Hyafil [11], Motoki [17], and Ramanan and Hyafil [20].

The classic paper by Blum, Floyd, Pratt, Rivest and Tarjan [2] in 1973 was the first to show that $M(n) = O(n)$, and therefore that finding the median is much easier than sorting. They gave an algorithm which requires at most about $5.43n$ comparisons, and introduced a technique which has been a basis of all subsequent improvements. In Section 3, I shall outline this technique, the improvements to $3n$ by Schönhage, Paterson and Pippenger [21], and the very recent further improvements by Dor and Zwick [5, 6, 7].

Blum et al. [2] were also the first to give a non-trivial lower bound for $M(n)$, and $V_k(n)$ when $k = \Omega(n)$. They showed that $M(n) \geq 3n/2 - O(1)$ and, more generally, that $V_k(n) \geq n + \min\{k, n-k\} - O(1)$, by using a simple adversary argument. This lower bound was successively improved by several authors (see [11, 13, 25, 18]), using more and more sophisticated adversaries and accounting schemes, and the coefficient was raised closer to 2.

A breakthrough came in 1985, with an elegant lower bound of $2n - o(n)$ by Bent and John [1]. It has taken a further ten years for this to be improved, by Dor and Zwick (again!) [5, 8]. In Section 4, I will review the adversary argument of Blum et al., and the use of a multitude of adversaries by Bent and John. I will also describe the subtle improvement in [5, 8].

Notation

We shall use the natural Hasse diagrams to describe partial orders. In our figures, the larger elements will be towards the top of the page. In the terminology of [21], a partial order which is composed of a *centre* element c together with u elements larger than c and v elements smaller than c is referred to as an S^u_v. The problem of selecting the k^{th} largest element corresponds to constructing some S^{k-1}_{n-k} from a given set of n elements, i.e., determining some partial order which contains S^{k-1}_{n-k}. (See Figure 1(a).) For example, the partial orders in Figure 1(b),(c) both yield the median of seven elements.

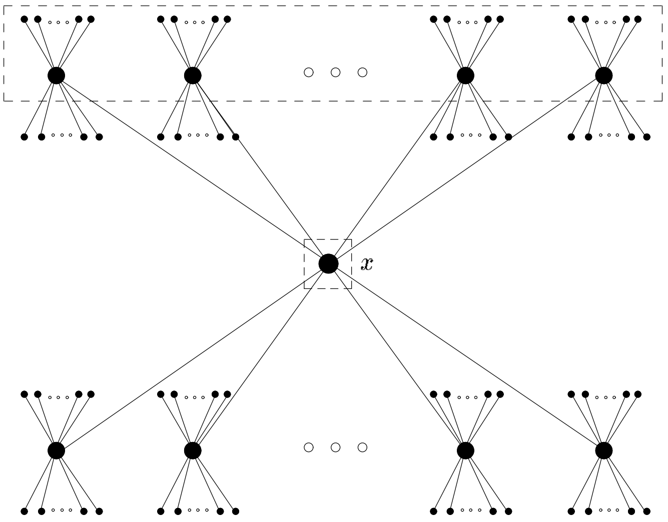

Fig. 2. Median of medians

3 Upper bounds

The general algorithmic technique introduced by Blum et al. is to generate many S_v^v's, for suitable v, and then to find the median x of the set of centre elements of the S_v^v's. The resulting partial order contains the partial order shown in Figure 2. The ranking of the element x with respect to the whole set is determined next. Suppose we are seeking the k^{th} largest element. If the rank of x is k then we are finished. If not, then suppose without loss of generality that the rank of x is less than k. In this case, x and all the elements above x in the partial order (outlined in dashed boxes in Figure 2) may be discarded. These constitute more than a quarter of the elements. The algorithm proceeds with the appropriate selection problem from the remaining set. If the algorithm requires only a linear number of comparisons to discard a linear number of elements then a linear upper bound is assured. Blum et al. found it convenient to use C_{2v+1}'s, sorted chains of length $2v + 1$, as the S_v^v's in their algorithm and took v to be a small constant, seven. Finding the median of the centre elements was done by using the algorithm recursively. With some careful optimisation, they achieved an upper bound of $391n/72 \sim 5.43n$.

Note that, in this algorithm, a set of elements which is discarded consists of the top or bottom $v + 1$ elements from several C_{2v+1}'s, and they leave behind a residue of disconnected C_v's. Crucial to the performance of the algorithm is how economically the C_{2v+1}'s can be produced and how efficiently the returned C_v's can be recycled into new C_{2v+1}'s.

The balance of parameters used by Schönhage et al. [21] was very different. We took v to be rather large, so that the number of S_v^v's at any stage was $O(n^{3/4})$. The centres of the S_v^v's could therefore be kept in sorted order, and the

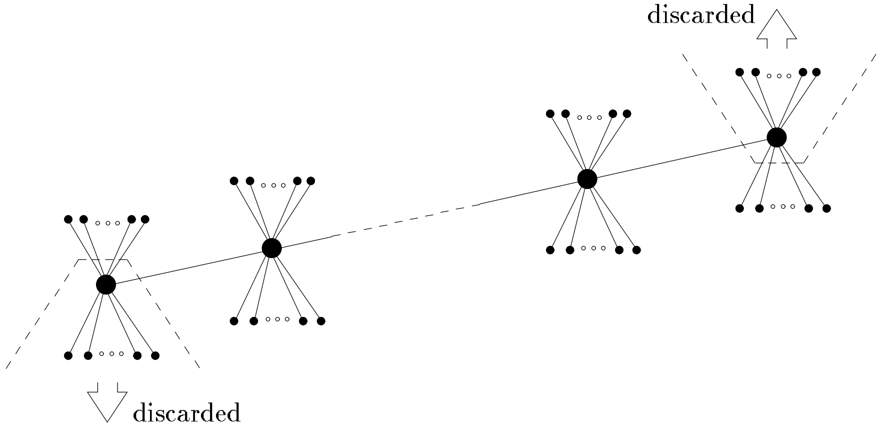

Fig. 3. Overall picture for SPP process.

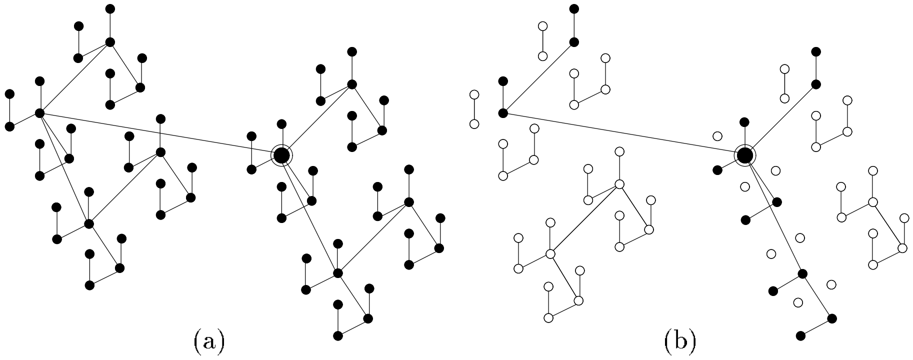

Fig. 4. (a) Hyperpair H_{011010}; (b) an S_7^7 produced by pruning

main part of the partial order maintained during the algorithm had the form shown in Figure 3.

The principal difference from the algorithm of Blum et al. was that whereas they produced S_v^v's (actually C_{2v+1}'s) in batches, we used a continuous production line. Much of the efficiency of our S_v^v factory arose from the nice properties of *hyperpairs*. Hyperpairs are built up recursively from smaller hyperpairs using symmetric comparisons. An illustration of the hyperpair H_{011010} is given in Figure 4(a). In (b) it is shown how an H_{011010} can be pruned to produce an S_7^7. Note also that the pieces pruned off are all in the form of smaller hyperpairs of various sizes. These are retained in the factory for recycling.

When the top or bottom parts of hyperpairs together with the central element are discarded during the algorithm, the remaining halves are returned to the factory for recycling. Unfortunately these are not themselves hyperpairs, so they are broken down to pairs and singletons, with a consequent loss in efficiency.

Dor and Zwick [5, 6, 7] introduce their *green* factories, which are able to recycle efficiently more of the debris from discarded S_v^v's. Instead of just pairs and singletons, they can recycle several larger partial order fragments. In [21], we describe a process where pairs and singletons are "grafted" onto structures to build S_v^v's more economically. Dor and Zwick show how to graft with a variety of different components, including even some 16-tuples, and they generalise hyperpairs to *hyperproducts*.

As a result of their many ingenious improvements, they achieve a median algorithm which requires at most $2.9423n$ comparisons. They also extend these techniques to the general selection problem to obtain improved bounds for $V_k(n)$.

4 Lower bounds

Many lower bound proofs for combinatorial algorithms such as selection can be usefully described in terms of an adversary, who "plays against" the algorithm by providing answers to the comparisons performed, in such a way that a fairly bad case is forced. The lower bound of $n + \min\{k-1, n-k\} - 1$ given by Blum et al. [2] was proved in such a way.

Their adversary also provides some extra information to the algorithm from time to time, which may even assist the algorithm, but is chiefly provided to keep the state of the computation simple enough to deal with. As soon as any element has been declared the larger in two comparisons, the adversary tells the algorithm that this element is actually larger than the element being sought and so there is no need to carry out any further comparisons involving this element. Elements which are declared smaller in two comparisons are similarly eliminated. The partial order on the remaining candidates for selection consists always of only pairs and singletons. It is easy for the adversary to choose an outcome for each comparison involving a pair so that one element is eliminated, two comparisons are removed, and the simple form of the partial order is restored.

The adversary can continue with this "generous" strategy until either $k-1$ elements have been declared too large or $n-k$ have been declared too small. At this point, the problem has been reduced to finding the maximum or minimum, respectively, of elements in the remaining r pairs or singletons. The adversary now keeps quiet and $r-1$ further comparisons are required. It can be seen that at least $2\min\{k-1, n-k\} + \max\{k-1, n-k\}$ comparisons are required overall, i.e., $n + \min\{k-1, n-k\} - 1$ comparisons.

A sequence of improvements on this lower bound [19, 13, 25, 18] took the coefficient for medians from $3/2$ up to about 1.837. The new proofs introduced many new ideas and became more and more intricate, but were still based on a similar type of adversary argument. The components of the partial order retained at any time became larger and more varied: in [18] several hundred cases needed to be verified. Elaborate accounting systems were employed, e.g., "safe-boxes" and "joint accounts" in [25].

A different bounding method was introduced by Fussenegger and Gabow [9]. Instead of directly considering the depth of the decision tree of an algorithm, they count the number of leaves. The power of the method comes from showing that the decision tree has to contain many binary trees, each with a large number of leaves. If the leaves of the different included trees are disjoint, or at least do not overlap too much, then the decision tree must have many leaves and hence large depth. Although they were able to raise the lower bounds for some ranges of $V_k(n)$ and for other similar problems, they did not improve the lower bound in [2] for median-finding. A major improvement was made by Bent and John [1] in 1985, using this "leaf-counting" argument.

Any algorithm requires $j-1$ comparisons to find the largest (or the smallest) element from a set of size j. The corresponding decision tree will have at least 2^{j-1} leaves. Also, any algorithm for selecting the k^{th} largest element x from a set of size n will need at least $n-1$ comparisons either involving x or between pairs of elements which are both above or both below x. A comparison between an element above x and an element below x will be called a *straddle.* Note that straddles do not contribute to the Hasse diagram of the final partial order for selection. So, to prove a lower bound of $n+m$, it would suffice to show that more than m straddles are sometimes made.

Bent and John use a multitude of adversaries in their lower bound proof for $V_k(n)$. Each adversary has its own special subset of elements A, where $|A| = k$. Up to a point, each adversary will answer comparison queries by using the rule that each element of A is above each element of the complement $\overline{A}$, i.e., $A > \overline{A}$. For other queries, of the form $A : A$ and $\overline{A} : \overline{A}$, both answers are valid and we regard the adversary as laying out a tree which branches at each such query. Unfortunately such simple adversaries are not strong enough to force a good bound – they need to be slightly malicious! When the algorithm is getting close to its goal of determining the minimum element of A, to return as the k^{th} largest, the adversary is allowed to shift the goal-posts.

I will outline the proof of a lower bound which depends on some suitable parameter q. The full strategy for the adversary with special set A is described below. At any stage let $\mathrm{Min}A$ be the set of minimal elements in the partial order restricted to A, and let $\mathrm{Max}\overline{A}$ be the set of maximal elements in the partial order on $\overline{A}$.

Phase 1. Answer comparison queries according to the partial order $A > \overline{A}$. For comparisons of the type $A : A$ or $\overline{A} : \overline{A}$, follow both answers, creating a branching tree. Continue Phase 1 until $|\mathrm{Min}A| = q$. Now if $|\mathrm{Max}\overline{A}| < 2q$ then continue with Phase 2a, otherwise use Phase 2b.

Phase 2a. Continue as in Phase 1 until the end, when $|\mathrm{Min}A| = 1$.

Phase 2b. Choose an element $y \in \mathrm{Min}A$ such that the set $B(y)$ of elements in $\mathrm{Max}\overline{A}$ which have been compared with y is as small as possible. Let $A' = A\backslash\{y\}$, and answer the remaining comparison queries using the partial order $A' > \overline{A'}$.

Phase 1 requires $k-|\mathrm{Min}A|+n-k-|\mathrm{Max}\overline{A}|$ branching comparisons. Phase 2a requires a further $|\mathrm{Min}A|-1$ of these, giving a total of $n-|\mathrm{Max}\overline{A}|-1 \geq n-2q$.

For Phase 2b, the algorithm must find the maximum of the set $\overline{A'}$ in order to satisfy the adversary's strategy. At the beginning of Phase 2b, the maximal elements of $\overline{A'}$ are $y \cup (\mathrm{Max}\overline{A} \setminus B(y))$, and so Phase 2b requires at least $|\mathrm{Max}\overline{A}| - |B(y)|$ comparisons, giving a total of at least $n - q - |B(y)|$ comparisons. But if $|B(y)| > q$ then, by the choice of y, every element of $\mathrm{Min}A$ has been compared with at least $q + 1$ elements of $\mathrm{Max}\overline{A}$ by the end of Phase 1. If y were to be chosen as the k^{th} largest element then the above comparisons already account for at least $(q-1)(q+1)$ straddles.

In summary, either there is a run of the algorithm producing $q^2 - 1$ straddles, and so at least $n + q^2 - 2$ comparisons, or else every adversary generates a binary tree with at least 2^{n-2q} leaves. Any leaf corresponds to a choice of a set A' for the $k - 1$ largest elements (together with a choice for the k^{th} largest) and can be reached only by an adversary whose set A contains A', i.e., at most $n - k + 1$ of them. The number of different adversaries is $\binom{n}{k}$, so the decision tree must have at least $2^{n-2q}\binom{n}{k}/(n-k+1)$ leaves. The choice $q = \sqrt{n}$ gives the bound

$$V_k(n) \geq n + \log \binom{n}{k} - O(\sqrt{n}).$$

In particular, Bent and John [1] prove $M(n) \geq 2n - O(\sqrt{n})$.

Dor and Zwick [5, 8] use a delicate argument to increase these lower bounds very slightly. They prove that $M(n) \geq (2+\epsilon)n + o(n)$ for some $\epsilon > 2^{-40}$. Their proof follows the "leaf-counting" style of [9, 1]. Although they are not able to prove that the decision tree contains more leaves than did Bent and John [1], they can show that the tree is not balanced, so that some paths are longer than others. There is now a single powerful adversary which summarises the combined effect of the multitude of simple adversaries introduced above. Dor and Zwick look at the earlier stages of an algorithm when there are still many singletons remaining, and focus on the first and second comparisons in which each element is involved. They show by a careful analysis that in many comparisons the adversary can choose an outcome leading to a significantly larger number of leaves than the alternative. At times this adversary discards some of the sets A which it has been monitoring, and at other times it chooses an outcome leading to the smaller number of leaves. Overall however it comes out a little bit ahead using its subtle strategy. This gives the payoff of ϵ.

5 Future progress

For ten years, the upper and lower bound coefficients for $M(n)$ were sitting sedately at integers, with neither showing much inclination to approach the other. Now, thanks to Dor and Zwick [5, 6, 8], both coefficients have been given a nudge. Where are they heading? Which of them is nearer to the true value? Will the coefficient be semi-natural, such as 2.5, or rather unnatural, such as $\log_{4/3} 2$? The intricacy of the arguments in [5, 6, 8] makes it appear that there is no easy way to take large further steps without some substantially new approach.

Yao's Hypothesis

In [23], Frances Yao considered the problem of finding a (u, v)-*mediocre element*,

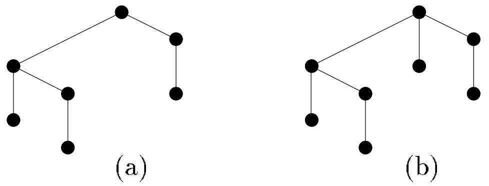

Fig. 5. Counter-example to generalised Yao's Hypothesis

from m elements, i.e., an element which is smaller than at least u elements and larger than at least v elements. This corresponds to constructing an S_v^u from m elements, where $m \geq u + v + 1$.

She defined $S(u, v, m)$ to be the worst-case minimum number of comparisons needed to find a (u, v)-mediocre element from a set of size m. Obviously, $S(u, v, m) \geq S(u, v, m')$ for $u + v + 1 \leq m \leq m'$. We have $V_k(n) = S(k-1, n-k, n)$. Let $V_k^*(n) = \lim_{m \to \infty} S(k-1, n-k, m)$, i.e., the cost of producing an S_{n-k}^{k-1} from arbitrarily many elements.

Yao observed that $V_1(n) = V_1^*(n)$ for all n. We define as *Yao's Hypothesis* the equation:

$$V_k(n) = V_k^*(n) \text{ for all } k, n. \tag{YH}$$

Yao proved in [23] that YH implies that $M(n) \leq 3n + o(n)$. An analogous proof in [21] showed that YH implies $M(n) \leq 2.5n + o(n)$.

To date, no counter-example to Yao's Hypothesis is known. Since the consequences of YH are so rewarding, it would be of great interest to resolve its truth.

The generalisation of YH from S_v^u's to arbitrary partial orders is known not to hold. The partial order (a) can be shown by an exhaustive analysis to require 8 comparisons when only 7 elements are present, whereas that shown in (b) requires only the obvious 7 comparisons.

Information theory methods

Information theoretic techniques provide a powerful tool to prove lower bounds for sorting and selection problems. In a typical application, a partial order π is assigned the weight $w(\pi)$, the number of its linear extensions, i.e., the number of total orders consistent with π. A further comparison $a : b$ yields one of the two extensions $(\pi \cup [a > b])$ or $(\pi \cup [a < b])$. Since $w(\pi \cup [a > b]) + w(\pi \cup [a < b]) = w(\pi)$, simple arguments show that the worst-case number of comparisons needed to sort a partial order π is at least $\log_2 w(\pi)$. Unfortunately, this technique is too weak for deriving useful bounds for selection. Since an S_v^u has weight $u!v!$, the information theoretic lower bound for $V_k(n)$ is only $\log_2(\binom{n}{k}k)$, which is at most $n + O(\log n)$.

For a variation on this technique which is more suitable for selection problems, we consider "bipartitions". A *bipartition* of the partial order π on the set X is a mapping g of the elements of X into $\{0, 1\}$ which is compatible with π, i.e., for all $x, y \in X$, if $x <_\pi y$ then $g(x) \leq g(y)$. Thus a bipartition of π is a partition of X into two sets $g^{-1}(1)$, the *up-set*, and $g^{-1}(0)$, the *down-set*, such that no element of the up-set is below an element of the down-set in π.

6 Partition measures

Let $P(\pi)$ be the set of bipartitions of π, and $p(\pi) = |P(\pi)|$. We will use the notation $P(\pi, a/b)$ for the set of bipartitions g of π such that $g(a) = 1$ and $g(b) = 0$, and similarly the notations $P(\pi, b/a)$, $P(\pi, ab/)$, and $P(\pi, /ab)$ for the other three cases where $(g(a), g(b)) = (0,1), (1,1), (0,0)$, respectively. The cardinalities of these sets are represented analogously using "p" in place of "P". Note that $p(\pi) = p(\pi, a/b) + p(\pi, b/a) + p(\pi, ab/) + p(\pi, /ab)$.

The effect of a comparison is not now to give a splitting of $P(\pi)$ into two disjoint sets, as it was for linear extensions. Instead, we find

$$P(\pi \cup [a > b]) = P(\pi, a/b) \cup P(\pi, ab/) \cup P(\pi, /ab),$$

and

$$P(\pi \cup [a < b]) = P(\pi, b/a) \cup P(\pi, ab/) \cup P(\pi, /ab).$$

This has the advantage for lower bounding that we can hope that p decreases more slowly per comparison than did w. Indeed the following theorem holds.

Theorem 1. *For any partial order π, and elements $a, b \in X$,*

$$\max\{p(\pi \cup [a > b]),\ p(\pi \cup [a < b])\ \} \geq \frac{3}{4}\ p(\pi).$$

Proof. For convenience, we denote $p(\pi), p(\pi, a/b), p(\pi, b/a), p(\pi, ab/), p(\pi, /ab)$ by $p, p_{10}, p_{01}, p_{11}, p_{00}$, respectively. The theorem is equivalent to showing that $\min\{p_{01}, p_{10}\} \leq p/4$.

I will prove below that $p_{10}p_{01} \leq p_{11}p_{00}$. The theorem follows quickly, since

$$\begin{aligned}(\sqrt{p_{01}} + \sqrt{p_{10}})^2 &= p_{01} + p_{10} + 2\sqrt{p_{10}p_{01}} \\ &\leq p_{01} + p_{10} + 2\sqrt{p_{11}p_{00}}, \quad \text{since } p_{10}p_{01} \leq p_{11}p_{00}, \\ &\leq p_{01} + p_{10} + p_{11} + p_{00}, \\ &\qquad \text{since the geometric mean is at most the arithmetic mean,} \\ &= p.\end{aligned}$$

Hence, $\min\{\sqrt{p_{01}}, \sqrt{p_{10}}\} \leq \frac{1}{2}\left(\sqrt{p_{01}} + \sqrt{p_{10}}\right) \leq \frac{1}{2}\sqrt{p}$, i.e., $\min\{p_{01}, p_{10}\} \leq \frac{1}{4}p$.

Finally, to prove that $p_{10}p_{01} \leq p_{11}p_{00}$, I give an explicit injective map F from $P(\pi, a/b) \times P(\pi, b/a)$ into $P(\pi, ab/) \times P(\pi, /ab)$. Any $h \in P(\pi) \times P(\pi)$ partitions X into four subsets $H_{i,j}$, where $i, j \in \{0, 1\}$, where $H_{i,j} = \{x \in X : h(x) = (i, j)\}$. These sets have the property that if $x \in H_{10}$ and $y \in H_{01}$ then x and y are unrelated in π. Note that if $h \in P(\pi, a/b) \times P(\pi, b/a)$ then $a \in H_{10}$ and $b \in H_{01}$, while for $h \in P(\pi, ab/) \times P(\pi, /ab)$ we have $a \in H_{10}$ and $b \in H_{10}$.

We regard the partial order as a directed graph on X, and h assigns labels $10, 01, 11, 00$ to the vertices. Vertex a has label 10, and b has label 01. Let

C_b be that weakly connected component of the vertices labelled 01 which contains b. The map $F : P(\pi, a/b) \times P(\pi, b/a) \to P(\pi, ab/) \times P(\pi, /ab)$ is defined by specifying the four sets $H'_{10}, H'_{01}, H'_{11}, H'_{00}$ corresponding to $h' = F(h)$. Let $H'_{10} = H_{10} \cup C_b$, $H'_{01} = H_{01} \setminus C_b$, $H'_{11} = H_{11}$, and $H'_{00} = H_{00}$. With respect to h', C_b is now the weakly connected component of the vertices labelled 10 which contains b. Therefore the mapping F is invertible and so injective, as required to complete the proof. □

Some extensive generalisations of this result have been given by J. Daykin [3]. (See also [4] for related results.)

Under this new measure for partial orders, an adversary can guarantee an outcome for each comparison such that the new p-value is at least 3/4 of the previous p-value. What are the applications? Disappointingly, the resulting bound for median-finding is very weak. For the initial, empty, partial order, p is 2^n, while, finally, we have $p\left(S_{\lfloor n/2 \rfloor}^{\lceil n/2 \rceil - 1}\right) \geq \Theta(2^{\lceil n/2 \rceil})$. The lower bound is therefore

$$M(n) \geq \log_{4/3}(2^{\lfloor n/2 \rfloor}) \sim 1.2n.$$

I shall suggest a possible alternative approach below. But first, for a small non-trivial result using the p-measure, here is a lower bound for an approximate sorting problem. Say that a set is *k-nearly sorted* if it is partially ordered so that, for any rank r, the number of elements which could possibly have rank r in the sorted order is at most k.

Theorem 2. *The worst-case number of comparisons required to k-nearly sort a set of n elements is at least $(n - k - O(\log n)) \log_{4/3} 2$.*

Proof. For any k-nearly sorted partial order π on n elements, $p(\pi) = O(n2^k)$, since the up-sets of any given cardinality r differ only with respect to the at-most-k elements of possible rank r. Our bipartition-counting technique gives a lower bound of about

$$\log_{4/3}(2^n/(n2^k)) = (n - k - O(\log n)) \log_{4/3} 2. \qquad \square$$

For comparison, the linear extension counting measure w yields a bound of $\log_2(n!/k^n) = n(\log_2(n/k) - O(1))$. For small k, the latter bound is better, but for $k > n/12$ the new bound is higher. For example:

Corollary 3. *To $n/8$-nearly sort n elements requires at least $2.1n - O(1)$ comparisons in the worst case.*

Equipartitions

In an attempt to tune our information-theoretic measure to be more sensitive to median-finding, let us now define an *equipartition* to be a bipartition where the up-set is of size $\lceil n/2 \rceil$. Let $Q(\pi)$ be the set of equipartitions compatible with π, and let $q(\pi) = |Q(\pi)|$. For median-finding, we have initially $q = \binom{n}{\lceil n/2 \rceil}$, for the empty partial order, and finally $q = 1$ for an $S_{\lfloor n/2 \rfloor}^{(\lceil n/2 \rceil - 1)}$. If we could establish for the q-measure the analogous result to Theorem 1, we would have a lower bound of about $n \log_{4/3} 2 \sim 2.41n$. We will see however that the analogous result does *not* hold.

We may view Theorem 1 as an affirmation that the probabilities of a and b being in the up-set of a random bipartition (under a suitable distribution) are either positively correlated or independent, i.e.,

$$\frac{p(\pi, a/)}{p(\pi)} \frac{p(\pi, b/)}{p(\pi)} \leq \frac{p(\pi, ab/)}{p(\pi)}$$

or, equivalently,

$$(p_{11} + p_{10})(p_{11} + p_{01}) \leq p_{11}(p_{10} + p_{01} + p_{11} + p_{00}).$$

Under the restriction to *equi*partitions, there is an extra slight tendency toward negative correlation: if a is in the up-set then there is less room up there for b. This effect is extreme in the limiting case of a partial order π where only a and b remain as candidates for the median, the remaining $\lceil n/2 \rceil - 1$ elements of the up-set being determined. In this case, $q(\pi, ab/) = q(\pi, /ab) = 0$, and a factor of $1/2$ (as opposed to $3/4$) results from the comparison $a : b$.

What can we hope to achieve? The measure p has the good $3/4$ ratio property, but is ill-matched to the median problem; the measure q fits the median problem well but fails to satisfy the ratio property. One approach would be to design a compromise measure between p and q, which retains enough of the good properties of each to be useful. For example, it could count all bipartitions, but assign a greater weight the more balanced the partition. An alternative approach is to retain the measure q, but try to show that ratios close enough to $3/4$ could be guaranteed for a sufficiently long initial phase to yield good bounds.

7 Conclusion

We have taken a quick tour of some old and new results in median-finding and selection problems. As Dor and Zwick [5, 6, 7, 8] have broken through the long-standing upper and lower barriers, the time seems ripe for a new leap forward.

I have proposed some speculative new approaches for lower bounds. In the same vein, it would seem that an algorithm might do very well by choosing, where possible, pairs of elements to compare which are each roughly equally likely to lie in the top or bottom halves of the ordering and which are reasonably independent. If this were achievable, the algorithm might come close to reducing the equipartition measure by a factor of $3/4$ per comparison. My ambitious guess with which I close is therefore the following.

Conjecture 1 *The worst-case number of comparisons required to find the median of n elements is asymptotic to $n \log_{4/3} 2 \sim 2.4094 \cdots n$.*

Acknowledgements

I am grateful to Dorit Dor and Uri Zwick for their helpful contributions to this paper.

References

1. S. W. Bent and J. W. John. Finding the median requires $2n$ comparisons. In *Proc. 17th ACM Symp. on Theory of Computing*, 1985, 213–216.
2. M. Blum, R. W. Floyd, V. R. Pratt, R. L. Rivest, and R. E. Tarjan. Time bounds for selection. *J. Comput. Syst. Sci.*, 7, 1973, 448–461.
3. J. W. Daykin. Inequalities for the number of monotonic functions of partial orders. *Discrete Mathematics*, 61, 1986, 41–55.
4. D. E. Daykin, J. W. Daykin, and M. S. Paterson. On log concavity for order-preserving maps of partial orders. *Discrete Mathematics*, 50, 1984, 221–226.
5. D. Dor. *Selection Algorithms.* PhD thesis, Tel-Aviv University, 1995.
6. D. Dor and U. Zwick. Selecting the median. In *Proc. 6th Annual ACM-SIAM Symp. on Discrete Algorithms*, 1995, 28–37.
7. D. Dor and U. Zwick. Finding the αn^{th} largest element. *Combinatorica*, 16, 1996, 41–58.
8. D. Dor and U. Zwick. Median selection requires $(2+\epsilon)n$ comparisons. Technical Report 312/96, April 1996, Department of Computer Science, Tel Aviv University.
9. F. Fussenegger and H. N. Gabow. A counting approach to lower bounds for selection problems. *J. ACM*, 26, 1978, 227–238.
10. A. Hadian and M. Sobel. Selecting the t^{th} largest using binary errorless comparisons. *Colloquia Mathematica Societatis János Bolyai*, 4, 1969, 585–599.
11. L. Hyafil. Bounds for selection. *SIAM J. on Computing*, 5, 1976, 109–114.
12. J. W. John. *The Complexity of Selection Problems.* PhD thesis, University of Wisconsin at Madison, 1985.
13. D. G. Kirkpatrick. Topics in the complexity of combinatorial algorithms. Tech. Rep. 74, Dept. of Computer Science, University of Toronto, 1974.
14. D. G. Kirkpatrick. A unified lower bound for selection and set partitioning problems. *J. ACM*, 28, 1981, 150–165.
15. S. S. Kislitsyn. On the selection of the k^{th} element of an ordered set by pairwise comparisons. *Sibirsk. Mat. Zh.*, 5, 1964, 557–564. (In Russian.)
16. D. E. Knuth. *Sorting and Searching*, volume 3 of *The Art of Computer Programming*. Addison-Wesley, Reading, MA, 1973.
17. T. Motoki. A note on upper bounds for the selection problem. *Inf. Proc. Lett.*, 15, 1982, 214–219.
18. J. I. Munro and P. V. Poblete. A lower bound for determining the median. Technical Report Research Report CS-82-21, University of Waterloo, 1982.
19. V. Pratt and F. F. Yao. On lower bounds for computing the i^{th} largest element. In *Proc. 14th IEEE Symp. on Switching and Automata Theory*, 1973, 70–81.
20. P. V. Ramanan and L. Hyafil. New algorithms for selection. *J. Algorithms*, 5, 1984, 557–578.
21. A. Schönhage, M. S. Paterson, and N. Pippenger. Finding the median. *J. Comput. Syst. Sci.*, 13, 1976, 184–199.
22. J. Schreier. On tournament elimination systems. *Mathesis Polska*, 7, 1932, 154–160. (In Polish.)
23. F. F. Yao. On lower bounds for selection problems. Technical Report MAC TR-121, M.I.T., 1974.
24. C. K. Yap. New upper bounds for selection. *Comm. ACM*, 19, 1976, 501–508.
25. C. K. Yap. New lower bounds for medians and related problems. Computer Science Report 79, Yale University, 1976.

This article was processed using the LaTeX macro package with LLNCS style