SIAM J. Comput.
Vol. 31, No. 3, pp. 683–705

OPTIMAL SAMPLING STRATEGIES IN QUICKSORT AND QUICKSELECT*

CONRADO MARTÍNEZ† AND SALVADOR ROURA†

Abstract. It is well known that the performance of quicksort can be improved by selecting the median of a sample of elements as the pivot of each partitioning stage. For large samples the partitions are better, but the amount of additional comparisons and exchanges to find the median of the sample also increases. We show in this paper that the optimal sample size to minimize the average total cost of quicksort, as a function of the size n of the current subarray size, is $a \cdot \sqrt{n} + o(\sqrt{n}\,)$. We give a closed expression for a, which depends on the selection algorithm and the costs of elementary comparisons and exchanges. Moreover, we show that selecting the medians of the samples as pivots is not the best strategy when exchanges are much more expensive than comparisons. We also apply the same ideas and techniques to the analysis of quickselect and get similar results.

Key words. quicksort, quickselect, sorting, selection, sampling, median-of-$(2k+1)$, analysis of algorithms, divide-and-conquer

AMS subject classifications. 68P10, 68Q25, 68W05, 68W40

PII. S0097539700382108

1. Introduction. Quicksort [8] and quickselect [7] are among the most thoroughly studied algorithms. Quicksort sorts an array A of n elements by rearranging A around one of its elements—the *pivot*—so that the elements smaller than the pivot are to its left and larger elements are to its right. After the array has been partitioned, quicksort is recursively applied to the subarrays on either side of the pivot. Quickselect[1] selects the mth element (equivalently, the element of *rank* m in ascending order, the mth order statistic) out of n using the same divide-and-conquer principle. Once the array is partitioned and the pivot brought into its correct position, say j, either the sought element is the pivot ($m = j$) or the algorithm recursively continues in the appropriate subarray: the left subarray if $m < j$, the right subarray if $m > j$. Excellent sources for background information, implementation, variants, and their analysis include [3, 12, 13, 22, 23, 24].

In *median-of-three quicksort* [25] the pivot of each recursive stage is the median of a sample of three elements. This variant is easily generalized to select the $(k+1)$th element of a sample of size $s = 2k+1$ as the pivot. Van Emden [26] showed that the average number of comparisons to sort an array of size n is $q(k) \cdot n \ln n + \mathcal{O}(n)$, where $q(k)$ steadily decreases from $q(0) = 2$ to $q(\infty) = 1/\ln 2$. The median-of-three strategy also improves the performance of quickselect [10].

The basic problem with large samples is that the savings achieved because of more balanced partitions can be swamped in practice by the time spent in finding the median of the samples. This cost shows up in the lower order terms of the

*Received by the editors December 8, 2000; accepted for publication (in revised form) June 12, 2001; published electronically October 23, 2001. This research was supported by ESPRIT LTR 20244 (ALCOM-IT), CICYT TIC97-1475-CE, DGES PB95-0787 (KOALA), DGES PB98-0926 (AEDRI), and CIRIT 1997SGR-00366 (SGR).

http://www.siam.org/journals/sicomp/31-3/38210.html

†Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, E-08034 Barcelona, Catalonia, Spain (conrado@lsi.upc.es, roura@lsi.upc.es). The first author was also supported by ACI-CONICYT DOG 2320 (Catalonia-Chile Cooperation Program) and CYTED (RITOS Network).

[1] Also known as Hoare's Find algorithm.

average performance of quicksort and quickselect and cannot be disregarded unless n is impractically large.

McGeoch and Tygar [17] analyzed the expected number of comparisons of quicksort for various sampling strategies. They considered fixed-size sampling and hybrid strategies, which first use a sample whose size is a function of the size of the array to be sorted, and fixed-size samples in subsequent recursive calls. They proved that these strategies are worse than using samples of size $\Theta(\sqrt{n}\,)$, where n denotes the size of the (sub-)array to be sorted in each recursive invocation.

We show in this paper that the optimal sample size as a function of the size n of the current subarray is $a \cdot \sqrt{n} + o(\sqrt{n}\,)$ and determine the value of the constant a in terms of the cost of the median-finding algorithm, taking into account the cost of comparisons and exchanges. Furthermore, we consider the more general setting where we pick the $(p+1)$th element in the sample with $0 \leq p < \lfloor s/2 \rfloor$. (The case for $\lfloor s/2 \rfloor \leq p < s$ reduces to the former by symmetry.)

This paper is organized as follows. In section 2 we set up the basic recurrences for quicksort and quickselect. In section 3 we analyze the variants with fixed-size sampling. We rederive known results such as those of van Emden [26] and those of quickselect with median-of-three [10]. As far as we know, our results regarding the total cost of quicksort and of quickselect with fixed $s > 3$ are original.

In section 4 we study the surprising behavior of quicksort when the cost of an exchange exceeds by far that of a comparison: on the one hand, the optimal value of p is not $p = \lfloor (s-1)/2 \rfloor$ but a function of the ratio between the cost of an exchange and the cost of a comparison; on the other hand, if we always select the medians of the samples as pivots, then fixed-size sampling is better than using samples whose sizes grow with n.

In sections 5 and 6 we tackle the study of the variants with sample sizes depending on n. We analyze quickselect first, since its analysis is easier. We prove in section 5 that if the size of the sample grows with n and is $o(n)$, then the average number of comparisons and exchanges to select an item of random rank are $2n + o(n)$ and $n/2 + o(n)$, respectively. By analyzing the lower order terms, we prove that the optimal sample size is $\Theta(\sqrt{n}\,)$ and give an explicit formula for the constant factor of the main term, which depends on the cost of the algorithm to select pivots and on the cost of elementary comparisons and exchanges.

We consider quicksort in section 6. We prove that when $s = \omega(1)$ and $s = o(n)$ the average number of comparisons is $n \log_2 n + o(n \log n)$ and the average number of exchanges is $\frac{1}{4} n \log_2 n + o(n \log n)$. Using similar techniques to those in section 5 we prove that the optimal sample size for quicksort is $\Theta(\sqrt{n}\,)$ as well and also find the constant factor of the main term. Moreover, we prove that the best pivots are the medians of the samples only when exchanges are not too expensive.

In section 7 we show that using fixed-size samples reduces the constant factor in the main term $\Theta(n^2)$ of the variance of quickselect. Another result with practical implications is that the variance of quickselect is $\mathcal{O}(\max\{n \cdot s, n^2/s\})$ when $s = \omega(1)$. For quicksort, we conjecture that similar results hold. We also discuss tuned implementations of partitioning for quicksort and quickselect, which avoid making redundant comparisons and exchanges.

A preliminary version of this work appears in [15] (see also [16, 20]). Some of the material there is also presented here, in order to make this paper more self-contained.

2. Preliminaries. A basic assumption of this paper is that s, the size of the samples, is a function of the size n of the (sub-)array to be sorted. We will not write down the dependence of s on n most of the time, though. We also assume $s(n) = o(n)$ and, of course, $s(n) \geq 1$. Finally, we assume that the given array contains a random permutation of distinct elements.

We will denote p (respectively, q) the number of elements in the sample smaller than (respectively, larger than) the pivot. Therefore the pivot is the $(p+1)$th element $(0 \leq p < s)$ in the sample, and $q = s - 1 - p$. For the particular case where the pivot is the median of a sample of odd size, we write $s = 2k + 1$, where $k + 1$ is the rank of the pivot, with $p = q = k$.

Most analyses of quicksort and quickselect consider only the number of comparisons. In this paper, we investigate the average *total cost*, that is, the sum of the cost of comparisons and exchanges. We take the cost of a single comparison as the unit and denote ξ the cost of an exchange relative to that of a comparison.

Both quicksort and quickselect operate by selecting a pivot and then partitioning the current subarray w.r.t. the pivot. This requires $n + 1$ comparisons irrespective of s and p.[2] Let $\mathsf{X}(n, s, p)$ denote the average number of exchanges during a single partition stage. We consider here a very common partitioning algorithm, which scans the subarray from left to right until an element larger than the pivot is found, then from right to left until an element smaller than the pivot is found, then exchanges the two designated elements, and then iterates the process again. The partitioning ends when the two scanning pointers meet [22, 23, 24].[3] The following lemma provides the value of $\mathsf{X}(n, s, p)$ for any n, s, and p such that $0 \leq p < s \leq n$. Its proof is given in Appendix B.

LEMMA 1. *The average number of exchanges to partition a random array of size n when the pivot is the $(p+1)$th element of a sample of s elements is*

$$\mathsf{X}(n, s, p) = \frac{(p+1)(q+1)}{(s+1)(s+2)} \frac{(n+1)(n+2)}{n-1} - \frac{n}{n-1}.$$

Let $\mathsf{S}(s, p)$ denote the average total cost of the algorithm to select the $(p+1)$th out of s elements. (This algorithm may or may not be quickselect or one of its variants.) Efficient selection algorithms work in linear time on the average, so we can safely assume $\mathsf{S}(s, p) = \beta \cdot s + o(s)$ for some constant β that depends on the selection algorithm, the costs of comparisons and exchanges, and typically on the ratio $\psi = p/s$. For instance, if we count only comparisons ($\xi = 0$), then $\beta = \frac{3}{2} - \left|\frac{1}{2} - \psi\right|$ for Floyd–Rivest's SELECT [4], $\beta = 2 - 2(\psi \ln \psi + (1-\psi)\ln(1-\psi))$ for Hoare's quickselect [11], and $\beta = 2 + 3\psi(1-\psi)$ for quickselect with median-of-three [10]. An important assumption that we make is that the selection process preserves the randomness of the input array, copying the sample to a separate area if necessary. Otherwise, the recurrences below would not be accurate (see section 7.2).

Let $\pi_{n,j}^{(s,p)}$ be the probability that the $(p+1)$th element of a sample of s elements is the $(j+1)$th element of a random permutation of size n. It is clear that

$$\pi_{n,j}^{(s,p)} = \frac{\binom{j}{p}\binom{n-1-j}{q}}{\binom{n}{s}}, \qquad 0 \leq p < s \leq n, \quad 0 \leq j < n. \tag{2.1}$$

[2] If partitioning uses sentinels, then the number of comparisons can be reduced to $n - 1$, but the main conclusions of this work do not significantly change.

[3] For other randomness-preserving partitioning schemes (e.g., Lomuto's partition, as cited by Bentley [2]) the main conclusions of this paper apply with minor quantitative differences.

The denominator is the number of ways to pick a sample of size s out of n elements; the numerator is the number of ways to choose p elements smaller than the pivot times the number of ways to choose q elements larger than the pivot.

Now we are ready to set the recurrences for Q_n, the average total cost of quicksort, and, for F_n, the average total cost of quickselect averaged w.r.t. the n possible values of the rank of the sought element. F_n has been investigated in [14, 18, 19] as a particular case of *multiple quickselect* with $p = 1$ in the two last references. (There, p is the number of sought elements, not the number of elements smaller than the pivot within the sample.) The term *grand average* has been used to describe F_n and its generalizations for multiple quickselect (see, for instance, [13, 19]). Let $\mathsf{F}_n^{(m)}$ denote the cost of selecting the mth element out of n with quickselect and let F_n denote the cost of selecting an element of random rank. Then

$$F_n = \mathbb{E}[\mathsf{F}_n] = \frac{1}{n} \sum_{1 \le m \le n} \mathbb{E}\left[\mathsf{F}_n^{(m)}\right],$$

but the analogous equation does not hold for the random variables.

For quicksort using samples of size s, the nonrecursive cost of selecting the pivot (the pth element of the sample) and partitioning is

$$\mathsf{S}(s,p) + n + 1 + \mathsf{X}(n,s,p) \cdot \xi,$$

and hence

$$\begin{aligned} Q_n &= \mathsf{S}(s,p) + n + 1 + \mathsf{X}(n,s,p) \cdot \xi \\ &\quad + \sum_{0 \le j < n} \Pr\{\text{the pivot is the } (j+1)\text{th}\} \cdot (Q_j + Q_{n-1-j}) \\ &= \mathsf{S}(s,p) + n + 1 + \mathsf{X}(n,s,p) \cdot \xi + \sum_{0 \le j < n} \left(\pi_{n,j}^{(s,p)} + \pi_{n,n-1-j}^{(s,p)}\right) \cdot Q_j, \end{aligned} \tag{2.2}$$

where we have used obvious symmetries in the last step.

The recurrence for F_n is a bit more involved. The nonrecursive cost is the same as in quicksort. However, the probability that a recursive call with an instance of size j is made is now $\pi_{n,j}^{(s,p)} \cdot \frac{j}{n}$, as we are looking for an element of random rank. Hence, applying the same symmetries as in the recurrence for quicksort,

$$F_n = \mathsf{S}(s,p) + n + 1 + \mathsf{X}(n,s,p) \cdot \xi + \sum_{0 \le j < n} \frac{j}{n} \left(\pi_{n,j}^{(s,p)} + \pi_{n,n-1-j}^{(s,p)}\right) \cdot F_j. \tag{2.3}$$

3. Fixed-size samples. We consider here the solution of (2.2) and (2.3) when $s = \Theta(1)$. This analysis is almost straightforward using the continuous master theorem (CMT) [20, 21]. We briefly review this theorem (Theorem 18) and the necessary definitions in Appendix A for the reader's convenience.

First of all, the *toll function* (the nonrecursive cost) t_n in the recurrences for Q_n and F_n is

$$t_n = \left(1 + \frac{(p+1)(q+1)}{(s+1)(s+2)} \cdot \xi\right) \cdot n + \mathcal{O}(1).$$

Notice that $\mathsf{S}(s,p)$ and several other terms are accounted for in the $\mathcal{O}(1)$-term above. Now we tackle the analysis of quicksort. The first step to apply the CMT requires that we compute the *shape function* $w^{(s,p)}(z)$ of the recurrence, which essentially is a continuous approximation of the discrete weights. In our instance we have

$$w^{(s,p)}(z) = \lim_{n\to\infty} n \cdot \left(\pi^{(s,p)}_{n,z\cdot n} + \pi^{(s,p)}_{n,n-1-z\cdot n}\right) = \frac{s!}{p!\,q!}\left(z^p(1-z)^q + z^q(1-z)^p\right).$$

Since t_n is linear, the limiting const-entropy is evaluated as $\mathcal{H}(s,p) = 1 - \int_0^1 z^1\, w^{(s,p)}(z)\,dz = 0$ for all s and p. Hence, Case 2 of the CMT states that $Q_n = t_n \cdot \ln n / \operatorname{VE}(s,p)$, where $\operatorname{VE}(s,p)$ denotes the limiting log-entropy. We use $\operatorname{VE}(s,p)$ for this value after van Emden, who first used this function for the particular case where $p = q = k$ and $\xi = 0$ [26]. Proposition 19(b) in Appendix B is useful in the following computation:

$$\operatorname{VE}(s,p) \stackrel{\triangle}{=} -\int_0^1 z^1 \,\ln z \,\cdot w^{(s,p)}(z)\,dz = H_{s+1} - \frac{(p+1)H_{p+1} + (q+1)H_{q+1}}{s+1},$$

where $H_n = \sum_{1\le j\le n} 1/j = \ln n + \gamma + \mathcal{O}(1/n)$ denotes the nth *harmonic number*, and $\gamma = 0.577\ldots$ is Euler's constant. Therefore, $Q_n = q_\xi(s,p)\cdot n\ln n + o(n\log n)$, where

$$q_\xi(s,p) \stackrel{\triangle}{=} \left(1 + \frac{(p+1)(q+1)}{(s+1)(s+2)}\cdot\xi\right) / \operatorname{VE}(s,p). \tag{3.1}$$

It is possible to get more information about Q_n by subtracting the main order term from Q_n and setting up a recurrence for the remaining part, i.e., for $R_n = Q_n - q_\xi(s,p)\,nH_n$. Applying the CMT again we get $R_n = \mathcal{O}(n)$. As a conclusion we have the following theorem.

THEOREM 2. *The average total cost of quicksort with fixed-size samples is* $Q_n = q_\xi(s,p)\cdot n\ln n + \mathcal{O}(n)$.

The analysis of quickselect follows similar steps. Computing the shape function yields[4] $w^{(s,p)}(z) = s!/(p!\,q!)\left(z^{p+1}(1-z)^q + z^{q+1}(1-z)^p\right)$. The limiting const-entropy is now

$$\mathcal{H}(s,p) = 1 - \frac{(p+1)(p+2)+(q+1)(q+2)}{(s+1)(s+2)} = \frac{2(p+1)(q+1)}{(s+1)(s+2)},$$

which is strictly positive for every s and p. Therefore $F_n = f_\xi(s,p)\cdot n + o(n)$, where

$$f_\xi(s,p) \stackrel{\triangle}{=} \left(1 + \frac{(p+1)(q+1)}{(s+1)(s+2)}\cdot\xi\right) / \mathcal{H}(s,p) = \frac{(s+1)(s+2)}{2(p+1)(q+1)} + \frac{\xi}{2}.$$

As before, we can estimate the lower order terms of F_n by iteratively using the CMT. Applying it twice, we get a precise asymptotic estimate for the second order term and the order of magnitude of the third order term (see [16] or [20] for additional details).

THEOREM 3. *The average total cost of quickselect with fixed-size samples is*

$$F_n = f_\xi(s,p)\cdot n + \left(\mathsf{S}(s,p) - 2 - \xi - \frac{3(q+1)-p(q+5)}{(s+1)(s+2)}\cdot\xi\right)\cdot\frac{\ln n}{VE(s,p)} + \mathcal{O}(1).$$

[4] We use the same symbol to denote quickselect's shape function and quicksort's shape function, even though they are different.

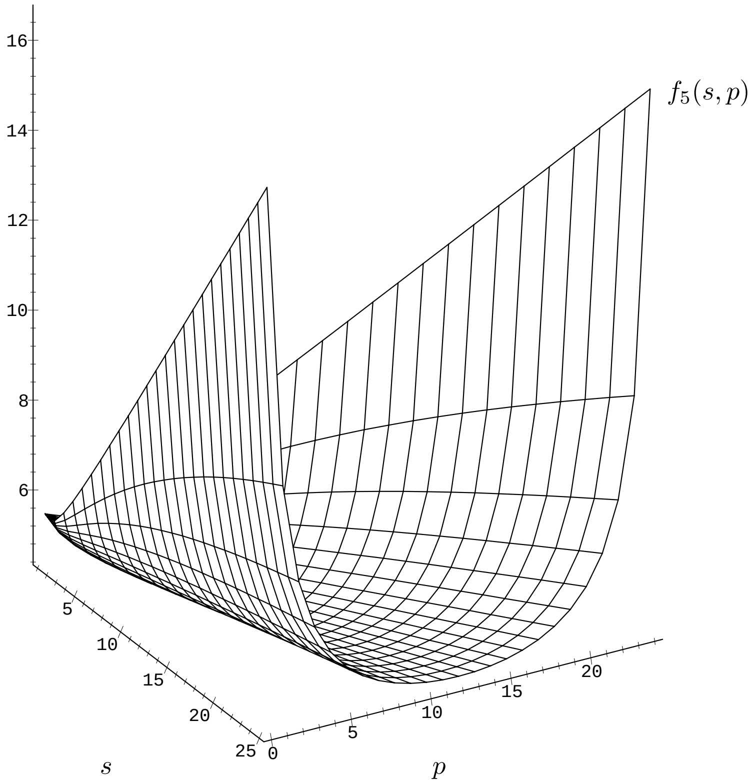

FIG. 1. *Values of* $f_5(s,p)$.

In section 4 we examine in depth the behavior of $q_\xi(s,p)$, the constant factor of the main term of Q_n. The behavior of $f_\xi(s,p)$ in the main term of F_n is much simpler. For any fixed s, the value of p that minimizes $f_\xi(s,p)$ is $p = (s-1)/2$ if s is odd, and $p = (s-2)/2$ (also $p = s/2$ by symmetry) if s is even, regardless of the value of ξ. Figure 1 depicts a typical situation with $\xi = 5$. Moreover, for any $k \geq 0$ we have that $f_\xi(2k+1,k) = f_\xi(2k+2,k) = 2 + 1/(k+1) + \xi/2$ is a decreasing function of k. Hence, our best choice for p is always the median of the sample and to let s be "as large as possible." In section 5 we precisely characterize the best way to do this.

4. Exchanges in quicksort with fixed-size sampling. In this section we study which values of s and p minimize the constant factor $q_\xi(s,p)$ of the $n \ln n$-term in the average cost of quicksort with fixed-size sampling. Given s and ξ, let $p^*(s,\xi)$ be the rank of the optimal pivot. For moderately large values of ξ (this includes the case $\xi = 0$, where we take into account only comparisons) the optimal pivot is $p^* = \lfloor (s-1)/2 \rfloor$, and $q_\xi(2k+1,k)$ is a strictly decreasing function of k. By contrast, for large values of ξ the best fixed-size strategy is indeed to have large samples, but the pivot should be an element of rank $p^* \neq \lfloor (s-1)/2 \rfloor$.

For convenience we will work with the quotients $\psi = p/s$ rather than with the actual ranks p for the rest of this section. Figure 2 shows the function $q_{30}(s,p)$ with two valleys symmetrically disposed at both sides of the line $\psi = 1/2$. For each s, we take the smallest p with minimum $q_{30}(s,p)$ to define ψ^*, which is clearly smaller than

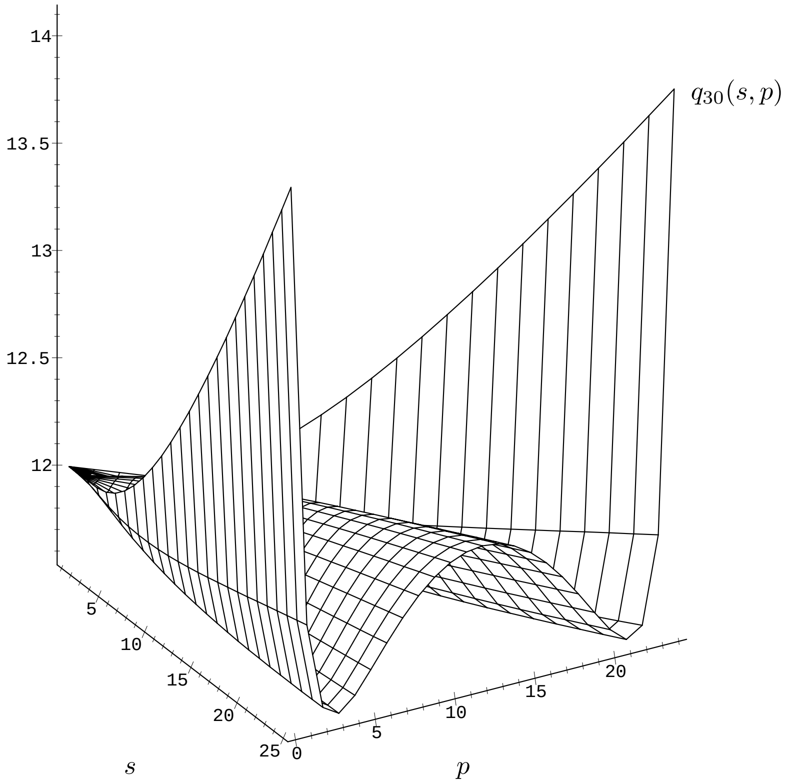

FIG. 2. *Values of* $q_{30}(s,p)$.

1/2.

In Figure 3 we plot the (scaled) values of the function

$$q_\xi(\psi) \overset{\triangle}{=} \lim_{s\to\infty} q_\xi(s, \psi\cdot s + o(s)) = -\frac{1+\psi(1-\psi)\xi}{\psi\ln\psi + (1-\psi)\ln(1-\psi)}$$

for $\xi = 0, 5, 15, 20$, and 30. Note that the denominator is, besides the limiting value $\lim_{s\to\infty} \mathrm{VE}(s, \psi\cdot s)$, the entropy for the probabilities ψ and $1-\psi$ (with the minus sign). It appears often enough to deserve a special name: for $0 < x < 1$, let

$$h(x) \overset{\triangle}{=} -x\ln x - (1-x)\ln(1-x). \tag{4.1}$$

The function is symmetric around $\psi = 1/2$ with only one minimum located at $\psi^* = 1/2$ when ξ is not larger than a threshold value τ; otherwise, we have a local maximum at $\psi = 1/2$ and two absolute minima at $\psi^* < 1/2$ and at $1-\psi^* > 1/2$. In this last case ψ^* is the unique solution in the interval $(0, 1/2)$ of the equation $\partial q_\xi(\psi)/\partial\psi = 0$, or, equivalently, of

$$\ln\psi + \xi\,\psi^2\ln\psi = \ln(1-\psi) + \xi(1-\psi)^2\ln(1-\psi). \tag{4.2}$$

The threshold value τ is given by the solution of the equation

$$\left.\frac{\partial^2 q_\xi(\psi)}{\partial\psi^2}\right|_{\psi=1/2} = 0,$$

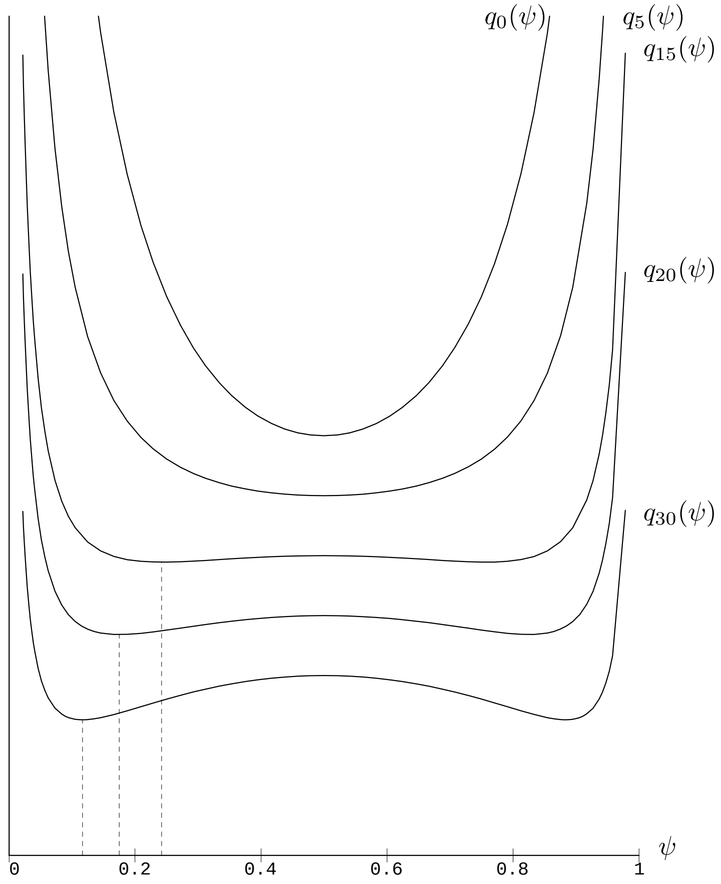

FIG. 3. *Values of $q_\xi(\psi)$ for $\xi = 0, 5, 15, 20,$ and 30.*

which is $\tau = 4/(2\ln 2 - 1) \approx 10.35480$. In Figure 4 we see that $\psi^*(\xi) = 1/2$ for $\xi \in [0, \tau]$, whereas $\psi^*(\xi)$ strictly decreases for $\xi > \tau$, tending to 0 as ξ grows.

For large s we know that p^* is roughly $\psi^* \cdot s$. However, now the question is whether taking large samples is the best choice (on the asymptotic regime) among fixed-size sampling strategies. The next theorem states just that.

THEOREM 4. *For any s and p, $q_\xi(s,p) > q_\xi(\psi^*)$.*

Proof. Let $u(z) = s!/p!\,q! \cdot z^p(1-z)^q$. From (3.1) and Proposition 19, we have

$$q_\xi(s,p) = \frac{\int_0^1 u(z)\,(1 + z(1-z)\xi)\;dz}{\int_0^1 u(z)h(z)\,dz}.$$

We will use the following fact: Let $f(z)$, $g(z)$ be positive functions over the interval $[0,1]$. Let z^* be the location of a minimum of $f(z)/g(z)$, and assume $0 < z^* < 1$ and $g(z) \neq 0$ for $0 < z < 1$. Then, as $f(z) \geq g(z)f(z^*)/g(z^*)$ and $u(z)$ is also positive in the interval $[0,1]$, it follows that

$$\frac{\int_0^1 u(z)f(z)\,dz}{\int_0^1 u(z)g(z)\,dz} \geq \frac{\int_0^1 u(z)\,[g(z)f(z^*)/g(z^*)]\;dz}{\int_0^1 u(z)g(z)\,dz} = \frac{f(z^*)}{g(z^*)}.$$

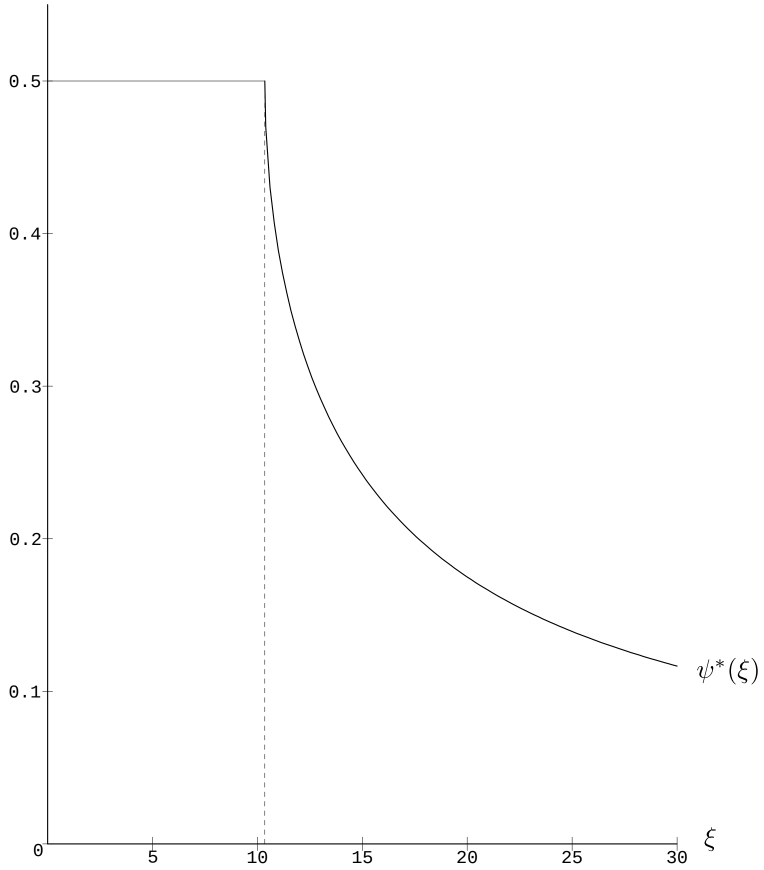

FIG. 4. *Values of* $\psi^*(\xi)$.

Now taking $f(z) = 1 + z(1-z)\xi$, $g(z) = h(z) = -z \ln z - (1-z) \ln(1-z)$, and since ψ^* is the minimum of $q_\xi(\psi) = f(\psi)/g(\psi)$, we have $q_\xi(s,p) \geq q_\xi(\psi^*)$, and the statement of the theorem is almost proved. Finally, notice that for any ξ there is always an interval $[\psi_1, \psi_2]$ with $0 \leq \psi_1 < \psi_2 \leq 1$ such that $q_\xi(\psi) > q_\xi(\psi^*(\xi))$ and $u(z) > 0$ for every ψ in (ψ_1, ψ_2). Thus $q_\xi(s,p) > q_\xi(\psi^*)$. □

An intuitive explanation for the theorem above goes as follows. Assume that we had a black-box routine such that, given an array of size n and a value $0 < \psi < 1$, it returned the $\lfloor \psi \cdot n \rfloor$th element in the array *at no cost*. Using such a routine (with some fixed ψ) in the pivot-finding stage of quicksort provides a variant of quicksort with toll function $(1 + \psi(1-\psi)\xi)$, log-entropy $h(\psi)$, and average total cost $q_\xi(\psi)\, n \ln n + o(n \log n)$. Thus the optimal choice for ψ is ψ^*. Now, as we are deprived from such a routine, we try to get the best possible estimate of the $\lfloor \psi^* \cdot n \rfloor$th element in the array, which can be done by taking large samples and selecting the $\lfloor \psi^* \cdot s \rfloor$th element.

Several remarks are in order. On the one hand, the fact that ψ^* tends to 0 as ξ tends to ∞ can be informally described as a smooth transition from quicksort to selection sort. Selection sort is a good sorting method when exchanges are extremely expensive because it minimizes their number, and quicksort behaves exactly as selec-

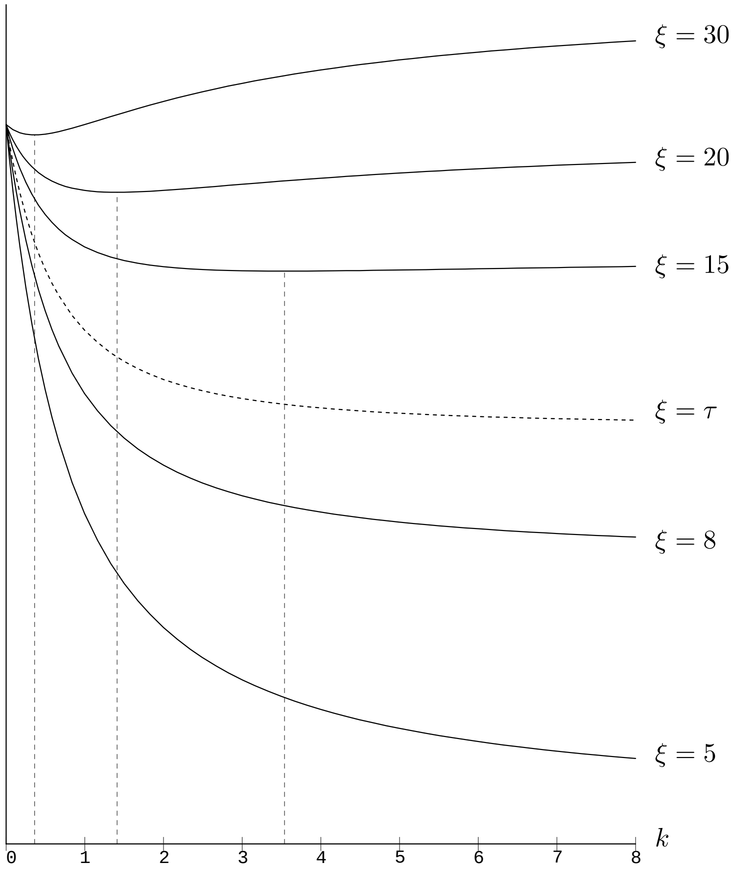

FIG. 5. *Values of* $q_\xi(k)$ *for* $\xi = 5,\ 8,\ \tau,\ 15, 20,$ *and* 30.

tion sort when the smallest element in the array is always selected as the pivot. On the other hand, we should be aware that the analysis of the case $\xi > \tau$ is mainly of theoretical interest, since we should sort an array of pointers to the actual records rather than sorting the records themselves if data movements were too expensive.

Now we restrict our attention to the variants of quicksort that always take the medians of samples of fixed size as the pivots (and therefore are not the best theoretical alternatives when $\xi > \tau$); that is, we take $s = 2k+1$ and $p = q = k$ for some $k \geq 0$. This time we have

$$q_\xi(k) \overset{\triangle}{=} q_\xi(2k+1, k) = \frac{1 + (k+1)/(4k+6)\cdot\xi}{H_{2k+2} - H_{k+1}}.$$

The (scaled) shape of this function is shown in Figure 5 for several values of ξ. The plot actually depicts the extension of $q_\xi(k)$ to the real numbers using the continuous function $\Psi(z) = d\ln\Gamma(z)/dz$; recall that $\Psi(n+1) = H_n - \gamma$ for positive integers [1]. For $\xi = 5$, $\xi = 8$, and $\xi = \tau$ (dashed in the figure) the function $q_\xi(k)$ steadily decreases with k in accordance with what we know. This behavior changes as soon as $\xi > \tau$, as $q_\xi(k)$ has one minimum at finite distance $k^*(\xi)$. Observe in Figure 5 that the location of the minima (which is shown for $\xi = 15$, 20, and 30) tends to 0

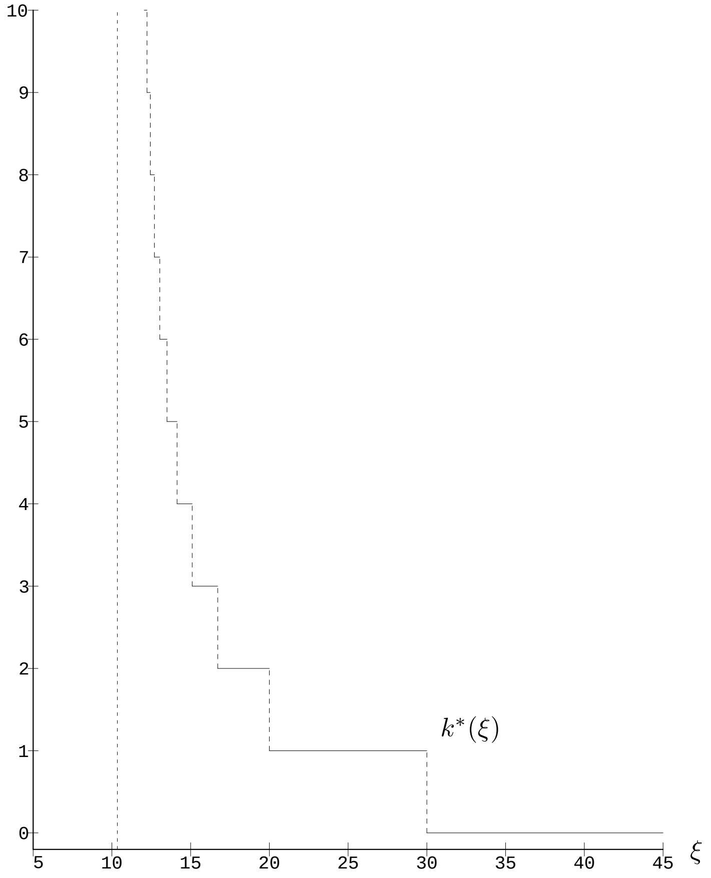

FIG. 6. *Values of* $k^*(\xi)$.

when ξ grows.

Figure 6 shows the function $k^*(\xi)$. There is a vertical asymptote as $\xi \to \tau^+$. For values of ξ larger than 30 we have $k^* = 0$ (plain quicksort). Note that k^* is not well defined for some values of ξ. For instance, 1 and 2 compete as optimal choices for k when $\xi = 20$.

The function

$$\xi^*(k) \triangleq \frac{4}{4(H_{2k} - H_k)\frac{k+1}{2k+3} - 1 + \frac{1}{2k+1}}$$

is the pseudoinverse of $k^*(\xi)$ in the sense that k is the optimal choice if ξ belongs to the open interval $(\xi^*(k+1), \xi^*(k))$. Therefore, $q_{\xi^*(k)}(k) = q_{\xi^*(k)}(k-1)$ for any $k > 0$. For instance, we have $k^* = 1$ when $20 < \xi < 30$ because $\xi^*(2) = 20$ and $\xi^*(1) = 30$. By convention we take $\xi^*(0) = \infty$.

As a conclusion, we could say that a good estimation of the median of the (sub-) array is always profitable as far as comparisons are concerned. However, regarding exchanges we have a trade-off between the short-term profit of selecting a pivot far away from the median (since only a few exchanges would be needed in that stage) and the long-term gains of selecting a pivot closer to the median (since an even partition means fewer recursive stages and fewer exchanges to be done in the long run). Roughly speaking, this is why small samples—and consequently, poor estimations of the medians—are preferable when the important quantity to minimize is the number of exchanges, i.e., when ξ is large.

5. Optimal samples for quickselect. We already know from section 3 that selecting the medians of the samples is the best choice to minimize the average cost of quickselect. Therefore, we will assume $s = 2k+1$ and $p = q = k$ for the rest of this section, and we will simplify the notation for functions like $\mathsf{X}(n, 2k+1, k)$, writing $\mathsf{X}(n,k)$ instead. Here we will consider the case where $k = k(n)$ is a function of the current subarray size n.

In principle it pays to have samples as large as possible. However, if $s = \Theta(n)$ the average total cost is not optimal, since the cost of the selection of the pivot is of the same order of magnitude as the cost of the partition. Hence, we shift our attention towards samples whose size increases with n but are sublinear, that is, $s = \omega(1)$ and $s = o(n)$.

Under the hypotheses above, $s \to \infty$ when $n \to \infty$, so the coefficient of n in F_n should be the limit of $f_\xi(s, s/2)$ for $s \to \infty$. The following theorem rigorously establishes our intuition. The CMT itself cannot be applied here because the shape function for this case is not well defined; we require more basic results that we also review in Appendix A.

THEOREM 5. *Let $s = 2k+1$ and $p = q = k$, where $k = \omega(1)$ and $k = o(n)$. Then the average total cost of quickselect to find an element of random rank out of n is $F_n = (2 + \xi/2)\, n + o(n)$.*

Proof. Let $F_n(t)$ be the average total cost of quickselect when we use samples of fixed size $2t+1$, and let $\pi_{n,j}^{(t)} = \pi_{n,j}^{(2t+1,t)}$. The const-entropy associated to $F_n(t)$ (see Appendix A) is

$$\mathcal{H}_n(t) = 1 - \frac{2}{n^2} \sum_{0 \le j < n} j^2\, \pi_{n,j}^{(t)} + o(1),$$

which by Proposition 21 in Appendix B has limit $\mathcal{H}(t) = \lim_{n\to\infty} \mathcal{H}_n(t) = 1/2 - 1/(4t+6)$. Fix any $t' \ge t$. Since x^2 is a convex function, the probabilities $\pi_{n,j}^{(\cdot)}$ are symmetric in j and $n-1-j$, and the weights for t' are more concentrated around $(n-1)/2$ than the weights for t, it follows that $\sum_{0\le j<n} j^2 \pi_{n,j}^{(t')} \le \sum_{0 \le j < n} j^2 \pi_{n,j}^{(t)}$, and hence $\mathcal{H}_n(t') \ge \mathcal{H}_n(t)$ if n is large enough.

On the other hand, the const-entropy associated to F_n is

$$\mathcal{H}_n = 1 - \frac{2}{n^2} \sum_{0 \le j < n} j^2 \pi_{n,j}^{(k(n))} + o(1).$$

Since $k = \omega(1)$, for any fixed t we have $k(n) > t$ and $\mathcal{H}_n \ge \mathcal{H}_n(t)$ as long as n is large enough. Hence $\mathcal{H} = \lim_{n\to\infty} \mathcal{H}_n \ge \lim_{n\to\infty} \mathcal{H}_n(t) = \mathcal{H}(t)$. Moreover, $\mathcal{H} \ge 1/2 - 1/(4t+6)$ for any t, so $\mathcal{H} \ge 1/2$. Furthermore, the upper bound $\mathcal{H}_n \le 1/2$ can be easily derived using the probabilities

$$\pi_{n,j} = \begin{cases} 1 & \text{if } j = \lfloor (n-1)/2 \rfloor, \\ 0 & \text{otherwise.} \end{cases}$$

Therefore, $\mathcal{H} = 1/2$. Since under the hypotheses the toll function is $t_n = (1 + \xi/4) \cdot n + o(n)$, we can conclude from (A.2) that $F_n = (1+\xi/4)\, n/\mathcal{H} + o(n) = (2 + \xi/2)\, n + o(n)$. □

If we consider only comparisons ($\xi = 0$) Theorem 5 states that the average cost of quickselect is $2n + o(n)$, well above the $1.25n + o(n)$ comparisons needed on average to locate an element of random rank using Floyd and Rivest's SELECT algorithm [4] but well below the $3n + o(n)$ comparisons of standard quickselect. Using rather different techniques than ours, Grübel [6] has shown that $\mathsf{W}_n = \sup_{1 \le m \le n}\{\mathsf{F}_n^{(m)}\}$ converges in probability to $2n$ if $k = \omega(1)$ and $k = o(\frac{n}{\log n})$.

Theorem 5 leaves open which is the optimal sample size. To find it we introduce the function

$$\mathcal{F}(n,k) = -\frac{1}{2} + \sum_{0 \le j < n} \frac{j^2}{n^2}\left(\pi_{n,j}^{(k)} + \pi_{n,n-1-j}^{(k)}\right) = -\frac{1}{2} + \frac{2}{n^2}\sum_{0 \le j < n} j^2 \pi_{n,j}^{(k)},$$

which is only a slight variation of the const-entropy associated to F_n (see the proof of Theorem 5). As $k = \omega(1)$ and $k = o(n)$, we have $F_n = (2 + \xi/2) \cdot n + G_n$ for some function $G_n = o(n)$. Substituting in the recurrence for F_n we get

$$F_n = n + 1 + \mathsf{S}(k) + \mathsf{X}(n,k) \cdot \xi + 2\sum_{0 \le j < n} \frac{j}{n}\pi_{n,j}^{(k)} G_j + (1 + \xi/4) \cdot n + (2 + \xi/2) \cdot \mathcal{F}(n,k) \cdot n.$$

The asymptotic behavior of $\mathsf{X}(n,k)$ and $\mathcal{F}(n,k)$ when k grows with n is stated in the following lemmas.

LEMMA 6. *If $k = \omega(1)$ and $k = o(n)$, then*

$$\mathsf{X}(n,k) = \frac{n}{4} - \frac{n}{8k} + \Theta\left(\frac{n}{k^2}\right).$$

Proof. Setting $s = 2k + 1$ and $p = k$ in Lemma 1 yields

$$\mathsf{X}(n,k) = \frac{(k+1)n^2 - (k+3)n + (2k+2)}{2(2k+3)(n-1)}.$$

Now it is a simple matter to rewrite the equality above to get the statement of the lemma. $\square$

LEMMA 7. *If $k = \omega(1)$ and $k = o(n)$, then*

$$\mathcal{F}(n,k) = \frac{1}{4k} + \Theta(\max\{1/k^2, 1/n\}).$$

Proof. Setting $s = 2k + 1$ and $p = k$ in Proposition 21 of Appendix B we get

$$\mathcal{F}(n,k) = \frac{1}{4k} - \frac{3}{4k(2k+3)} - \frac{3}{2n} + \frac{3}{2(2k+3)n} + \frac{1}{(2k+3)n^2}. \qquad \square$$

Since $G_n = o(F_n)$, it is rather intuitive that the contribution of the sum of G_j's to the asymptotic location of k^* is irrelevant. The argument can be formalized as follows. Fix any $\delta > 0$. By hypothesis, there exists some N such that $|G_j| \le \delta \cdot j$ for every $j \ge N$. On the other hand, we are assuming $k = \omega(1)$, so for large n we have $k \ge N$ and $\pi_{n,j}^{(k)} = 0$ for every $j < N$. Therefore,

$$2\sum_{0 \le j < n} \frac{j}{n} \cdot \pi_{n,j}^{(k)} \cdot |G_j| \le 2\delta \cdot \sum_{0 \le j < n} \frac{j^2}{n} \cdot \pi_{n,j}^{(k)} = \left(\delta \cdot \mathcal{F}(n,k) + \frac{\delta}{2}\right) \cdot n$$

when n is large enough. Thus

$$F_n = n + 1 + \mathsf{S}(k) + \mathsf{X}(n,k) \cdot \xi + E_1(n) \cdot n + E_2(n) \cdot \mathcal{F}(n,k) \cdot n, \tag{5.1}$$

where we have the bounds $(1+\xi/4)-\delta/2 \le E_1(n) \le (1+\xi/4)+\delta/2$ and $(2+\xi/2)-\delta \le E_2(n) \le (2+\xi/2)+\delta$. Notice that the bounds for $E_1(n)$ and $E_2(n)$ do not depend on k. Moreover, $\mathsf{S}(k) \sim 2\beta\, k$ for some constant β, where β depends on the chosen selection algorithm. Then it is easy to see that, for large n, k^* belongs to the interval $[k_1, k_2]$, where k_1 minimizes the function of k

$$2(1+\delta)\,\beta\, k - \frac{(1+\delta)\cdot\xi}{8k}\cdot n + \left[\left(2+\frac{\xi}{2}\right)-\delta\right]\frac{(1-\delta)}{4k}\cdot n,$$

that we obtain by plugging the asymptotic estimates of $\mathsf{S}(k)$, $\mathsf{X}(n,k)$, and $\mathcal{F}(n,k)$ and the lower bounds on $E_1(n)$ and $E_2(n)$ in (5.1), disregarding those terms not depending on k. Similarly, k_2 minimizes the function of k

$$2(1-\delta)\,\beta\, k - \frac{(1-\delta)\cdot\xi}{8k}\cdot n + \left[\left(2+\frac{\xi}{2}\right)+\delta\right]\frac{(1+\delta)}{4k}\cdot n.$$

This yields

$$k_1 = \sqrt{\frac{((4+\xi)-2\delta)(1-\delta)-(1+\delta)\cdot\xi}{16(1+\delta)\beta}}\cdot\sqrt{n},$$

$$k_2 = \sqrt{\frac{((4+\xi)+2\delta)(1+\delta)-(1-\delta)\cdot\xi}{16(1-\delta)\beta}}\cdot\sqrt{n}.$$

The reasoning above holds for every $\delta > 0$, no matter how small we choose it. Therefore, we can conclude the following theorem.

THEOREM 8. *Let $s^* = 2k^*+1$ be the optimal sample size w.r.t. the average total cost of quickselect to select a random element when the median of the sample is used as the pivot. Then $k^* = \sqrt{n/4\beta} + o\,(\sqrt{n}\,)$.*

The procedure that we have just outlined here will be used again in section 6. Observe that it has been enough to consider the main terms of $\mathsf{S}(k)$, $\mathsf{X}(n,k)$, and $\mathcal{F}(n,k)$ to find the main term of k^*. In section 6 we will avoid dealing with the tedious details by disregarding terms of small order and by obtaining minima as if these terms did not exist. We have already shown here that it can be done safely and yields asymptotically valid conclusions.

Finally, notice that $\Theta(\sqrt{n}\,)$-sampling improves the worst-case complexity of quickselect from $\Theta(n^2)$ to $\Theta(n^{3/2})$; the same happens with quicksort [17]. This is valid no matter what the rank of the sought element is and even if the selection of pivots is made using an algorithm with quadratic worst case.

6. Optimal samples for quicksort.

As shown in section 4, picking the median of the sample as the pivot is not always the best choice. Therefore, we set $p = \psi\cdot s + o(s)$ for some fixed $0 < \psi \le 1/2$. We do not necessarily assume $\psi = \psi^*(\xi)$.

The following theorem provides the main term of the total cost of quicksort when we make the size of the sample grow with the size of the input. The proof (which we do not give here) uses an argument quite similar to that of Theorem 5, finding matching lower and upper bounds for the limiting log-entropy of Q_n. (For further details see [16, 20].)

THEOREM 9. *Let $p = \psi \cdot s + o(s)$, where $0 < \psi < 1$ and $s = \omega(1)$ and $s = o(n)$. Then the average total cost of quicksort is $Q_n = q_\xi(\psi) \cdot n \ln n + o(n \log n)$.*

If we measure only the number of comparisons ($\xi = 0$), the theorem above states that *any* sample size $s = \omega(1)$ and $s = o(n)$ with $\psi = 1/2$ is asymptotically optimal w.r.t. the main term of quicksort, as the expected number of comparisons for all of them is $Q_n \sim n \log_2 n$.

To investigate the optimal sample size we will assume $p = \lceil \psi \cdot (s+1) \rceil - 1$, but any other discretization satisfying $\lim_{s \to \infty}(p/s) = \psi$ would yield similar results.

Let us introduce

$$\mathcal{Q}(n, s, \psi) = -\left(H_n - h(\psi)\right) \frac{n-1}{n} + \sum_{0 \le j < n} \left(\pi_{n,j}^{(s,p)} + \pi_{n,n-1-j}^{(s,p)}\right) \frac{jH_j}{n},$$

where $h(x) = -x \ln x - (1-x)\ln(1-x)$ (see (4.1)). The function $\mathcal{Q}(n, s, \psi)$ plays a role analogous to that of $\mathcal{F}(n, k)$ in the analysis of quickselect. By Theorem 9 we can decompose Q_n as $Q_n = q_\xi(\psi)\, nH_n + R_n$, where $R_n = o(n \log n)$. The recurrence for quicksort (2.2) can be then rewritten as

$$Q_n = \mathsf{S}(s, \psi) + n + 1 + \mathsf{X}(n, s, \psi) \cdot \xi + \sum_{0 \le j < n} \left(\pi_{n,j}^{(s,p)} + \pi_{n,n-1-j}^{(s,p)}\right) \cdot R_j$$
$$+ \Big(H_n - h(\psi)\Big) \cdot q_\xi(\psi) \cdot (n-1) + \mathcal{Q}(n, s, \psi) \cdot q_\xi(\psi) \cdot n.$$

Let $\chi_\psi(s)$ be the oscillating factor induced by taking ceilings: $\chi_\psi(s) = \lceil \psi \cdot (s+1) \rceil - \psi \cdot (s+1)$. We have the following lemma for the asymptotic behavior of $\mathcal{Q}(n, s, \psi)$.

LEMMA 10. *If $s = \omega(1)$, $s = o(n)$, and $p = \lceil \psi \cdot (s+1) \rceil - 1$ for some ψ such that $0 < \psi \le 1/2$, then $\mathcal{Q}(n, s, \psi) = 1/2s - (\ln(1-\psi) - \ln \psi) \cdot \chi_\psi(s)/s + \mathcal{O}\left(\max\left\{1/n, 1/s^2\right\}\right)$.*

Proof. Using Proposition 20(b) in Appendix B yields

$$\sum_{0 \le j < n} \left(\pi_{n,j}^{(s,p)} + \pi_{n,n-1-j}^{(s,p)}\right) jH_j = (H_n - \mathrm{VE}(s,p)) \cdot (n-1)$$
$$+ \frac{p+1}{s+1}(H_{p+1} - H_{q+1}) + \frac{q+1}{s+1}(H_{q+1} - H_{p+1}) + \frac{1}{p+1} + \frac{1}{q+1} - \frac{2}{s+1} - 1.$$

The last four terms above are $\mathcal{O}(1)$. On the other hand, from the equalities $p + 1 = \psi \cdot (s+1) + \chi_\psi(s)$, $H_n = \ln n + \gamma + 1/(2n) + \Theta(n^{-2})$, and $\ln(1 + 1/x) = 1/x + \Theta(x^{-2})$, it is not difficult to deduce that

$$-H_{s+1} + \frac{p+1}{s+1} \cdot H_{p+1} + \frac{q+1}{s+1} \cdot H_{q+1}$$
$$= -h(\psi) + \frac{1}{2s} - \Big(\ln(1-\psi) - \ln \psi\Big) \frac{\chi_\psi(s)}{s} + \Theta\left(\frac{1}{s^2}\right),$$

and the lemma follows after a few simple manipulations. □

The factor $\ln(1-\psi) - \ln \psi$ in the statement of Lemma 10 is zero only when $\psi = 1/2$. Therefore, the perturbation $\chi_\psi(s)/s$ has to be taken into account whenever we are not selecting the median of the sample as the pivot.

As pointed out in section 2, $\mathsf{S}(s, p)$ is linear with the constant of proportionality typically dependent on the quotient $\psi = p/s$. Thus we assume $\mathsf{S}(s, p) = \mathsf{S}(s, \psi) = \beta \cdot s + o(s)$ for some constant $\beta = \beta(\psi)$. We also need the asymptotic properties of

$\mathsf{X}(n,s,\psi)$ given in the next lemma. It extends Lemma 6 for $\psi < 1/2$. The proof, very similar to others in this work, is omitted.

LEMMA 11. *If $s = \omega(1)$, $s = o(n)$, and $p = \lceil \psi \cdot (s+1) \rceil - 1$ for some ψ such that $0 < \psi \le 1/2$, then*

$$\mathsf{X}(n,s,\psi) = \psi(1-\psi)n\left(1 - \frac{1}{s}\right) + (1-2\psi)\cdot \chi_\psi(s) \cdot \frac{n}{s} + \mathcal{O}\left(\max\left\{n/s^2, 1\right\}\right).$$

Therefore, by a reasoning identical to that of section 5, we can obtain the leading term of the optimal sample size s^* by minimizing

$$\beta \cdot s + \frac{n}{s}\cdot\left(\frac{q_\xi(\psi)}{2} - \psi\cdot(1-\psi)\cdot\xi\right) + \frac{n}{s}\cdot\Big((1-2\psi)\cdot\xi - (\ln(1-\psi) - \ln\psi)\cdot q_\xi(\psi)\Big)\cdot\chi_\psi(s). \tag{6.1}$$

We will consider two variants of quicksort with sample sizes depending on n. The first one is setting $\psi = \psi^*(\xi)$, i.e., choosing the optimal ratio p/s, and the second one is setting $\psi = 1/2$, that is, always picking the median of the sample (if $\xi \le \tau$, both variants are identical, as $\psi^* = 1/2$ in this case). Fortunately, in any of these two variants the last term of (6.1) vanishes because of (4.2). Therefore, we can follow similar steps to those in the analysis of quickselect to obtain the two theorems below.

THEOREM 12. *Let s^* denote the optimal sample size w.r.t. Q_n when we use the $\lceil \psi^*(s+1) \rceil$th element of the sample as the pivot. Then*

$$s^* = \sqrt{\left(\frac{q_\xi(\psi^*)}{2} - \psi^*\cdot(1-\psi^*)\cdot\xi\right)\frac{1}{\beta(\psi^*)}}\cdot\sqrt{n} + o\left(\sqrt{n}\right).$$

THEOREM 13. *Let s^* denote the optimal sample size w.r.t. Q_n when the median of the sample is used as the pivot. Then*

$$s^* = \sqrt{\left(\frac{4 - (2\ln 2 - 1)\cdot\xi}{8\ln 2}\right)\frac{1}{\beta(1/2)}}\cdot\sqrt{n} + o\left(\sqrt{n}\right).$$

Notice that Theorem 13 makes no sense when $\xi > \tau$, as the factor multiplying $\sqrt{n}$ would be the square root of a negative number. This provides a further check for the conclusions of section 4 and is consistent with the observation that the optimal samples in that situation have constant size.

In [17] a dynamic-programming algorithm was used to compute several exact values of the optimal sample size for quicksort to minimize the average number of comparisons, assuming that standard quickselect was the median-finding algorithm. The authors reported that their data suggests a best power-law fit proportional to $n^{0.588}$. Also, for a tuned variant of quicksort (see section 7.2) they found that the best fit is proportional to at least $n^{0.475}$.

Figure 7 shows those exact values as a staircase plot. The continuous curve is the leading term of the optimal sample size s^* as stated by Theorem 13, setting $\xi = 0$ (minimize comparisons) and $\beta(1/2) = 2(1 + \ln 2)$ (use standard Hoare's quickselect algorithm). It is in very good accordance with the exact values, even for small values of n. We conjecture that the $o(\sqrt{n})$ term in s^* is actually $\Theta(1)$.

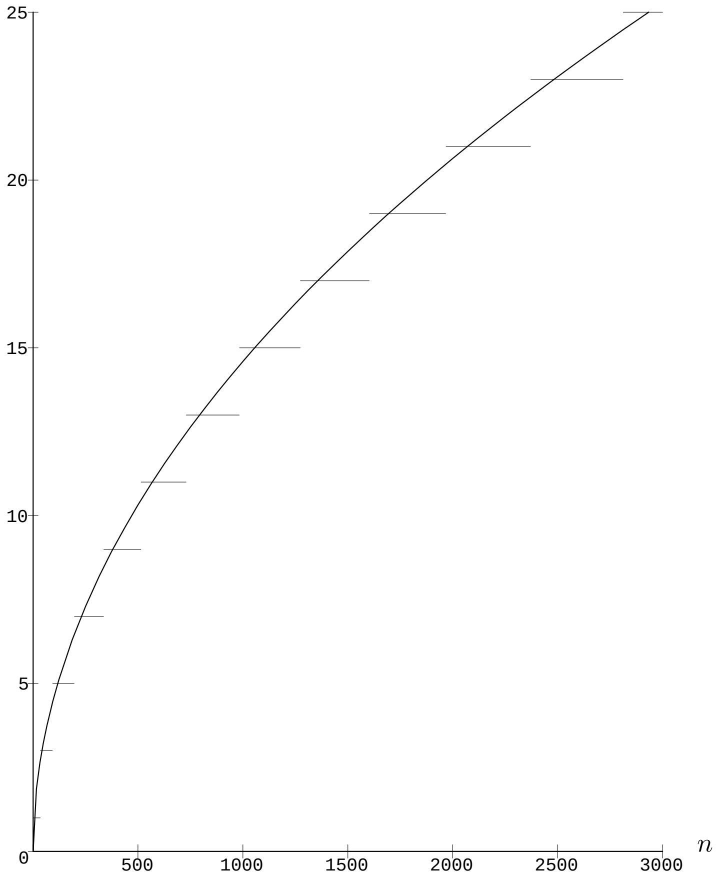

FIG. 7. *Exact values of the optimal samples to minimize the average number of comparisons compared with the main term of s^* (Theorem 13).*

7. Related issues.

7.1. Variance of quickselect. The variance of the number of comparisons of standard quickselect is $\Theta(n^2)$ [14] (see also [9, 18]). Thus, there is a low but nonnegligible probability that the number of comparisons actually made is much larger than the expected number. In this subsection we will study the variance of the number of comparisons of quickselect with sampling, specifically, for the case where the selected pivot is the median of the sample. This analysis is somewhat more complicated than the analysis of its expected performance.

It is worth noting again that we are investigating here the variance of the cost of quickselect when the rank of the sought element is given by a uniformly distributed random variable, not the average of the variances of the costs of selecting each element (see the remark on page 2).

We assume that the median-finding algorithm to select the pivots has quadratic variance. (If the variance were subquadratic, the basic conclusions of this section would not change.) A key observation that makes the analysis relatively simple is that the number of comparisons needed to select the pivot, the number of comparisons made to partition the array, and the number of comparisons made in further invocations of quickselect in smaller subarrays are mutually independent. If the random variable Z_n is the sum of two independent random variables, X_n and Y_n, and $\{A_j\}_{0\le j<n}$ is a family of independent events, then

$$\begin{aligned}\mathbb{E}\big[Z_n^2\big] &= \mathbb{E}\big[(X_n+Y_n)^2\big] = \mathbb{E}\big[X_n^2\big] + 2\,\mathbb{E}[X_n]\,\mathbb{E}[Y_n] + \mathbb{E}\big[Y_n^2\big] \\ &= \mathbb{E}\big[X_n^2\big] + 2\,\mathbb{E}[X_n]\,(\mathbb{E}[Z_n]-\mathbb{E}[X_n]) + \mathbb{E}\big[Y_n^2\big] \\ &= 2\,\mathbb{E}[X_n]\,\mathbb{E}[Z_n] + \mathbb{E}\big[X_n^2\big] - 2\,\mathbb{E}[X_n]^2 + \sum_{0\le j<n} \Pr\{A_j\}\,\mathbb{E}\big[Y_n^2 \mid A_j\big]\,.\end{aligned}$$

Applying the equation above with

$$\begin{aligned}X_n &= \text{number of comparisons made to select the pivot and partition the array,}\\ Y_n &= \text{number of comparisons made in further recursive invocations,}\end{aligned}$$

and A_j = "the pivot is the $(j+1)$th", we can easily obtain recurrences for $F_n^{(2)}$, the second moment of the number of comparisons made by quickselect to find an element of random rank out of n. We first consider the situation where $s = 2k+1 = \Theta(1)$. Then the recurrence for $F_n^{(2)}$ is

$$\begin{aligned}F_n^{(2)} &= 2(n+\Theta(1))F_n - (n+\Theta(1))^2 + \sum_{0\le j<n} w_{n,j}^{(k)} F_j^{(2)} \\ &= \frac{3k+5}{k+1}\,n^2 + \mathcal{O}(n) + \sum_{0\le j<n} w_{n,j}^{(k)} F_j^{(2)}.\end{aligned}$$

The weights and shape function are the same as for the analysis of the expected cost of quickselect with fixed-size samples, but the toll function is now quadratic. Therefore, the limiting const-entropy is

$$\mathcal{H}(k) = 1 - \frac{k+3}{2(2k+3)} = \frac{3(k+1)}{2(2k+3)},$$

and $F_n^{(2)} = (3k+5)(4k+6)n^2/(3(k+1)^2) + o(n^2)$. Finally, subtracting F_n^2 from $F_n^{(2)}$ we get the variance of quickselect.

THEOREM 14. *The variance of the number of comparisons made by quickselect when using samples of fixed size $s = 2k+1$ is*

$$V_n = \frac{2k+3}{3(k+1)^2}\,n^2 + o(n^2) = \left(\frac{2}{3k} + \mathcal{O}\left(\frac{1}{k^2}\right)\right) n^2 + o(n^2).$$

As expected, sampling not only reduces the expected number of comparisons, it also reduces its variance: if $v(k)$ denotes the coefficient of n^2 in V_n, we have $v(0) = 1, v(1) = 5/12, v(2) = 7/27\ldots$, and $\lim_{k\to\infty} v(k) = 0$. This suggests that for samples of increasing size ($k = \omega(1)$, $k = o(n)$) the variance is $o(n^2)$. Indeed, this is what actually happens.

The analysis of this case goes along the same lines as the analysis of the expected value F_n in section 5. Similar techniques to those used in the proof of Theorem 5 yield here that the limiting entropy associated to $F_n^{(2)}$ is $\mathcal{H} = 3/4$. Since the toll function is $3n^2 + o(n^2)$, we have $F_n^{(2)} = 4n^2 + o(n^2)$, and the variance is $V_n = F_n^{(2)} - F_n^2 = 4n^2 + o(n^2) - (2n + o(n))^2 = o(n^2)$.

A refinement of this calculation is possible since we know that, because of (5.1), $F_n = 2n + n/(2k) + \Theta(k) + \Theta\left(n/k^2\right)$. Substituting this estimate into the recurrence for $F_n^{(2)}$ and using $F_j^{(2)} \sim 4j^2$ to compute $\sum_{0\le j<n} w_{n,j}^{(k)} F_j^{(2)}$, we get $F_n^{(2)} = 4n^2 + \Theta\left(\max\left\{nk, n^2/k\right\}\right)$. We thus have the following theorem.

Theorem 15. *The variance V_n of the number of comparisons made by quickselect when the samples are of size $2k+1$, where $k = \omega(1)$ and $k = o(n)$, is $V_n = \mathcal{O}\left(\max\left\{n^2/k, nk\right\}\right)$.*

Notice that the upper bound given above is $o(n^2)$ for any k such that $k = \omega(1)$ and $k = o(n)$. When $k = \Theta(\sqrt{n}\,)$ the variance is $\mathcal{O}(n^{3/2})$ and the standard deviation is $\mathcal{O}(n^{3/4})$. This is the right choice, since it simultaneously minimizes the average number of comparisons (Theorem 8) and the order of magnitude of the variance.

Concerning quicksort, we have not been able to analyze the effect of sampling in the variance of the number of comparisons. We conjecture that for $s = \omega(1)$ and $s = o(n)$ the variance of the number of comparisons satisfies $V_n = \mathcal{O}\left(\max\left\{n^2/s, ns\right\}\right)$, as in Theorem 15.

7.2. Tuning performance. We might avoid redundant comparisons and exchanges as follows [17]: if the input subarray is $A[l\mathinner{..}u]$, we would take the segments $A[l\mathinner{..}l+p]$ and $A[u-q+1\mathinner{..}u]$ as our sample of $s = s(u-l+1)$ elements and apply a slightly modified version of quickselect, which would bring the $(p+1)$th element of the sample to $A[l]$, and put all smaller elements in $A[l+1\mathinner{..}l+p]$ and all larger elements in $A[u-q+1\mathinner{..}u]$. Then the partition of $A[l\mathinner{..}u]$ around the pivot at $A[l]$ would initially set the left scanning pointer to $l+p+1$ and the right scanning pointer to $u-q$. The number of comparisons would then be reduced to $n+2-s$.

This selection-plus-partition combination is more efficient than the standard mechanism assumed in previous sections, but it does not preserve randomness, because the selection of the pivot reorganizes the elements of the sample. There seems not to exist—for general s—an efficient way to perform selection in-place and partition without redundant comparisons and exchanges while preserving randomness. However, we disregard the small amount of sortedness that the selection process introduces and assume that (2.2) and (2.3) are still valid, yielding at least good approximate results.

The steps and computations are absolutely analogous to those of previous sections. The results are, qualitatively speaking, identical. For instance, a straightforward computation—very similar but even simpler than that in the proof of Lemma 1 (see Appendix B)—yields

$$\mathsf{X}(n,s,p) = \frac{(p+1)(q+1)}{(s+1)(s+2)}(n-1-s) \qquad \text{if } s < n.$$

We present here a couple of results for these tuned variants of quickselect and quicksort.

Theorem 16. *Let $s^* = 2k^*+1$ be the optimal sample size w.r.t. the average total cost of the tuned variant of quickselect to select a random element when the median of the sample is used as pivot. Then $k^* = \sqrt{n/(4\beta-4-\xi)} + o\left(\sqrt{n}\,\right)$.*

Theorem 17. *Let s^* be the optimal sample size w.r.t. the average total cost of the tuned variant of quicksort when the $\lceil\psi^*(s+1)\rceil$th element of the sample is used as the pivot. Then*

$$s^* = \sqrt{\frac{q_\xi(\psi^*)/2 - \psi^*(1-\psi^*)\cdot\xi}{\beta(\psi^*) - 1 - \psi^*(1-\psi^*)\cdot\xi}}\cdot\sqrt{n} + o\left(\sqrt{n}\,\right).$$

Appendix A. The CMT.

We briefly review in this appendix the CMT; since no proofs are given here, we refer the reader to [20, 21] for a detailed treatment.

Consider a recurrence like

$$F_n = t_n + \sum_{0 \le j < n} w_{n,j} F_j, \qquad n \ge n_0, \tag{A.1}$$

where t_n is the so-called *toll function* and the quantities $w_{n,j} \ge 0$ are called *weights*. This recurrence is a *divide-and-conquer recurrence* if and only if the following conditions hold:

1. There exists the limiting value $W = \lim_{n\to\infty} \sum_{0\le j<n} w_{n,j}$.
2. The sequence $W_n = \sum_{0\le j<n} w_{n,j}$ converges to W fast enough, namely,

$$|W_n - W| = \mathcal{O}(n^{-c})$$

for some constant $c > 0$.

3. The sequence

$$Z_n = \frac{1}{W_n} \cdot \sum_{0 \le j < n} w_{n,j} \cdot \frac{j}{n}$$

is strictly bounded above by 1 for sufficiently large n.

While the weights $w_{n,j}$ represent the (average) number of recursive calls made to subproblems of size j when the given instance is of size n, W_n is the (average) number of recursive calls made for instances of size n and Z_n is the average size (as a fraction of n) of the subproblems.

THEOREM 18 (CMT). *Let F_n be the solution of* (A.1)*, where $t_n \sim K n^a \log^b n$ for some constants K, $a \ge 0$, and $b > -1$, and let $w(z)$ be a real function over* $[0,1]$ *such that*

$$\sum_{0\le j<n} \left| w_{n,j} - \int_{j/n}^{(j+1)/n} w(z)\,dz \right| = \mathcal{O}(n^{-d})$$

for some constant $d > 0$. Let $\phi(x) = \int_0^1 z^x\, w(z)\,dz$ and define $\mathcal{H} = 1 - \phi(a)$. Then

1. *if $\mathcal{H} > 0$, then $F_n \sim t_n \,/\, \mathcal{H}$;*
2. *if $\mathcal{H} = 0$, then $F_n \sim t_n \ln n \,/\, \mathcal{H}'$, where $\mathcal{H}' = -(b+1)\int_0^1 z^a \,\ln z\, w(z)\,dz$;*
3. *if $\mathcal{H} < 0$, then $F_n = \Theta(n^\alpha)$, where α is the unique real solution of $\phi(x) = 1$.*

The proof of this theorem is based upon more basic and technical results which are also useful to solve or show particular properties of divide-and-conquer recurrences which do not satisfy the hypothesis of the CMT, e.g., for recurrences for which a shape function $w(z)$ cannot be found.

We next consider an example of the results mentioned above. These results have already been used in sections 5 and 6.

Given a function $\gamma_n > 0$ such that $\gamma_n = o(n^\sigma)$ and $\gamma_n = \omega(n^{-\sigma})$ for any $\sigma > 0$ and a toll function t_n, γ_n is a *bounding function* of t_n if and only if there exist a constant $n_\gamma \ge 1$ and a strictly positive nonincreasing function $\beta(z)$ in $[0,1]$ such that $t_n\gamma_n - t_j\gamma_j \ge \beta(j/n)\cdot t_n$ for all $n > n_\gamma$ and for all $n_\gamma \le j < n$.

Given a divide-and-conquer recurrence (A.1) for F_n and a bounding function γ_n of the toll function t_n, the *entropy* $\mathcal{H}_n^{(\gamma)}$ of F_n with respect to γ_n (or γ-entropy of F_n, for short) is

$$\mathcal{H}_n^{(\gamma)} = \gamma_n - \sum_{n_\gamma \le j < n} w_{n,j} \cdot \frac{t_j}{t_n} \cdot \gamma_j.$$

Also, we define the *limiting* γ*-entropy* as $\mathcal{H}^{(\gamma)} = \lim_{n\to\infty} \mathcal{H}_n^{(\gamma)}$ if it exists. Then it is possible to prove that

$$F_n \sim t_n \gamma_n \,/\, \mathcal{H}^{(\gamma)}\,. \tag{A.2}$$

The CMT provides an easy way to determine a bounding function and to compute the limiting value of the corresponding entropy under appropriate circumstances. Notice that $\mathcal{H} \equiv \mathcal{H}^{(1)}$ is the limiting const-entropy and $\mathcal{H}' \equiv \mathcal{H}^{(\ln)}$ is the limiting log-entropy.

Appendix B. Combinatorial identities and proofs. In this appendix we collect several standard results for Beta integrals (see, for instance, [1]) and their discrete analogues, which easily follow from basic results in discrete calculus (see, for instance, [5]). We therefore leave the first two propositions without proof.

We also prove Lemma 1 in this appendix.

PROPOSITION 19.

$$\text{(a)} \qquad \int_0^1 z^\alpha (1-z)^\beta \, dz = \frac{\alpha!\beta!}{(\alpha+\beta+1)!},$$

$$\text{(b)} \qquad -\int_0^1 z^\alpha (1-z)^\beta \ln z \, dz = \frac{\alpha!\beta!}{(\alpha+\beta+1)!}(H_{\alpha+\beta+1} - H_\alpha).$$

For any real x and integer $\ell \ge 0$, $x^{\underline{\ell}} = x\cdot(x-1)\cdots(x-\ell+1)$ will denote the ℓth *falling factorial* of x [5]. Many of the computations in this work involve evaluating sums of the type

$$\sum_{0\le j<n} g(j,n)\binom{j}{p}\binom{n-1-j}{q} = \frac{1}{p!\,q!}\sum_{0\le j<n} g(j,n)\, j^{\underline{p}}\,(n-1-j)^{\underline{q}}$$

for some appropriate function $g(j,n)$. For instance, to evaluate $\mathsf{X}(n,s,p)$ we use $g = j(n-1-j)$, for $\mathcal{F}(n,k)$ we use $g = j^2$, and for $\mathcal{Q}(n,\psi)$ we use $g = jH_j$.

Proposition 20 is helpful in coping with these sums. Notice the parallelism with Proposition 19.

PROPOSITION 20.

$$\text{(a)} \quad A_{\ell,m}(n) = \sum_{0\le j<n} j^{\underline{\ell}}(n-1-j)^{\underline{m}} = \ell!\; m!\binom{n}{\ell+m+1},$$

$$\text{(b)} \quad B_{\ell,m}(n) = \sum_{0\le j<n} j^{\underline{\ell}}(n-1-j)^{\underline{m}} H_j = \ell!\; m!\binom{n}{\ell+m+1}\left(H_n - H_{\ell+m+1} + H_\ell\right).$$

The following intermediate result, which is used in section 5, is just an example of the usefulness of the proposition above. Similar intermediate results in the analysis of quicksort (section 6) follow also from Proposition 20(b).

PROPOSITION 21. *For all s and p such that $0 \le p < s$,*

$$(s+1)(s+2) \sum_{0\le j<n} j^2 \left(\pi_{n,j}^{(s,p)} + \pi_{n,n-1-j}^{(s,p)}\right) = \big((p+1)(p+2) + (q+1)(q+2)\big)n^2 \\ - 6(p+1)(q+1)n + (p-q)^2 + (s+1).$$

Proof. Apply Proposition 20(a) twice, using $j^2 = (j-x)(j-1-x) + (2x+1)(j-x) + x^2$, with $x = p$ the first time and with $x = q$ the second time. □

We end this appendix with a proof of Lemma 1, which gives a closed expression for $\mathsf{X}(n,s,p)$.

Proof of Lemma 1. Here we assume that the subarray to be partitioned contains the pivot in its first position. If the pivot ends at $A[j+1]$ after partitioning and there were t elements in $A[2\,..\,j+1]$ greater than the pivot, then exactly t swaps were needed to arrange the array. (We do not count the one or two additional swaps just before and after the partitioning.) The probability of this to happen is

$$\frac{\binom{j}{j-t}\binom{n-1-j}{t}}{\binom{n-1}{j}},$$

by an argument largely analogous to the one used to obtain the value of $\pi_{n,j}^{(s,p)}$ in (2.1). Thus

$$\mathsf{X}(n,s,p) = \sum_{0\le j<n} \pi_{n,j}^{(s,p)} \sum_t t \cdot \frac{\binom{j}{j-t}\binom{n-1-j}{t}}{\binom{n-1}{j}}.$$

We can now use the "derivative" of Vandermonde's convolution [5],

$$\sum_t t\binom{b}{c-t}\binom{a}{t} = a\binom{a+b-1}{c-1},$$

to get

$$\mathsf{X}(n,s,p) = \sum_{0\le j<n} \pi_{n,j}^{(s,p)} \cdot \frac{j(n-1-j)}{(n-1)}.$$

Finally, using the equality $j(n-1-j) = (j-p)(n-1-j-q) + (j-p)(q-p) + (pn-p^2-p)$ yields

$$\mathsf{X}(n,s,p) = \frac{A_{p+1,q+1}(n) + (q-p)\cdot A_{p+1,q}(n) + (pn-p^2-p)\cdot A_{p,q}(n)}{p!\,q!\,(n-1)\binom{n}{s}},$$

where $P_{p,q}(n)$ is as in Proposition 20(a). The rest of the proof is a simple matter of computation. □

Acknowledgments. We thank H.-K. Hwang for various useful comments and suggestions, especially for encouraging us to study the variance of quicksort and quickselect. We also wish to thank the anonymous referees of this paper for their comments and useful suggestions.

REFERENCES

[1] M. Abramowitz and I. Stegun, eds., *Handbook of Mathematical Functions*, Dover, New York, 1964.

[2] J. Bentley, *Programming Pearls*, Addison-Wesley, Reading, MA, 1986.

[3] J. Bentley and M. McIlroy, *Engineering a sort function*, Software—Practice and Experience, 23 (1993), pp. 1249–1265.

[4] R. Floyd and R. Rivest, *Expected time bounds for selection*, Comm. ACM, 18 (1975), pp. 165–173.

[5] R. Graham, D. Knuth, and O. Patashnik, *Concrete Mathematics*, 2nd ed., Addison-Wesley, Reading, MA, 1994.

[6] R. Grübel, *On the median-of-k version of Hoare's selection algorithm*, Theor. Inform. Appl., 33 (1999), pp. 177–192.

[7] C. Hoare, *Algorithm* 65: Find, Commu. ACM, 4 (1961), pp. 321–322.

[8] C. Hoare, *Quicksort*, Computer Journal, 5 (1962), pp. 10–15.

[9] P. Kirschenhofer and H. Prodinger, *Comparisons in Hoare's Find algorithm*, Combin. Probab. Comput., 7 (1998), pp. 111–120.

[10] P. Kirschenhofer, H. Prodinger, and C. Martínez, *Analysis of Hoare's* Find *algorithm with median-of-three partition*, Random Structures Algorithms, 10 (1997), pp. 143–156.

[11] D. Knuth, *Mathematical analysis of algorithms*, in Information Processing '71, North-Holland, Amsterdam, 1972, pp. 19–27.

[12] D. Knuth, *The Art of Computer Programming: Sorting and Searching*, Vol. 3, 2nd ed., Addison-Wesley, Reading, MA, 1998.

[13] H. Mahmoud, *Sorting: A Distribution Theory*, John Wiley and Sons, New York, 2000.

[14] H. M. Mahmoud, R. Modarres, and R. T. Smythe, *Analysis of quickselect: An algorithm for order statistics*, Theor. Inform. Appl., 29 (1995), pp. 255–276.

[15] C. Martínez and S. Roura, *Optimal sampling strategies in quicksort*, in Proceedings of the 25th International Colloquium on Automata, Languages, and Programming (ICALP'98), K. Larsen, S. Skyum, and G. Winskel, eds., Lecture Notes in Comput. Sci. 1443, Springer-Verlag, Berlin, 1998, pp. 327–338.

[16] C. Martínez and S. Roura, *Optimal Sampling Strategies in Quicksort and Quickselect*, Tech. Rep. LSI-98-1-R, LSI-UPC, 1998; also available online from www.lsi.upc.es/dept/techreps/1998.html.

[17] C. McGeoch and J. Tygar, *Optimal sampling strategies for quicksort*, Random Structures Algorithms, 7 (1995), pp. 287–300.

[18] A. Panholzer and H. Prodinger, *A generating functions approach for the analysis of grand averages for multiple QUICKSELECT*, Random Structures Algorithms, 13 (1998), pp. 189–209.

[19] H. Prodinger, *Multiple Quickselect—Hoare's Find algorithm for several elements*, Inform. Process. Lett., 56 (1995), pp. 123–129.

[20] S. Roura, *Divide-and-Conquer Algorithms and Data Structures*, Ph.D. thesis, Dept. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 1997.

[21] S. Roura, *Improved master theorems for divide-and-conquer recurrences*, J. ACM, 48 (2001), pp. 170–205.

[22] R. Sedgewick, *The analysis of quicksort programs*, Acta Inform., 7 (1976), pp. 327–355.

[23] R. Sedgewick, *Implementing quicksort programs*, Comm. ACM, 21 (1978), pp. 847–856.

[24] R. Sedgewick, *Quicksort*, Garland, New York, 1978.

[25] R. Singleton, *Algorithm* 347*: An efficient algorithm for sorting with minimal storage*, Comm. ACM, 12 (1969), pp. 185–187.

[26] M. van Emden, *Increasing the efficiency of quicksort*, Comm. ACM, 13 (1970), pp. 563–567.