A BRIEF HISTORY OF SELECTION

M Paterson

Rapporteur: Avelino Zorzo

A BRIEF HISTORY OF SELECTION

Mike Paterson
Department of Computer Science
University of Warwick
Coventry CV4 7AL

Abstract
The selection problem, determining the k'th largest out of a set of n elements, is less fundamental than sorting, but has still been studied extensively over several decades.

Our focus will be on the worst-case complexity of selection, measured by the number of comparisons required. In contrast to sorting, there is a considerable gap between the upper and lower bounds known for this problem. Although the comparison count is not of prime importance for overall efficiency in most applications, some of the lower bound methods and algorithms derived over the last twenty-five years are of combinatorial interest, and have a place in undergraduate courses on the design and analysis of algorithms.

There has been recent progress in the selection problem, and in median-finding in particular, after a lull of ten years. In these talks I shall review some ancient and modern results on this problem, and suggest possibilities for future research. A full paper appears in [19].

Outline

Let $S(n)$ be the worst-case minimum number of pairwise comparisons required to sort n elements, and $V_k(n)$ the corresponding number to find the k^{th} largest out of n elements. In particular, we are interested in $M(n) = V_{\lceil n/2 \rceil}(n)$, the complexity of finding the *median*. It is well known that $S(n) = n \log_2 n + O(n)$, but for $M(n)$ only rather loose bounds have so far been found.

References [23, 15, 10, 16, 13, 14, 26, 11, 17, 21] describe early work in this area, but 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. Their bound was improved to $3n$ in 1976 [22], and then, only after a further 20 years, has it been further improved slightly by Dor and Zwick [5, 6, 7].

Blum et al. [2] also showed a *lower bound* of $M(n) \geq 3n/2 - O(1)$ by using a simple adversary argument. This lower bound was gradually improved by several authors [11, 20, 13, 26, 18], taking the coefficient for medians from 3/2 up to about 1.837. A major step was taken by Bent and John [1] in 1985 using a "leaf-counting" argument from [9]. They proved a lower bound of $2n - o(n)$. This stood for ten years until a recent tiny improvement by Dor and Zwick [5, 8].

Frances Yao [24] considered the problem of finding a (u, v)-*mediocre element* from m elements, i.e., an element which is smaller than at least u elements and larger than at least v elements. If $m = u + v + 1$, the complexity is just $V_k(m)$, but if more than $u + v + 1$ elements are available the complexity might be less. Yao explored the hypothesis (YH) that the latter complexity is never less, and so far no counter-example to YH is known. However, since YH implies $M(n) \leq 2.5n + o(n)$, it is of interest to determine the truth of YH.

The usual "information theoretic" measure used to prove lower bounds for sorting problems is $w(\pi)$, the number of total orders consistent with the partial order π reached at some stage of an algorithm. Then $\log_2(w(\pi))$ gives a lower bound on the worst-case number of comparisons to complete the sorting of π. For median-finding, the measure w is inappropriate and yields only trivial bounds. We investigate some better measures, based on counting numbers of partitions. A few preliminary results are presented, and a conjecture made as to the asymptotic value of $M(n)$.

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] M. S. Paterson. Progress in selection. *Algorithm Theory - SWAT'96*, LNCS 1097, 368–379. Springer-Verlag, Berlin, Heidelberg, 1996.

[20] 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.

[21] P. V. Ramanan and L. Hyafil. New algorithms for selection. *J. Algorithms*, 5, 1984, 557–578.

[22] A. Schönhage, M. S. Paterson, and N. Pippenger. Finding the median. *J. Comput. Syst. Sci.*, 13, 1976, 184–199.

[23] J. Schreier. On tournament elimination systems. *Mathesis Polska*, 7, 1932, 154–160. (In Polish.)

[24] F. F. Yao. On lower bounds for selection problems. Technical Report MAC TR-121, M.I.T., 1974.

[25] C. K. Yap. New upper bounds for selection. *Comm. ACM*, 19, 1976, 501–508.

[26] C. K. Yap. New lower bounds for medians and related problems. Computer Science Report 79, Yale University, 1976.

(1)

Selection

Recent progress
(and some of the background)

Mike Paterson
University of Warwick

(2)

Sorting, selection, median finding
minimize number of comparisons

Not critical for running time but
- classic combinatorial problems
- nice constructions and proofs
- recent progress

(3)

Worst-case minimum number of pairwise comparisons

$S(n)$: to sort n elements

$V_k(n)$: to find k^{th} largest

$M(n)$: to find $\lceil \frac{n}{2} \rceil^{th}$ largest, the median

(4)

$n =$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$\lceil \log_2 n! \rceil =$	0	1	3	5	7	10	13	16	19	22	26	29	33	37	41	45	49
$B(n) =$	0	1	3	5	8	11	14	17	21	25	29	33	37	41	45	49	54

$n =$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$\lceil \log_2 n! \rceil =$	0	1	3	5	7	10	13	16	19	22	26	29	33	37	41	45	49
$F(n) =$	0	1	3	5	7	10	13	16	19	22	26	30	34	38	42	46	50

$n =$	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
$\lceil \log_2 n! \rceil =$	53	57	62	66	70	75	80	84	89	94	98	103	108	113	118	123
$F(n) =$	54	58	62	66	71	76	81	86	91	96	101	106	111	116	121	126

VALUES OF FACTORIALS IN BINARY NOTATION

⑤

1	= 1!
10	= 2!
110	= 3!
11000	= 4!
1111000	= 5!
1011010000	= 6!
1001110110000	= 7!
1001110110000000	= 8!
1011000100110000000	= 9!
1101110101111100000000	= 10!
10011000010001010100000000	= 11!
11100100011001111110000000000	= 12!
101110011001010001100110000000000	= 13!
10100010011011111110110010100000000000	= 14!
10011000001110111011101110101100000000000	= 15!
100110000011101110111011101011000000000000000	= 16!

⑥

$$n\log_2 n - 1.45n \lesssim S(n) \lesssim n\log_2 n - 1.33n$$

and e.g. $S(21) = 66$

but we have only

$$(2+\delta_1)n \lesssim M(n) \lesssim (3-\delta_2)n .$$

⑦

Pre-history

1883: Charles Dodgson: how to design tennis tournament which also finds 2nd, 3rd best

1929: Hugo Steinhaus: find $V_2(n)$

1932: Schreier: $V_2(n) \le n + \lceil \log_2 n \rceil - 2$

1964: Kislitsyn: $V_2(n) = n + \lceil \log_2 n \rceil - 2$

1969: Hadian & Sobel:

$$V(k) \le n-k+(k-1)\lceil \log_2 (n-k+2) \rceil$$

for all k, n

Asymptotically optimal for $k = o(n/\log n)$

but only gives

$$M(n) \le O(n \log n)$$

⑧

Modern history

1973: Blum, Floyd, Pratt, Rivest, Tarjan

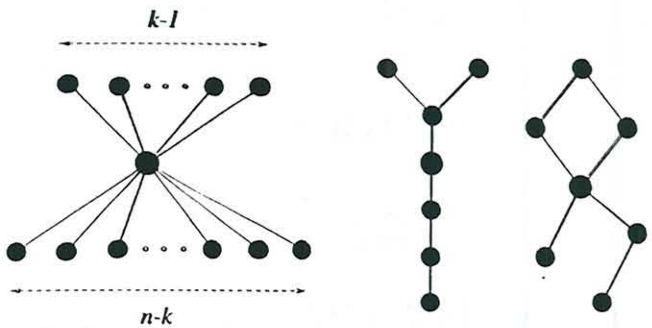
Median-finding is easier than sorting

$$M(n) = O(n)$$

- median of medians
- discard extreme elements
- recursion

$$M(n) \lesssim 5.43n$$

(9)



S^{k-1}_{n-k}

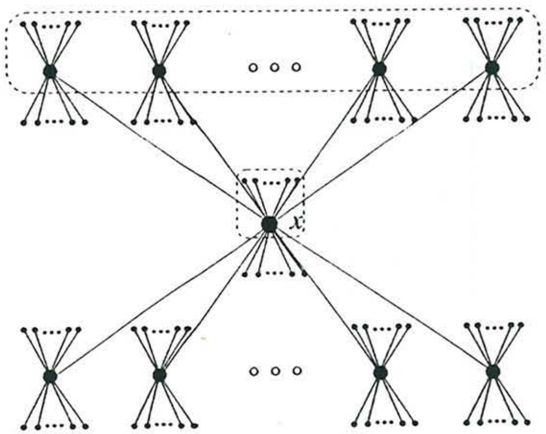
k^{th} largest

contain S^3_3

larger ↑

↓ smaller

(10)



(11)

BFPRT use fixed length sorted chains for the 'small' partial orders

C_{2v+1}  $\Big\}2v+1$

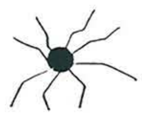
After discarding extreme elements, they leave disconnected C_v's

Task to optimize:
generate new C_{2v+1}'s
using some recycled C_v's

(12)

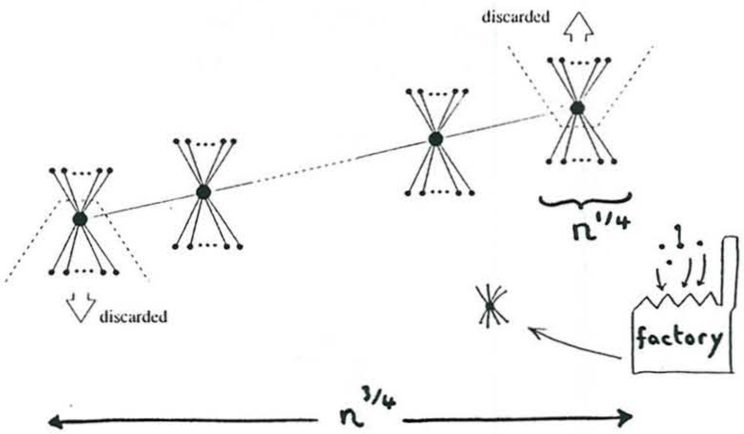
1976: Schönhage, Paterson, Pippenger
different balance of parameters

- "small" S^v_v partial orders are <u>large</u>
  $v = \Theta(n^{1/4})$
- set of medians is kept <u>sorted</u>
- discarding and recycling more continuous, mass-production

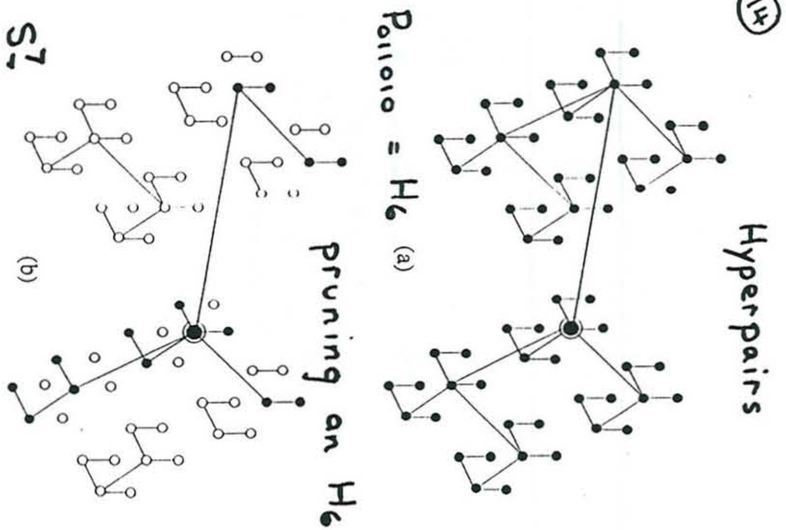
"spider" factory 

VI.8

(13)



(14)



(15)

Spider factories use hyperpairs

- parts pruned off in factory are (smaller) hyperpairs :-)
- parts recycled after discards are not always hyperpairs :-(SPP break these into pairs & singletons
- grafting of •'s and ⫶'s

$$M(n) \lesssim 3n$$

(16)

1976 ······· 20 years on ····· 1995 1996

Dor and Zwick

- 'green' factories recycle more different small partial orders (not just •'s and ⫶'s)
- use more partial orders for grafting
- generalize hyperpairs to hyperproducts
- complex interacting economy of subfactories

$$M(n) \lesssim 2.95\, n$$

and improved $V_k(n)$

17

Lower bounds for $M(n)$

Adversary plays against the algorithm

- decides comparison results
- may give some extra information (to keep the situation simple)

1973: BFPRT $M(n) \gtrsim 1.5n$
1973: Pratt & Yao 1.75
1974: (Schönhage) Kirkpatrick 1.75
1976: Yap 1.81
1982: Munro & Poblete 1.84
(1978: Fussenegger & Gabow)
1985: Bent & John 2

18

Bent & John lower bound for $V_k(n)$

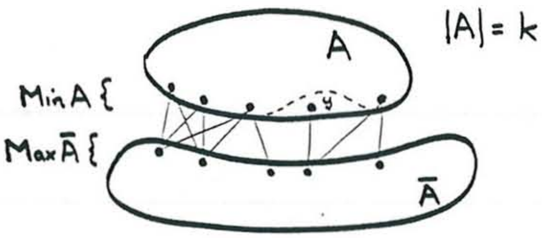
For each subset A of k elements, there is an adversary Y_A

Strategy for Y_A

Phase 1: Answer comparisons by rule that $x \in A$ is above $y \in \bar{A}$. For pairs both in A or both in $\bar{A}$, follow both outcomes generating a tree.

Phase 1 ends when $|\mathrm{Min}\,A| = \sqrt{n}$

19

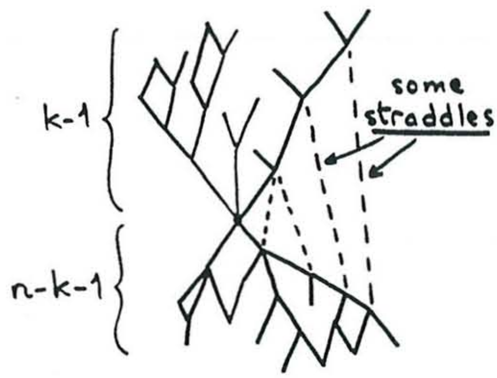


Phase 2: If $\underline{|\mathrm{Max}\,\bar{A}| < 2\sqrt{n}}$
then 2a: Continue as before
At least $\underline{n-2\sqrt{n}}$ total

else 2b: Let y be an element of Min A above the smallest number b of elements of Max $\bar{A}$
$\{|\mathrm{Max}\,\bar{A}| \geq 2\sqrt{n}\}$
Let $A' = A - \{y\}$. Continue, using A' in place of A. Must find largest of $\{y\} \cup \bar{A}$.
Needs at least $\underline{n-3\sqrt{n}+b}$

20

$V_k(n)$



OK if we can show at least n straddles

(21)

If $b > \sqrt{n}$ then there are more than n straddles, ✓ and so more than $2n$ total

Else, each adversary finds a tree with depth $\geq n - 2\sqrt{n}$, i.e. at least $2^{n-2\sqrt{n}}$ leaves

Each leaf corresponds to a set A' of $k-1$ largest elts. and is reached by at most $n-k+1$ adversaries $(A \supseteq A')$

Hence decision tree has at least $\binom{n}{k} \frac{2^{n-2\sqrt{n}}}{n-k+1}$ leaves

$$\Rightarrow \quad V_k(n) \geq n + \log_2\binom{n}{k} - O(\sqrt{n})$$
$$M(n) \geq 2n - O(\sqrt{n})$$

(22)

1996: Dor & Zwick (FOCS)

$M(n) \gtrsim (2+\varepsilon)n, \quad \varepsilon > 2^{-40}$

They prove no more leaves than B&J but show that tree cannot be balanced

Focus on arrival of singletons – their first and second comparisons

(23) Yao's Hypothesis

Frances Yao (1974) considered finding (u,v)-mediocre elements, i.e., find an S^u_v in some <u>larger</u> set

$S(u,v,m)$: # comparisons to find an S^u_v from m elements

$V_k(n) = S(k-1, n-k, n)$

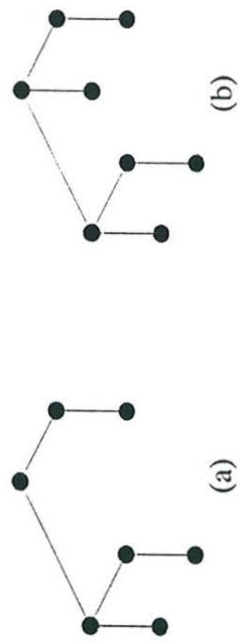
Let $V^*_k(n) = \lim_{m\to\infty} S(k-1, n-k, m)$

Yao's Hypothesis (YH):
$$V^*_k(n) = V_k(n)$$

$$YH \Rightarrow M(n) \lesssim 2.5n$$

Open problem:
prove or disprove YH

(24)



Counter-example to <u>Generalized</u> Yao Hypothesis

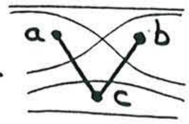
25

bipartition of partial order π on X is a mapping $g : X \to \{0,1\}$ consistent with π

$P(\pi)$: set of bipartitions on π

$p(\pi) = |P(\pi)|$

E.g. $p(\mathrm{V}) = 5$



$P(\pi, a/b)$, $p(\pi, a/b)$: bipartitions where $g(a)=1$, $g(b)=0$

b/a, $ab/$, $/ab$ similarly

26

$$p(\pi) = p(\pi, a/b) + p(\pi, b/a) + p(\pi, ab/) + p(\pi, /ab)$$

$$p(\pi \cup [a>b]) = p(\pi) - p(\pi, b/a)$$
$$p(\pi \cup [a<b]) = p(\pi) - p(\pi, a/b)$$

	b/	/b
a/	ab/	a/b
/a	b/a	/ab

Theorem 1 For any π, a, b, $a/$ and $b/$ are non-negatively correlated

27

Corollary

$$\max \left\{ \begin{matrix} p(\pi \cup [a>b]) \\ p(\pi \cup [a<b]) \end{matrix} \right\} \geq \frac{3}{4} p(\pi)$$

Lower bound theorem

To produce partial order π requires at least

$$\log_{4/3} P(\emptyset)/p(\pi)$$

comparisons

28

For median of n elements

$p(\emptyset) = 2^n$, $\log_{4/3} 2^n \simeq 2.41n$

but

$p(S_{\lfloor n/2 \rfloor}^{\lceil n/2 \rceil - 1}) \simeq 2^{n/2}$

and $\log_{4/3} 2^{n/2} \simeq 1.2n$ ☹

For median and quartiles,

$n \cdot \frac{3}{4} \log_{4/3} 2 \simeq 1.8n$ ☹

29

Set is k-nearly sorted if $\forall r$, # possible elements for r^{th} largest is at most k

k-nearly sorted $\Rightarrow$ no indept. set $>$ k

Corollary
k-nearly sorting needs at least $(n-k-O(\log n))\log_{4/3} 2$

e.g. $n/8$-nearly sort
$\Rightarrow$ 2.1 n comparisons

30

Equipartition:
bipartion into equal parts

Define $Q(\pi)$, $q(\pi)$ similarly to $P(\pi)$, $p(\pi)$

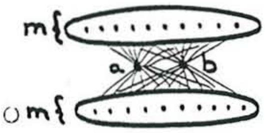
For medians,
initially: $q(\emptyset) = \binom{n}{\lceil n/2 \rceil}$
finally: $q(S) = 1$

An analogue to Theorem 1 for Q would give 2.41n

31

Problem!
Equipartition introduces some negative correlation

If a is up, there is less room at the top for b



Extreme case

Conjecture !?
$M(n) \sim n \log_{4/3} 2$

32

Open problems / ideas

Look for a "φ" which combines good features of p and q

See whether the bound for k-nearly sorting is useful!

Do the p- and q-measures suggest better algorithms?

Prove (or disprove) Yao's Hypothesis

DISCUSSION

Rapporteur: Avelino Zorzo

During his two lectures Professor Paterson described the progress of selection during the last years. In the first talk he described the upper bound, while in the second he concentrated on the lower bound number of comparisons required during the selection. He said that little progress has been done in this area, and that the intended progress is to "reduce the number of comparisons by 2% or 5%". In his lectures, he showed how to obtain such numbers using practical exercises.

Lecture One

When Professor Paterson commented that he found it remarkable that they got so close to the upper bound, Dr Raghavan asked whether the algorithm used was a uniform algorithm or was it necessary to look at the input space. Professor Paterson said that it was a very simple recursive algorithm.

Talking about the same subject, Dr Andersson said that the time could be higher than that mentioned, but Professor Paterson said that it would not be higher than those used in data structures algorithms, and in the data structures algorithms the results are n log n, which was confirmed by Professor Mehlhorn.

Before talking about the lower bound, Professor Tedd wanted to know if the upper bound of 2.95 had been proven or was it just an empirical view. Professor Paterson said that it had been proven.

Lecture Two

Professor Henderson asked about the use of Monte Carlo algorithms to show the upper bound and Professor Paterson answered that he thought that the use of Monte Carlo algorithms was not tried. Professor Paterson also said that he is more concerned about knowing how many comparisons are necessary in the selection process and not in finding a real algorithm to show the solution.