Journal of Algorithms **30,** 423–448 (1999)
Article ID jagm.1998.0986, available online at http://www.idealibrary.com on IDEAL®

The Cost Distribution of Queue-Mergesort, Optimal Mergesorts, and Power-of-2 Rules

Wei-Mei Chen

Department of Computer Science and Information Engineering, National Taiwan University, Taipei 107, Taiwan

Hsien-Kuei Hwang*

Institute of Statistical Science, Academia Sinica, Taipei 115, Taiwan

and

Gen-Huey Chen

Department of Computer Science and Information Engineering, National Taiwan University, Taipei 107, Taiwan

Received August 6, 1998; revised September 22, 1998

Queue-mergesort is introduced by Golin and Sedgewick as an optimal variant of mergesorts in the worst case. In this paper, we present a complete analysis of the cost distribution of queue-mergesort, including the best, average, and variance cases. The asymptotic normality of its cost is also established under the uniform permutation model. We address the corresponding optimality problems and we show that if we fix the merging scheme then the optimal mergesort as far as the average number of comparisons is concerned is to divide as evenly as possible at each recursive stage (top-down mergesort). On the other hand, the variance of queue-mergesort reaches asymptotically the minimum value. We also characterize a class of mergesorts with the latter property. A comparative discussion is given on the probabilistic behaviors of top-down mergesort, bottom-up mergesort, and queue-mergesort. We derive an ‘‘invariance principle’’ for asymptotic linearity of divide-and-conquer recurrences based on general ‘‘power-of-2’’ rules of which the underlying dividing rule of queue-mergesort is a special case. These analyses reveal an interesting algorithmic feature for general power-of-2 rules.

* E-mail: hkhwang@stat.sinica.edu.tw.

0196-6774/99 $30.00

1. INTRODUCTION

Mergesort is one of the very first sorting algorithms in computer science and is usually the method of choice for sorting linked lists (like in MAPLE). It is a typical example solved by divide-and-conquer paradigm and admits several variants according to different underlying dividing rules: top-down mergesort, bottom-up mergesort, and queue-mergesort (cf. [10, Section 5.2.4; 14, Chap. 8; 2; 12; 5]). The number of comparisons used by these alternatives for sorting n elements is expressed by the recurrence $f_1 := 0$ and

$$f_n = f_{\tau(n)} + f_{n-\tau(n)} + g_n, \qquad (n \geq 2),$$

where g_n is the merging cost and

— $\tau(n) = \lfloor n/2 \rfloor$ (half–half rule) in top-down mergesort (TDM; cf. [2]);

— $\tau(n) = 2^{\lceil \log_2 n/2 \rceil}$ (max power-of-2 rule) in bottom-up mergesort (BUM; cf. [12]), and

— $\tau(n) = 2^{\lfloor \log_2 2n/3 \rfloor}$ (balanced power-of-2 rule) in queue-mergesort (QM; cf. [5]).

Note that $2^{\lfloor \log_2 2n/3 \rfloor}$ is the unique power-of-2 lying between $n/3$ and $2n/3$ and that the choice of rationals other than $\frac{2}{3}$ is not more balanced. For example, if $\tau(n) = 2^{\lfloor \log_2 n/2 \rfloor}$ or $\tau(n) = 2^{\lfloor \log_2 5n/9 \rfloor}$, then the sizes of two subproblems are (2, 5) for $n = 7$, while the balanced power-of-2 gives (3, 4).

Among these dividing rules (not restricted to mergesort), the half–half rule is undoubtedly the most widely used one; it is almost the synonym of divide-and-conquer in several problems. On the other hand, the balanced power-of-2 rule appeared in considerably fewer problems, its usefulness being usually neglected. We briefly indicate some of the major problems in which this rule appeared. The associated recurrence is essentially the *heap recurrence* (see [9] or next section for details),

$$f_n = f_{2^{\lfloor \log_2 2n/3 \rfloor}} + f_{n - 2^{\lfloor \log_2 2n/3 \rfloor}} + g_n, \qquad (n \geq 2),$$

(with f_1 given) because it corresponds modulo shift to the sizes of the left and right subheaps of a heap of size n. The recurrence was first (implicitly) studied by Knuth [10] (cf. [9]). It appeared as the solution to the recurrence with minimization (cf. Hammersley and Grimmett [7]): $f_1 := 0$ and for $n \geq 2$,

$$f_n := \min_{1 \leq j < n} \{f_j + f_{n-j}\} + g_n, \tag{1}$$

where g_n is increasing and concave; cf. also [1, 14, 16]. Glassey and Karp [4] independently discussed this rule referred to as power-of-2 rule. For reason of distinction, we add the adjective "balanced." Walsh [16] developed a mergesort algorithm with the same number of comparisons as TDM in the worst case (thus optimal). A similar version with different merging orders named *queue-mergesort* was introduced by Golin and Sedgewick [5], which is also optimal in the worst case. We show in the next section that the underlying dividing principle in Walsh's and Golin and Sedgewick's mergesorts is essentially the balanced power-of-2.

This variant of mergesort enjoys several intriguing probabilistic properties which remain unknown and it is the purpose of this paper to prove them by suitable analytic tools. For example, we show that the variance of the number of comparisons used by QM is asymptotically linear, the leading constant being approximately 0.3073 (instead of a periodic function as in TDM). Furthermore, we show that this constant is also asymptotically optimal (minimal) for mergesorts using two-way linear merge (cf. [10, 14]) as the underlying merging procedure; see Section 6.2 for details. A general class of mergesorts with the same property is also characterized. Briefly, although the mean value of QM is slightly higher in the linear term than that of TDM, the global "silhouette" of its stochastic behavior is more smooth than those of TDM and of BUM.

From a practical viewpoint, QM is easily implemented on linked list, its code being simpler than those of TDM and BUM. Also the size of the input need not be known in advance and it can be implemented in either a top-down or a bottom-up manner, making QM an attractive alternative to TDM and BUM. The price we pay is *stability*: QM is not stable for equal keys.

Periodic fluctuation is usually an accompaniment of divide-and-conquer recurrences, especially when more precise asymptotic approximations are needed. QM is no exception. Our approach to handling periodicity is based on the exact formula derived in [9] for heap recurrence and on digital sums. This *elementary*[1] approach is in certain respects simpler than the *analytic* one for TDM for which Mellin transform and complex analysis are required; cf. [2, 8]. To keep our analyses as uniform as possible, we apply the same elementary approach to prove the asymptotic normality (in the sense of convergence in distributions) of the cost of QM. This in turn introduces some technical problems for which new arguments are developed.

The study of divide-and-conquer recurrences is closely related to the solution of recurrence with minimization (or maximization) or the generalized subadditive inequality (cf. [1, 7]). For example, Hammersley and

[1] In number-theoretic sense, namely, without recourse to complex analysis.

Grimmett [7] showed that the minimum of the recurrence (1) is attained at (i) $j = 1$ if g_n is decreasing; (ii) $j = \lfloor n/2 \rfloor$ if g_n is increasing and convex; and (iii) $j = 2^{\lfloor \log_2 2n/3 \rfloor}$ if g_n is increasing and concave. Thus TDM and QM are natural candidates if one is concerned with optimality of certain characteristics on divide-and-conquer problems.

In the next section, the link of the cost of QM with the heap recurrence is first proved. An interesting algorithmic feature of power-of-2 rules is revealed. We then analyze the cost of QM in the worst, best, average, and variance cases. A comparative discussion is also given on the costs of TDM, BUM, and QM. The asymptotic normality of the cost under the uniform permutation model is established in Section 4. An interesting invariance principle for the asymptotic linearity of divide-and-conquer recurrences based on general power-of-2 rule is proved in Section 5. We then identify, with the aid of the invariance principle, optimal mergesorts in the average and variance cases in Section 6. For the reader's convenience, an appendix is also given on the exact solutions of the divide-and-conquer recurrences discussed in this paper.

Notation. Throughout this paper, all unspecified limits (including O, $\sim$, o) will be taken to be $n \to \infty$. The binary representation of n will be written consistently as $(1b_{L-1} \cdots b_0)_2$, where $L = \lfloor \log_2 n \rfloor$. We also define $n_j = (1b_{j-1} \cdots b_0)_2 = 2^j(1 + \{n/2^j\})$ for $j = 0, \ldots, L$; so that $n_L = n$ and $n_0 = 1$. Write for $n \geq 2$ $\rho = \rho(n) = \min\{2^{\lfloor \log_2 2n/3 \rfloor}, n - 2^{\lfloor \log_2 2n/3 \rfloor}\}$ and $\lambda = \lambda(n) = n - \rho$.

2. QUEUE-MERGESORT AND THE HEAP RECURRENCE

In this section, we first describe the QM algorithm and then we prove its relation with heap recurrence. Briefly, start with n elements each in its own list and arrange these lists as a queue. Then take the first two lists, merge them by the linear merge algorithm, and put the new list at the tail of the queue. Repeat the preceding steps until the single list remains in the queue. The list will contain all of the elements in sorted order. An example is given in Fig. 1.

This algorithm is reminiscent of the well-known Huffman coding procedure.

Walsh's mergesort [16] is similar but proceeds in a different order. First merge the rightmost with the leftmost lists, putting the resulting list in the leftmost; then merge the remaining lists in adjacent pairs from the second to leftmost to right. Iterate this procedure until a single sorted list remains. Obviously, the algorithm is, up to different merging orders, the same as QM.

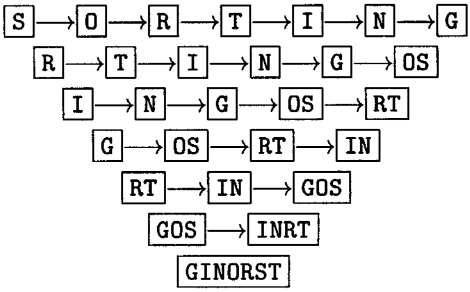

FIG. 1

The following lemma says that QM may also be implemented in a top-down manner, which is useful for sorting arrays; the dividing strategy is the balanced power-of-2 rule.

LEMMA 1. *The cost of QM is generally described by the heap recurrence,*

$$f_n = f_\lambda + f_\rho + g_n, \qquad (n \geq 2), \tag{2}$$

with a suitable f_1, where g_n denotes the merging cost.

Proof. The proof follows essentially the Huffman coding argument as that given in Section 4 of Glassey and Karp [4] by "describing" the list sizes at each stage. Assume the queue sizes at the ith run are given by $a_{i,i} \leq a_{i,i+1} \leq \cdots \leq a_{i,n}$. Initially, we have $a_{1,j} = 1$ for $j = 1, \ldots, n$, the final state being $a_{n,n} = n$. At each iteration, QM transforms $(a_{i,i}, \ldots, a_{i,n})$ into $(a_{i+1,i+1}, \ldots, a_{i+1,n})$, where $a_{i+1,j} = a_{i,j+1}$ for $j = i+1, \ldots, n-1$ and $a_{i+1,n} = a_{i,i} + a_{i,i+1}$. By induction, the sequences $\{a_{i,j}\}$ satisfy the following properties:

— There exists an integer k such that $2^k \leq a_{i,j} \leq 2^{k+1}$ for $j = i, \ldots, n$;

— $\frac{1}{2} \leq a_{i,j}/a_{i,j+1} \leq 1$ for $j = i, \ldots, n-1$;

— At most one $a_{i,j}$ (i fixed) is not a power-of-2.

These properties are preserved by the merging transformation. It follows that $a_{n-1,n-1} = \rho$ and $a_{n-1,n} = \lambda$. ∎

Note that the recurrence (2) can be written in the following form,

$$f_{2^L+j} = \begin{cases} f_{2^{L-1}} + f_{2^{L-1}+j} + g_{2^L+j}, & \text{if } 0 \leq j < 2^{L-1}, \\ f_{2^L} + f_j + g_{2^L+j}, & \text{if } 2^{L-1} \leq j < 2^L. \end{cases}$$

The recurrence (2) has been studied in [9] where an exact solution is given.

LEMMA 2 (Hwang and Steyaert). *The solution f_n of the heap recurrence* (2) *with $f_1 = g_1$ is given by*

$$f_n = \sum_{1 \le j \le L} \left(\left\lfloor \frac{n}{2^{j-1}} \right\rfloor - \left\lfloor \frac{n}{2^j} \right\rfloor - 1 \right) g_{2^j} + \sum_{0 \le j \le L} g_{n_j}, \qquad (n \ge 1), \quad (3)$$

for any given sequence $\{g_n\}$.

For completeness, we give an alternative proof. The proof in [9] is "combinatorial" and relies on a counting argument from [10]; our proof is "computational" and can be applied to max and other power-of-2 rules (see Section 5).

Proof. First, the recurrence (2) can be written in the following form,

$$f_n = f_{n_{L-1}} + f_{2^{L-1}} + g_{n_L} + b_{L-1}(f_{2^{L-1}} + g_{2^L}).$$

Iterating once yields

$$f_n = f_{n_{L-2}} + f_{2^{L-2}} + f_{2^{L-1}} + g_{n_{L-1}} + g_{n_L} + b_{L-2}(f_{2^{L-2}} + g_{2^{L-1}})$$
$$+ b_{L-1}(f_{2^{L-1}} + g_{2^L}).$$

Thus by induction,

$$f_n = \sum_{0 \le j \le L-1} (1 + b_j) f_{2^j} + \sum_{1 \le j \le L} b_{j-1} g_{2^j} + \sum_{0 \le j \le L} g_{n_j}.$$

Using the explicit expressions $b_j = \lfloor n/2^j \rfloor - 2\lfloor n/2^{j+1} \rfloor$,

$$f_{2^m} = \sum_{0 \le i \le m} 2^{m-i} g_{2^i}, \quad \sum_{j \le i \le L-1} (1 + b_i) 2^{i-j} = \lfloor n/2^j \rfloor - 1, \qquad (4)$$

we obtain exactly (3). ∎

In certain cases the use of the following formula is preferable,

$$f_n = \sum_{0 \le j \le L} \left\lfloor \frac{n}{2^j} \right\rfloor (g_{2^{j+1}} - g_{2^j}) + \sum_{0 \le j \le L} (g_{n_j} - g_{2^{j+1}}) + g_1 n. \qquad (5)$$

For example, taking $g_1 = 0$ and $g_n = an + c$ for $n \ge 2$, we obtain

$$f_n = anL - 2^{L+1} a + 2an + c(n-1), \qquad (n \ge 2). \qquad (6)$$

LEMMA 3. *Assume that* f_n *satisfies* (2). *Then* $f_n/n \to l < \infty$ *iff* $g_n = o(n)$ *and* $\sum_{j \geq 0} g_{2^j}/2^j = l$.

Proof. The "if" part is easy (cf. [9]). We prove the "only if" part. Assume that $f_n/n \to l$. Then by definition,

$$g_n = f_n - f_\rho - f_\lambda = o(n).$$

It follows from (3) that

$$\sum_{1 \leq j \leq L} \left(\left\lfloor \frac{n}{2^{j-1}} \right\rfloor - \left\lfloor \frac{n}{2^j} \right\rfloor \right) g_{2^j} = ln + o(n).$$

From this we deduce that $\sum_{0 \leq j \leq L} g_{2^j}/2^j \to l$. ∎

In particular, when $g_n = O(1)$, we have a more precise error term.

LEMMA 4. *If* f_n *satisfies* (2) *and* $g_n = O(1)$, *then* $f_n = ln + O(\log n)$, *where* $l = \sum_{j \geq 0} g_{2^j}/2^j$.

Similar results hold for very general power-of-2 rules; see Section 5.

Note that the leading constant is never periodic when f_n is linear, in contrast to the half−half recurrence (cf. [2, 8, 12]).

These lemmas also reveal, in a decidedly way, another advantage of power-of-2 rules over the half−half rule: *the cost may remain linear even when the cost of half−half rule is superlinear*. For example, if $g_n = \Theta(n/\log n)$ for $n \neq 2^L$ and $g_{2^L} = O(2^L/L^{1+\varepsilon})$, then f_n is linear by the previous lemmas. But the same g_n gives rise to $O(n \log\log n)$ bound for the half−half recurrence, which is tight for almost all values of n. This leads to the following important *algorithmic* implication. Assume that the "merging" cost is of order $o(n)$. Then the total cost will be linear if we can improve problems of sizes 2^L (and these only!) so that $\sum_{j \geq 0} g_{2^j}/2^j < \infty$. No such simple improvement is available for the half−half rule as is easily seen from its solution (17). Note that the power-of-2 rules discussed in this paper all satisfy the property that they evenly divide into two when $n = 2^L$.

3. THE ANALYSIS OF QUEUE-MERGESORT

In this section we study the cost (i.e., the number of comparisons) of QM in most cases of interests: the worst, the best, the average, and the variance cases. We also compare the results with those of TDM and BUM. A summary of these comparisons is given in Table 1.

TABLE I
Asymptotic Behaviors of the Three Mergesorts; the Best Case is the Same and Thus not Shown Here.[a]

	Worst case	Average case	Variance	Asymptotic normality (?)
QM	$n \log_2 n - 0.943n$	$n \log_2 n - 1.207n$	$0.307n$	Yes
TDM	$n \log_2 n - 0.943n$	$n \log_2 n - 1.248n$	$0.345n$	Yes
BUM	$n \log_2 n - 0.701n$	$n \log_2 n - 0.965n$	$O(n^2)$	No

[a] The constants of the linear terms in the first two columns and the variance of TDM are mean values of certain periodic functions.

We need the following expressions for the costs of the two-way linear merge for merging two sorted files of sizes x and y (cf. [2, 10]):

— Best case: $\min\{x, y\}$;
— Worst case: $x + y - 1$;
— Average case,

$$u(x, y) := x + y - \frac{x}{y+1} - \frac{y}{x+1}; \tag{7}$$

— Variance case,

$$v(x, y) = \frac{x(2x+y)}{(y+1)(y+2)} + \frac{y(2y+x)}{(x+1)(x+2)} - \left(\frac{x}{y+1} + \frac{y}{x+1}\right)^2. \tag{8}$$

We assume in the last two formulae that each of the $\binom{x+y}{x}$ possible orderings is equally likely.

3.1. *Worst Case*

The number of comparisons used by QM is optimal in the worst case (cf. [5]). It is given by $W_1 = 0$, and

$$W_n = W_\rho + W_\lambda + n - 1, \qquad (n \geq 2),$$

By (6), W_n satisfies (cf. [2, 5]),

$$W_n = n \log_2 n + nA(\log_2 n) + 1, \qquad (n \geq 1), \tag{9}$$

where $A(t)$ is a continuous periodic function of period 1,

$$A(t) = 1 - \{t\} - 2^{1-\{t\}}, \tag{10}$$

whose mean value is $\frac{1}{2} - 1/\log 2$.

Actually, from Hammersley and Grimmett's result in [7] and the sum expression (cf. [2, 10]) $W_n = \sum_{1 \le j \le n} \lceil \log_2 j \rceil$, we deduce that any mergesort using two-way linear merge such that the dividing rule $(n \mapsto (j, n-j))$ satisfies $\rho \le j \le \lfloor n/2 \rfloor$ at each recursive stage is worst case optimal. Thus we have a spectrum of *optimal mergesorts* of which TDM and QM are the "boundaries." BUM is however not optimal for general values of n; see [5] for a modification.

3.2. *Best Case*

The best case cost B_n of QM satisfies the recurrence (2) with

$$g_n = \min(\lambda, \rho) = \rho = \begin{cases} 2^{L-1}, & \text{if } b_{L-1} = 0, \\ 2^L \left\{ \dfrac{n}{2^L} \right\} = n - 2^L, & \text{if } b_{L-1} = 1, \end{cases}$$

By (2) and (3), it can be verified that $B_1 = 0$ and for $n \ge 1$,

$$\begin{cases} B_{2n} = 2B_n + n, \\ B_{2n+1} = B_n + B_{n+1} + n; \end{cases}$$

so we obtain $B_n = \sum_{0 \le j < n} \nu(j)$, where $\nu(j)$ denotes the sum-of-digits function of the binary representation of n. Its asymptotic behavior is well known (cf. [3] and the references therein): The cost function B_n of QM in the best case satisfies

$$B_n = \frac{1}{2} n \log_2 n + nD(\log_2 n),$$

where $D(u)$ is a continuous, nowhere differentiable, periodic function of period 1.

The best case costs of the three mergesorts are the same. This implies that the recurrence $C_1 := 0$ and (cf. [6, Section 2.2.1]),

$$C_n = \max_{0 \le j < n} \left(C_j + C_{n-j} + \min\{j, n-j\} \right), \qquad (n \ge 2)$$

has at least three generally different indices at which the maximum is attained: $j = \lfloor n/2 \rfloor$, $j = 2^{\lceil \log_2 n/2 \rceil}$, and $j = 2^{\lfloor \log_2 2n/3 \rfloor}$ (because $C_n = B_n$). A complete description of the indices attaining the maximum is an interesting problem but lies outside the scope of this paper.

3.3. *Average Case*

Henceforth, we assume that each of the $n!$ permutations of n elements is equally likely. Denote by X_n the number of comparisons used by QM to sort a random permutation.

THEOREM 1. *The average number of comparisons used by QM to sort a random permutation of n elements satisfies*

$$U_n^{[q]} := E(X_n) = n\log_2 n + \big(A(\log_2 n) - \alpha\big)n + 4\varpi_1(\log_2 n) + O(1),$$

where $\alpha = \sum_{j\geq 0} 1/(2^j+1) - 1 \approx 0.26449978$. $A(t)$ *is defined in* (10) *and* $\varpi_1(u)$ *is oscillating between* $O(u)$ *and* $\Omega(1)$ *defined by*

$$\varpi_1(u) = \sum_{0\leq j\leq \lfloor u\rfloor} \frac{\left(\{2^{u-j}\} - \{2^{u-j-1}\}\right)^2}{\left(1+\{2^{u-j}\}\right)\left(1-\{2^{u-j}\}+2\{2^{u-j-1}\}\right)}, \qquad (u\geq 1).$$

Proof. By (7), $U_n^{[q]}$ satisfies (2) with

$$g_n = u(\rho,\lambda) = n - \frac{\rho}{\lambda+1} - \frac{\lambda}{\rho+1}.$$

It is evident that

$$\frac{\rho}{\lambda+1} + \frac{\lambda}{\rho+1} = O(1),$$

By (6) and Lemma 4, we obtain

$$U_n^{[q]} := E(X_n) = n\log_2 n + \big(A(\log_2 n) - \alpha\big)n + O(\log n).$$

To make explicit the $O(\log n)$ term, we start from (5) and proceed as follows. Observe that the first sum will not contribute any term of logarithmic order. It suffices to investigate the sum,

$$\sum_{0\leq j\leq L} \left(g_{2^j} - g_{n_j}\right),$$

when $g_n = \lambda/(\rho+1) + \rho/(\lambda+1)$. By considering the value of b_{j-1}, we have (recall that $n_j = (1b_{j-1}\cdots b_0)_2 = 2^j(1+\{n/2^j\})$),

$$\begin{aligned} g_{n_{j+1}} &= \frac{n_j}{2^j(1+b_j)+1} + \frac{2^j(1+b_j)}{n_j+1} \\ &= \frac{1+\{n/2^j\}}{1+b_j} + \frac{1+b_j}{1+\{n/2^j\}} + O(2^{-j}). \end{aligned}$$

From the expression $b_j = 2\{n/2^{j+1}\} - \{n/2^j\}$, it follows that

$$g_{n_{j+1}} = 2 - 4\frac{\left(\{n/2^j\} - \{n/2^{j+1}\}\right)^2}{\left(1 + \{n/2^j\}\right)\left(1 - \{n/2^j\} + 2\{n/2^{j+1}\}\right)} + O(2^{-j}).$$

Consequently,

$$\sum_{0 \le j \le L} (g_{2^j} - g_{n_j})$$

$$= 4 \sum_{0 \le j \le L} \frac{\left(\{n/2^j\} - \{n/2^{j+1}\}\right)^2}{\left(1 + \{n/2^j\}\right)\left(1 - \{n/2^j\} + 2\{n/2^{j+1}\}\right)} + O(1).$$

The upper bound for $\varpi_1(\log_2 n)$ is tight as is seen by the integer sequence $\{\frac{4}{3}(4^k - 1)\}_k$. ▮

Note that $\varpi_1(1 + \log_2 n) = \varpi_1(\log_2 n)$. A graphical rendering of $\varpi_1(u)$ is given in Fig. 2.

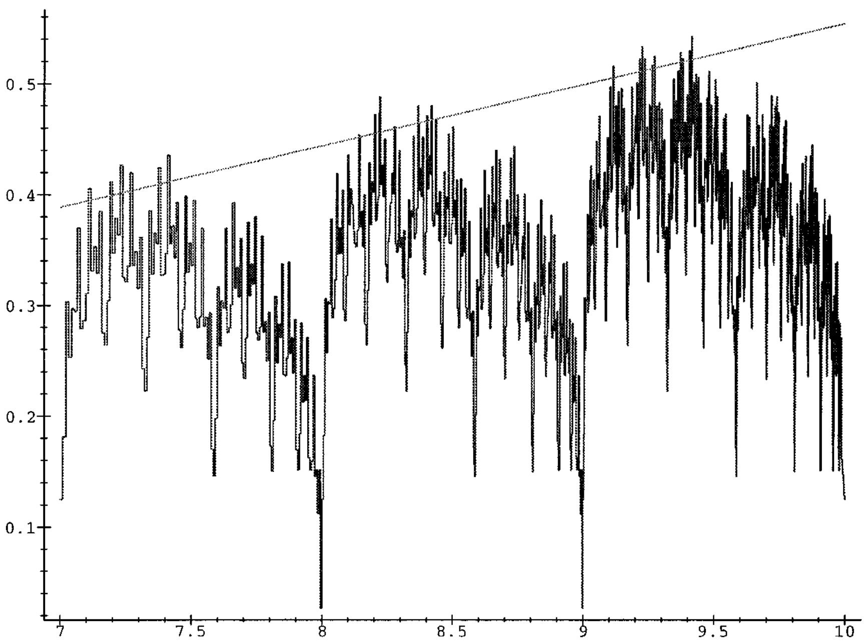

FIG. 2. Periodic fluctuations of the function $\varpi_1(\log_2 n)$ versus $(\log_2 n)/18$ in logarithmic scale.

Comparing the foregoing result with those in [2, 12] (cf. (17) and (18)), we conclude that the average cost of QM lies between that of TDM and of BUM and that the average difference between that of TDM and QM is about $0.039n$.

3.4. *Variance*

In this subsection, we consider the variance of X_n.

THEOREM 2. *The variance of the number of comparisons used by QM to sort a random permutation of n elements satisfies*

$$V_n^{[q]} := \operatorname{Var}(X_n) = \beta n + 4\varpi_2(\log_2 n) + O(1),$$

where $\beta = \sum_{j\geq 0}[2^j(2^j-1)/(2^j+1)^2(2^j+2)] \approx 0.3073049590$ *and* $\varpi_2(u)$ *is oscillating between* $O(u)$ *and* $\Omega(1)$ *defined by*

$$\begin{aligned}\varpi_2(u) &\\ &= \sum_{0\leq j\leq \lfloor u\rfloor} \times \frac{\left(\{2^{u-j}\} - \{2^{u-j-1}\}\right)^2 \left(5\left(\{2^{u-j-1}\}+1\right)^2 - \left(\{2^{u-j}\} - \{2^{u-j-1}\}\right)^2\right)}{\left(1+\{2^{u-j}\}\right)^2\left(1-\{2^{u-j}\}+2\{2^{u-j-1}\}\right)^2}.\end{aligned}$$

for $u \geq 1$.

Proof. (Sketch). The variance $V_n^{[q]}$ satisfies (2) with (cf. (8)),

$$\begin{aligned}g_n &= v(\rho, \lambda)\\ &= \frac{\rho(2\rho+\lambda)}{(\lambda+1)(\lambda+2)} + \frac{\lambda(2\lambda+\rho)}{(\rho+1)(\rho+2)} - \left(\frac{\rho}{\lambda+1} + \frac{\lambda}{\rho+1}\right)^2.\end{aligned}$$

We then apply *mutatis mutandis* the method of proof of Theorem 1. ∎

Note that

$$\beta = -2\alpha + 2\sum_{j\geq 0} \frac{1}{(2^j+1)^2}.$$

Unlike TDM analyzed in [2], the variance of X_n is not oscillating in the dominant term; it is smaller than that of TDM for large enough n (with exceptions at $2^k \pm 2, 3, 4$ for $k \geq 7$). On the other hand, the variance of the cost of BUM is $O(n^2)$ as is easily seen by considering the case $n = 2^k + c$, where $1 \leq c = O(1)$; see (18). [Actually, it oscillates between $O(n^2)$ and $\Omega(n)$.]

Higher cumulants of X_n can be considered in a similar way and are all asymptotically linear. This suggests that the distribution of X_n is asymptotically normal as we prove in the next section. In particular, we can show that the third cumulant is asymptotic to $-\eta n$ with $\eta = \sum_{j\ge 1} 2^j(3\cdot 2^j - 1)(2^j - 1)^2/[(2^j + 1)^3(2^j + 2)(2^j + 3)] > 0$, implying that the distribution is skew to the left.

4. ASYMPTOTIC NORMALITY

In this section we prove the asymptotic normality of the cost of QM. Let $P_n(z)$ denote the probability generating function of X_n. Then $P_n(z)$ satisfies the recurrence,

$$\begin{cases} P_n(z) = P_\lambda(z)P_\rho(z)Q_n(z), & (n \ge 2), \\ P_1(z) = 1, \end{cases} \tag{11}$$

where $Q_n(z)$ is the probability generating function of Y_n, the number of comparisons used by the two-way linear merge algorithm for merging two sorted files of sizes ρ and λ. From [10, Exercise 5.2.4.2] or [2], we have $Q_1(z) = 1$ and

$$Q_n(z) = E(z^{Y_n}) = \sum_{1\le k\le\lambda} \frac{\binom{n-k-1}{\rho-1} + \binom{n-k-1}{\lambda-1}}{\binom{n}{\lambda}} z^{n-k}, \qquad (n \ge 2).$$

Define $\mu_n := W_n - \alpha n$ and $\sigma_n := \sqrt{\beta n}$, where W_n is defined in (9), α and β are given as in Theorems 1 and 2. Let

$$F_n(x) = \mathrm{P}\left\{\frac{X_n - \mu_n}{\sigma_n} < x\right\} \quad \text{and}$$

$$\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-t^2/2}\,dt, \qquad (x \in \mathbb{R}).$$

THEOREM 3. *The distribution function of the random variable X_n is asymptotically normal,*

$$F_n(x) = \Phi(x) + O\left(\frac{\log n}{\sqrt{n}}\right), \qquad (n \to \infty),$$

uniformly in x.

To prove the asymptotic normality of X_n, we need an asymptotic estimate of $P_n(z)$ for z lying near the unity. We give a uniform treatment of this problem by taking logarithms on both sides of (11) and then applying again Lemma 4. This in turn introduces two problems: the location of the nonzero region of $P_n(z)$ and the boundedness of $\log Q_n(z) - (n-1)\log z$. The first problem is resolved by a similar argument used in [8] (Eneström–Kakeya theorem); the second is handled by a new inequality for characteristic functions which is also of some independent interest per se.

Once an asymptotic estimate of $P_n(e^{it})$ is available, we consider the characteristic function of the random variable $(X_n - \mu_n)/\sigma_n$ and then we apply the Berry–Esseen inequality (cf. [13]) which reduces the estimation of the discrepancy of two distributions to a certain average of the associated characteristic functions. But another problem arises when we apply the Berry–Esseen inequality: Lemma 4 is not precise enough for use because it stops at $O(\log n)$ term and this is in the exponent! We overcome this difficulty by arguments based on the analyticity of $P_n(z)$ and properties of analytic characteristic functions. Thus this part is no more elementary but analytic in nature.

We first state and prove three lemmas and then we proceed to the proof of the theorem. For the rest of the paper, δ represents always a small positive quantity whose value may vary from one occurrence to another (but independent of n and other asymptotic parameters).

4.1. *Lemmas*

The first lemma says that the probability distribution of Y_n is roughly majorized by a geometric distribution (the probabilities decrease geometrically).

LEMMA 5. *Write $Q_n(z) = \sum_{1 \le k \le \lambda} \pi_{n,k} z^{n-k}$ for $n \ge 2$, where*

$$\pi_{n,k} = \frac{\binom{n-k-1}{\rho-1} + \binom{n-k-1}{\lambda-1}}{\binom{n}{\lambda}}, \qquad (1 \le k \le \lambda).$$

Then for $1 \le k \le \lambda$,

$$\pi_{n,k} \le \tfrac{2}{3}\pi_{n,k-1}, \qquad (n \ge 2).$$

Proof. We have

$$\pi_{n,k} - \frac{2}{3}\pi_{n,k-1} = \frac{\binom{n-k-1}{\rho-1} + \binom{n-k-1}{\lambda-1}}{\binom{n}{\lambda}} - \frac{2}{3}\frac{\binom{n-k}{\rho-1} + \binom{n-k}{\lambda-1}}{\binom{n}{\lambda}}$$
$$= \frac{(n-k-1)!}{3\binom{n}{\lambda}}\left(\frac{n-k-3(\rho-1)}{(\rho-1)!(n-k-\rho+1)!} + \frac{n-k-3(\lambda-1)}{(\lambda-1)!(n-k-\lambda+1)!}\right)$$
$$\leq 0.$$

for $k \geq 3$. But for $k = 2$,

$$\pi_{n,2} - \frac{2}{3}\pi_{n,1} = \frac{(n-3)!\lambda!(n-\lambda)!}{3\cdot n!(\lambda-1)!(\rho-1)!}(-2n+2) \leq 0, \qquad (n \geq 2).$$

∎

The nonzero region of $Q_n(z)$ can now be located.

LEMMA 6. *For $n \geq 1$ the polynomials $Q_n(z)$ have no zero with modulus larger than $\frac{2}{3}$.*

Proof. Apply Lemma 5 and the Eneström–Kakeya theorem (cf. [11]): Let $\Pi(x) = \sum_{0 \leq j \leq m} a_j x^j$, where $m > 1$ and $a_j > 0$ for $j = 0, \ldots, m$. Then the moduli of the zeros of $\Pi(x)$ all lie between (including) $\min_{1 \leq j \leq m} a_{j-1}/a_j$ and $\max_{1 \leq j \leq m} a_{j-1}/a_j$. ∎

LEMMA 7. *Let $\{a_k\}_{k=0}^{n}$ be a sequence of nonnegative real numbers such that $a_0 > 0$. Assume that $a_k \leq \varepsilon a_{k-1}$ for $k = 1, \ldots, n$, where $0 < \varepsilon < 1$. Then for any complex $|z| \leq 1$,*

$$\left|\sum_{0 \leq k \leq n} a_k z^k\right| \geq \frac{1-\varepsilon}{1+\varepsilon}\sum_{0 \leq k \leq n} a_k.$$

Proof. Let $c_k = a_k/\varepsilon^k - a_{k+1}/\varepsilon^{k+1}$ for $k = 0, \dots, n-1$ and $c_n = a_n/\varepsilon^n$. Then $c_k \geq 0$. Now

$$\begin{aligned}\left|\sum_{0\le k\le n} a_k z^k\right| &= \left|\sum_{0\le k\le n} \varepsilon^k z^k \sum_{k\le j\le n} c_j\right| \\ &= \left|\sum_{0\le j\le n} c_j \sum_{0\le k\le j} \varepsilon^k z^k\right| \\ &\geq \frac{1}{1+\varepsilon}\left|\sum_{0\le j\le n} c_j\left(1-\varepsilon^{j+1}|z|^{j+1}\right)\right| \\ &\geq \frac{1}{1+\varepsilon}\sum_{0\le j\le n} c_j\left(1-\varepsilon^{j+1}\right) \\ &= \frac{1-\varepsilon}{1+\varepsilon}\sum_{0\le k\le n} a_k.\end{aligned}$$

∎

COROLLARY 1. *Let $\{a_k\}_{k=0}^n$ be a probability distribution satisfying $a_k \leq \varepsilon a_{k-1}$ for $1 \leq k \leq n$, where $0 < \varepsilon < 1$. Then the associated characteristic function satisfies*

$$\left|\sum_{0\le k\le n} a_k e^{ikt}\right| \geq \frac{1-\varepsilon}{1+\varepsilon},$$

for all real t.

4.2. *Proof of Theorem* 3

Taking logarithms on both sides of (11), we obtain

$$p_n(z) := \log P_n(z) = p_\lambda(z) + p_\rho(z) + q_n(z) + (n-1)\log z,$$

for $|z| \geq \frac{2}{3} + \delta$, where $q_n(z) := \log Q_n(z) - (n-1)\log z$. [The nonzero property of $P_n(z)$ follows from that of $Q_n(z)$ and (11).]

From Lemmas 5 and 7 with $a_k = \pi_{n,k}|z|^{-k}$, we deduce that $q_n(z) = O(1)$ uniformly for $\frac{2}{3} + \delta \leq |z| \leq \frac{3}{2} - \delta$. By linearity of recurrence and Lemma 4, we have

$$p_n(z) = W_n \log z + h(z)n + R_n(z), \qquad \left(\tfrac{2}{3} + \delta \leq |z| \leq \tfrac{3}{2} - \delta\right),$$

where $h(z) = \sum_{k \geq 1} q_{2^k}(z)/2^k$ and $R_n(z) = O(\log n)$, uniformly in z. Observe that $h(1) = 0$, $R_n(1) = 0$ and that

$$h'(1) = -\alpha, \qquad h''(1) + h'(1) = \beta.$$

Let $\varphi_n(t) = e^{-\mu_n it/\sigma_n} P_n(e^{it/\sigma_n}) = E(e^{it(X_n - \mu_n)/\sigma_n})$. By Taylor expansion, we have, for $|t| \leq \delta\sigma_n$,

$$\begin{aligned}\log \varphi_n(t) &= -(W_n - \alpha n)\frac{it}{\sigma_n} + W_n \frac{it}{\sigma_n} \\ &\quad + \left(h'(1)\frac{it}{\sigma_n} - \frac{h'(1) + h''(1)}{2\sigma_n^2} t^2 + O\left(|t|^3 \sigma_n^{-3}\right)\right) n \\ &\quad + R_n(e^{it/\sigma_n}).\end{aligned}$$

Because $R_n(1) = 0$, we have, by analyticity and Cauchy's integral formula,

$$\begin{aligned}R_n(e^{i\theta}) &= R_n(e^{i\theta}) - R_n(1) \\ &= \frac{e^{i\theta} - 1}{2\pi i} \int_{|w-1| = 2\delta} \frac{R_n(w)}{(w - e^{i\theta})(w - 1)}\, dw \\ &= O(|\theta| \log n),\end{aligned}$$

for $|\theta| \leq \delta$. It follows that

$$R_n(e^{it/\sigma_n}) = O\left(\frac{\log n}{\sigma_n}|t|\right), \qquad (|t| \leq \delta\sigma_n).$$

Thus,

$$\varphi_n(t) = \exp\left(-\frac{t^2}{2} + O\left(\frac{|t|^3}{\sigma_n} + \frac{\log n}{\sigma_n}|t|\right)\right), \qquad (|t| \leq \delta\sigma_n). \tag{12}$$

We now show that $|\varphi_n(t) - e^{-t^2/2}|/|t|$ is uniformly small by dividing the range of t into two parts.

(i) When $|t| < \sigma_n^{1/4}$, we have $|t|^3/\sigma_n = o(1)$ and $(\log n)|t|/\sigma_n = o(1)$. Thus,

$$\begin{aligned}|\varphi_n(t) - e^{-t^2/2}| &= e^{-t^2/2} |e^{O(|t|^3/\sigma_n + (\log n/\sigma_n)|t|)} - 1| \\ &= O\left(e^{-t^2/4}\left(\frac{|t|^3}{\sigma_n} + \frac{\log n}{\sigma_n}|t|\right)\right).\end{aligned}$$

(ii) When $\sigma_n^{1/4} \le |t| \le \delta\sigma_n$, we rewrite (12) as $\varphi_n(t) = e^{-t^2/2+E_n(t)}$, where $(K > 0)$,

$$|E_n(t)| \le K\frac{|t|^3}{\sigma_n} + K\frac{\log n}{\sigma_n}|t| \le \frac{t^2}{4}, \qquad \left(|t| \le \frac{\sigma_n}{4K}\right).$$

[So it suffices to take $\delta = 1/(4K)$.] Note that $|\varphi_n(t)| \le 1$, implying that $|E_n(t)| \le t^2/2$. Now using $|e^z - 1| \le |z|e^{|z|}$ for complex z, we obtain

$$\begin{aligned}|\varphi_n(t) - e^{-t^2/2}| &= e^{-t^2/2}|e^{E_n(t)} - 1| \\ &= O\left(e^{-t^2/4}\left(\frac{|t|^3}{\sigma_n} + \frac{\log n}{\sigma_n}|t|\right)\right),\end{aligned}$$

for $|t| \le \delta\sigma_n$.

By the Berry–Esseen smoothing inequality (cf. [13]), we have

$$\sup_x |F_n(x) - \Phi(x)| = O\left(\int_{-T}^{T}\left|\frac{\varphi_n(t) - e^{-t^2/2}}{t}\right| dt + \frac{1}{T}\right).$$

Taking $T = \delta\sigma_n$, we obtain

$$\begin{aligned}\int_{-T}^{T}\left|\frac{\varphi_n(t) - e^{-t^2/2}}{t}\right| dt &= O\left(\int_{-\infty}^{\infty} e^{-t^2/4}\left(\frac{t^2}{\sigma_n} + \frac{\log n}{\sigma_n}\right) dt\right) \\ &= O\left(\frac{\log n}{\sqrt{n}}\right),\end{aligned}$$

from which Theorem 3 follows. ∎

Figure 3 shows that the cost of QM satisfies actually a local limit theorem. This can be proved along a similar line of arguments as in [8] and the proof techniques of Theorem 3; details are omitted here. Note that the cost of BUM is not asymptotically normal as is easily seen by the case $n = 2^k + 1$; cf. (18).

5. AN INVARIANCE PRINCIPLE FOR POWER-OF-2 RECURRENCES

We discuss the generality of Lemmas 3 and 4 in this section; the results are needed when studying optimal variance of the cost of mergesorts. We show that the results of Lemmas 3 and 4 are invariant for the more

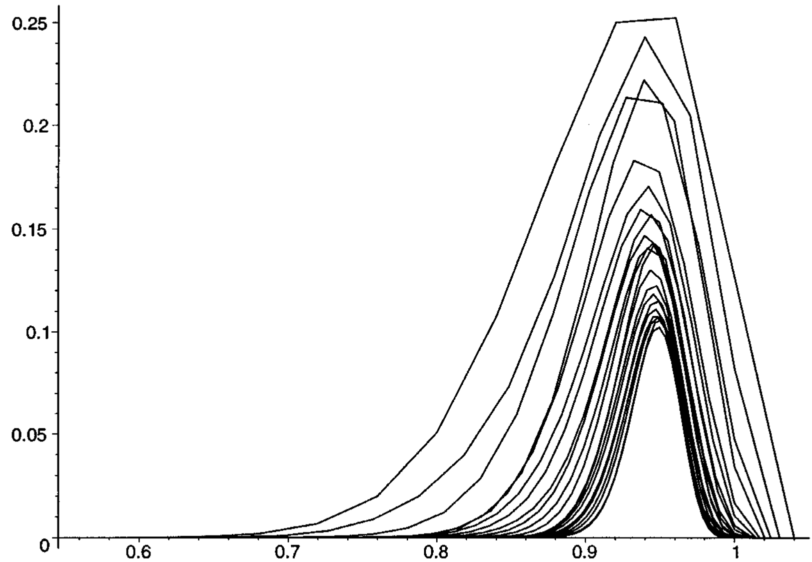

FIG. 3. The cost distribution of QM for $n = 10, 12, \ldots, 50$.

general recurrence $f_1 = g_1$ and

$$f_n = f_{2^{\lfloor \log_2 \theta n \rfloor}} + f_{n - 2^{\lfloor \log_2 \theta n \rfloor}} + g_n, \qquad (n \geq 2), \tag{13}$$

provided that $\frac{1}{2} \leq \theta < 1$ is a fixed constant. Thus $\theta = \frac{2}{3}$ is not a "magic number."

THEOREM 4 (Invariance Principle). *Assume that f_n satisfies the recurrence* (13) *with* $\frac{1}{2} \leq \theta < 1$. *Then* $f_n/n \to l < \infty$ *iff* $g_n = o(n)$ *and* $\sum_{j \geq 0} g_{2^j}/2^j = l$.

The same result subsists for the corresponding recurrence with the floor function replaced by the ceiling function: $f_1 = g_1$ and

$$f_n = f_{2^{\lceil \log_2 \theta n \rceil}} + f_{n - 2^{\lceil \log_2 \theta n \rceil}} + g_n, \qquad (n \geq 2), \tag{14}$$

where $\frac{1}{4} < \theta \leq \frac{1}{2}$ is a fixed constant. [This generalizes the max power-of-2 rule.] Because the proof of this result is similar, we omit the details.

Also the same algorithmic interpretation as that given in the final paragraph of Section 2 applies.

For the proof of Theorem 4, it suffices (cf. the proof of Lemma 3) to derive the following exact formula generalizing Lemma 2.

PROPOSITION 1. *The solution* f_n *to the recurrence* (13) *satisfies*

$$f_n = \sum_{0 \le j \le L} \left(\left(\left\lfloor \frac{n}{2^j} \right\rfloor - 1 \right) g_{2^j} + g_{n_j} \right)$$
$$+ \sum_{0 \le j \le L-1} \left(\zeta_j g_{2^{j+1}} + (b_j - \zeta_j) g_{n_j + b_j 2^j} \right), \qquad (n \ge 1), \quad (15)$$

where

$$\zeta_j = \left\lfloor \theta \left(1 + \left\{ \frac{n}{2^{j+1}} \right\} \right) \right\rfloor, \qquad (j = 0, 1, \dots, L-1).$$

Proof. Observe first that

$$\zeta_{L-1} = \begin{cases} 0, & \text{if } 2^{L-1} \le \theta n < 2^L; \\ 1, & \text{if } 2^L \le \theta n < 2^{L+1}; \end{cases}$$

thus $\lfloor \log_2 \theta n \rfloor = 2^{L-1}$ if $\zeta_{L-1} = 0$ and $\lfloor \log_2 \theta n \rfloor = 2^L$ if $\zeta_{L-1} = 1$. Likewise, ζ_j is either 0 or 1 according as $\lfloor \log_2 \theta n_{j+1} \rfloor = 2^j$ or $\lfloor \log_2 \theta n_{j+1} \rfloor = 2^{j+1}$, respectively. Note that

$$f_{2^L} = 2 f_{2^{L-1}} + g_{2^L} = \sum_{0 \le j \le L} 2^{L-j} g_{2^j}.$$

We divide the discussions into four cases.

1. $(b_{L-1}, \zeta_{L-1}) = (0, 0)$. Then

$$f_n = f_{n_{L-1}} + f_{2^{L-1}} + g_n.$$

2. $(b_{L-1}, \zeta_{L-1}) = (0, 1)$. Then

$$\begin{aligned} f_n &= f_{2^L} + f_{n-2^L} + g_n \\ &= f_{2^{L-1}} + f_{2^{L-1}} + f_{n-2^L} + g_n + g_{2^L} \\ &= f_{n_{L-1}} + f_{2^{L-1}} + g_n + g_{2^L} - g_{n_{L-1}}, \end{aligned}$$

because $\zeta_{L-2} = 1$, for $\zeta_{L-1} = 1$ implies that $\theta n \ge 2^L$. Thus,

$$\begin{aligned} \theta^{-1} \le \frac{n}{2^L} &= 1 + \sum_{0 \le j \le L-2} b_j 2^{-L+j} \\ &\le 1 + \sum_{0 \le j \le L-2} b_j 2^{-L+j+1} = 1 + \left\{ \frac{n}{2^{L-1}} \right\}, \end{aligned}$$

which in turn yields $\theta n_{L-1} \geq 2^{L-1}$.

3. $(b_{L-1}, \zeta_{L-1}) = (1, 0)$. Using the inequalities,

$$2^{L-1} \leq \theta(n - 2^{L-1}) < 2^L,$$

we obtain

$$\begin{aligned} f_n &= f_{2^{L-1}} + f_{n-2^{L-1}} + g_n \\ &= f_{n_{L-1}} + f_{2^{L-1}} + f_{2^{L-1}} + g_n + g_{n-2^{L-1}}. \end{aligned}$$

4. $(b_{L-1}, \zeta_{L-1}) = (1, 1)$. Then

$$\begin{aligned} f_n &= f_{2^L} + f_{n-2^L} + g_n \\ &= f_{n_{L-1}} + f_{2^{L-1}} + f_{2^{L-1}} + g_n + g_{2^L}. \end{aligned}$$

The previous four equations for f_n can be encapsulated into one,

$$\begin{aligned} f_n = f_{n_L} = f_{n_{L-1}} &+ (1 + b_{L-1}) f_{2^{L-1}} + g_{n_L} + \zeta_{L-1} g_{2^L} \\ &+ (b_{L-1} - \zeta_{L-1}) g_{n_{L-1} + b_{L-1} 2^{L-1}}. \end{aligned}$$

Iterating and simplifying this equation using (4), we obtain (15). ∎

An interesting by-product of (15) is that when $\theta = \frac{2}{3}$,

$$b_j = \zeta_j = \left\lfloor \frac{2}{3}\left(1 + \left\{\frac{n}{2^{j+1}}\right\}\right)\right\rfloor,$$

for $j = 0, 1, \ldots, L$.

Lemma 4 bears the following general form.

Lemma 8. *If* $g_n = O(1)$ *then the solution to the recurrence* (13) *satisfies* $f_n = ln + O(\log n)$, *where* $l = \sum_{j \geq 0} g_{2^j}/2^j$.

This result has an important application to the variance of the cost of mergesort; see the next section.

The invariance principle fails (or takes different forms) if $\theta < \frac{1}{2}$. For example, if $\theta = \frac{1}{4}$, then (13) reduces for $n = 2^L$ to

$$f_{2^L} = f_{2^{L-1}} + f_{2^{L-2}} + g_{2^L} - g_{2^{L-1}} + g_{2^{L-1} + 2^{L-2}},$$

which implies that the asymptotic nature is similar (linear difference equation) but the technicalities are much more complicated. Roughly, the expression of the leading constant will be different. In particular, the discrepancy of the sizes of the two subproblems increases when θ becomes smaller. Consequently, the total cost is nondecreasing, rendering these divide-and-conquer variants less attractive from a practical point of view.

6. OPTIMAL MERGESORTS

In this section we consider optimal mergesorts in the average and variance cases. *Our model is restricted to those mergesorts merging two subfiles at each stage and using linear merge as the universal merging scheme*. Our main results state that TDM is average-case optimal (it uses the least number of comparisons in the average case) and that QM is asymptotically optimal (actually for $n \geq 28$ with obvious modifications for small n) as far as the variance is concerned. Also we show that QM is not the unique mergesort with this latter property: every mergesort whose underlying dividing rule satisfies $n \mapsto (2^{\lfloor \log_2 \theta n \rfloor}, n - 2^{\lfloor \log_2 \theta n \rfloor})$ with $\frac{1}{2} \leq \theta < 1$ enjoys the same property!

6.1. *Average Case*

Our problem here is to find the indices achieving the minimum of the recurrence: $U(1) = 0$ and

$$U(n) = \min_{1 \leq j \leq n-1} \{U(j) + U(n-j) + u(j, n-j)\}, \qquad (n \geq 2),$$

where $u(x, y)$ is defined in (7).

We show that $j = \lfloor n/2 \rfloor$ (TDM) is the optimal choice.

THEOREM 5. *The average-case optimal mergesort is to divide as evenly as possible at each recursive stage*.

Proof. By induction on n. The result is obvious for $n = 2, 3$. Assume $n \geq 4$. We show that for $1 \leq j < \lfloor n/2 \rfloor$,

$$\begin{aligned}\Delta := U(j) + U(n-j) + u(j, n-j) - U\left(\left\lfloor \frac{n}{2} \right\rfloor\right) - U\left(\left\lceil \frac{n}{2} \right\rceil\right) \\ -u\left(\left\lfloor \frac{n}{2} \right\rfloor, \left\lceil \frac{n}{2} \right\rceil\right) > 0.\end{aligned}$$

By induction, we have for $1 \leq j < \lfloor n/2 \rfloor$,

$$\begin{aligned}&U(j) + U(n-j) + u(j, n-j) \\ &\quad = U\left(\left\lfloor \frac{j}{2} \right\rfloor\right) + U\left(\left\lceil \frac{j}{2} \right\rceil\right) + u\left(\left\lfloor \frac{j}{2} \right\rfloor, \left\lceil \frac{j}{2} \right\rceil\right) + U\left(\left\lfloor \frac{n-j}{2} \right\rfloor\right) \\ &\qquad + U\left(\left\lceil \frac{n-j}{2} \right\rceil\right) + u\left(\left\lfloor \frac{n-j}{2} \right\rfloor, \left\lceil \frac{n-j}{2} \right\rceil\right) + u(j, n-j). \quad (16)\end{aligned}$$

1. If one of j or $n-j$ is even, then

$$\left\lfloor \frac{j}{2} \right\rfloor + \left\lfloor \frac{n-j}{2} \right\rfloor = \left\lfloor \frac{n}{2} \right\rfloor \quad \text{and} \quad \left\lceil \frac{j}{2} \right\rceil + \left\lceil \frac{n-j}{2} \right\rceil = \left\lceil \frac{n}{2} \right\rceil.$$

Thus, by (16), we have, for $1 \le j < \lfloor n/2 \rfloor$,

$$\begin{aligned}\Delta = u\left(\left\lfloor \frac{j}{2} \right\rfloor, \left\lceil \frac{j}{2} \right\rceil\right) + u\left(\left\lfloor \frac{n-j}{2} \right\rfloor, \left\lceil \frac{n-j}{2} \right\rceil\right) - u\left(\left\lfloor \frac{j}{2} \right\rfloor, \left\lfloor \frac{n-j}{2} \right\rfloor\right) \\ - u\left(\left\lceil \frac{j}{2} \right\rceil, \left\lceil \frac{n-j}{2} \right\rceil\right) + u(j, n-j) - u\left(\left\lfloor \frac{n}{2} \right\rfloor, \left\lceil \frac{n}{2} \right\rceil\right).\end{aligned}$$

By straightforward computations,

$$\begin{aligned}&u(x,y) + u(z,w) - u(x,z) - u(y,w) + u(x+y, z+w) \\ &\quad -u(x+z, y+w) \\ &\qquad = (z-y)(w-x) \\ &\qquad\quad \times \left\{ \frac{1}{(y+1)(z+1)} + \frac{1}{(x+1)(w+1)} \right. \\ &\qquad\qquad \left. - \frac{(x+y+z+w+2)(x+y+z+w+1)}{(x+y+1)(x+z+1)(y+w+1)(z+w+1)} \right\},\end{aligned}$$

and the terms in { } can be shown to contain only positive terms of x, y, z, w. Thus $\Delta > 0$.

2. If both j and $n-j$ are odd, then

$$\left\lfloor \frac{j}{2} \right\rfloor + \left\lceil \frac{n-j}{2} \right\rceil = \left\lfloor \frac{n}{2} \right\rfloor \quad \text{and} \quad \left\lceil \frac{j}{2} \right\rceil + \left\lfloor \frac{n+j}{2} \right\rfloor = \left\lceil \frac{n}{2} \right\rceil.$$

Thus,

$$\begin{aligned}\Delta = u\left(\left\lfloor \frac{j}{2} \right\rfloor, \left\lceil \frac{j}{2} \right\rceil\right) + u\left(\left\lfloor \frac{n-j}{2} \right\rfloor, \left\lceil \frac{n-j}{2} \right\rceil\right) - u\left(\left\lfloor \frac{j}{2} \right\rfloor, \left\lceil \frac{n-j}{2} \right\rceil\right) \\ - u\left(\left\lceil \frac{j}{2} \right\rceil, \left\lfloor \frac{n-j}{2} \right\rfloor\right) + u(j, n-j) - u\left(\left\lfloor \frac{n}{2} \right\rfloor, \left\lceil \frac{n}{2} \right\rceil\right).\end{aligned}$$

Proceeding in a similar way as in the previous text, we can show that $\Delta > 0$.

This completes the proof. ∎

6.2. *Variance*

The first problem here is to find the indices minimizing the recurrence: $V(1) = 0$ and

$$V(n) = \min_{1 \le j \le n-1} \{V(j) + V(n-j) + v(j, n-j)\}, \qquad (n \ge 2),$$

where $v(x, y)$ is defined in (8).

We show that $V(n) = \beta n + O(\log n)$, implying that the variance of QM ($j = \rho(n)$) is asymptotically minimal. Numerical data show that the minimum is attained at $j = \rho(n)$ for $n \ne 6, 10, 11, 12, 13, 21, \ldots, 27$.

THEOREM 6. *The variance of QM is asymptotically optimal in the sense that* $V(n) = V_n^{[q]} + O(\log n)$.

Proof. Obviously,

$$V(n) \le V_n^{[q]} = \beta n + O(\log n).$$

On the other hand,

$$V(n) \ge \min_{1 \le j < n} \{V(j) + V(n-j)\} + \min_{1 \le j < n} v(j, n-j).$$

Because $j \mapsto v(j, n-j)$ is convex and symmetric for fixed n, we have

$$\min_{1 \le j < n} v(j, n-j) = v\left(\left\lfloor \frac{n}{2} \right\rfloor, \left\lceil \frac{n}{2} \right\rceil\right) \ge \gamma(n) := \frac{2n^2(n-2)}{(n+2)^2(n+4)}, \qquad (n \ge 2).$$

But $\gamma(n)$ is increasing and strictly concave for $n \ge 2$. By the Hammersley and Grimmett result (cf. [7]), we deduce that the minimum of the recurrence $\phi(1) = 0$ and

$$\phi(n) = \min_{1 \le j < n} \{\phi(j) + \phi(n-j)\} + \gamma(n), \qquad (n \ge 2)$$

is attained at $j = \rho(n)$. Thus,

$$\phi(n) = \phi(\rho) + \phi(\lambda) + \gamma(n), \qquad (n \ge 2).$$

Now applying Lemma 4, we obtain

$$\phi(n) = \beta n + O(\log n),$$

This completes the proof. ∎

Now let $\mathscr{M}(\theta)$ denote the set of mergesorts whose costs are described by the recurrence (13), where g_n denotes the cost of (linearly) merging two sorted subfiles of sizes $2^{\lfloor \log_2 \theta n \rfloor}$ and $n - 2^{\lfloor \log_2 \theta n \rfloor}$.

THEOREM 7. *The variance of the cost of any mergesort belonging to* $\mathscr{M}(\theta)$ *is asymptotically optimal for any fixed* θ, $\frac{1}{2} \leq \theta \leq 1 - \delta$, $\delta > 0$.

Proof. Because

$$\frac{n}{4} \leq \frac{\theta}{2} n < 2^{\lfloor \log_2 \theta n \rfloor} \leq \theta n \leq (1 - \delta) n,$$

the theorem follows from the boundedness of g_n,

$$g_n = v\left(2^{\lfloor \log_2 \theta n \rfloor}, n - 2^{\lfloor \log_2 \theta n \rfloor}\right) = O_\delta(1),$$

Lemma 8 and the preceding theorem. ▮

Similarly, mergesorts whose costs are described by (14) with $\frac{1}{4} < \theta \leq \frac{1}{2} - \delta$ possess the same property. But the case $\theta = \frac{1}{2}$ has to be excluded as discussed earlier.

APPENDIX

For completeness and for convenience of the reader, we list the exact solutions of the divide-and-conquer recurrences based on half−half and max power-of-2 rules.

The solution to the recurrence $f_1 = g_1$ and

$$f_n = f_{\lfloor n/2 \rfloor} + f_{\lceil n/2 \rceil} + g_n, \qquad (n \geq 2)$$

is given by (cf. [8]),

$$f_n = \sum_{0 \leq j \leq L} 2^j \left[\left(1 - \left\{ \frac{n}{2^j} \right\} \right) g_{\lfloor n/2^j \rfloor} + \left\{ \frac{n}{2^j} \right\} g_{\lfloor n/2^j \rfloor + 1} \right], \qquad (n \geq 1). \quad (17)$$

The solution to the recurrence $f_1 = g_1$ and

$$f_n = f_{2^{\lceil \log_2 n/2 \rceil}} + f_{n - 2^{\lceil \log_2 n/2 \rceil}} + g_n, \qquad (n \geq 2)$$

satisfies (cf. [12] or the proof technique of Lemma 2),

$$f_n = \sum_{0 \leq j \leq L} \left(\left\lfloor \frac{n}{2^j} \right\rfloor g_{2^j} + b_j g_{n_j} \right) - g_{2^{1/2(n)}}, \qquad (n \geq 1), \qquad (18)$$

where $v_2(n)$ is the dyadic valuation of n, namely, the largest j for which 2^j divides n.

ACKNOWLEDGMENT

We are indebted to Professor Zhi-Dong Bai for helpful suggestions.

REFERENCES

1. C. J. K. Batty and D. G. Rogers, Some maximal solutions of the generalized subadditive inequality, *SIAM J. Algebraic Discrete Methods* **3** (1982), 369–378.
2. P. Flajolet and M. Golin, Mellin transforms and asymptotics: the mergesort recurrence, *Acta Inform.* **31** (1994), 673–696.
3. P. Flajolet, P. Grabner, P. Kirschenhofer, H. Prodinger, and R. F. Tichy, Mellin transforms and asymptotics: digital sums, *Theoret. Comput. Sci.* **123** (1994), 291–314.
4. C. R. Glassey and R. M. Karp, On the optimality of Huffman trees, *SIAM J. Appl. Math.* **31** (1976), 368–378.
5. M. Golin and R. Sedgewick, Queue-mergesort, *Inform. Process. Lett.* **48** (1993), 253–259.
6. D. H. Greene and D. E. Knuth, ''Mathematics for the Analysis of Algorithms,'' Birkhäuser, Boston, 1981.
7. J. M. Hammersley and G. R. Grimmett, Maximal solutions of the generalized subadditive inequality, *in* ''Chapter 4 of Stochastic Geometry'' (E. F. Harding and D. G. Kendall, Eds.), Wiley, New York, 1974.
8. H.-K. Hwang, Limit theorems for mergesort, *Random Structures Algorithms* **8** (1996), 319–336.
9. H.-K. Hwang and J.-M. Steyaert, On the number of heaps and the cost of heap construction, *Theoret. Comput. Sci.*, to appear.
10. D. E. Knuth, ''The Art of Computer Programming, Volume III–Sorting and Searching,'' 2nd ed., Addison-Wesley, Reading, MA, 1998.
11. M. Marden, ''Geometry of Polynomials,'' 2nd ed., Mathematical Surveys, No. 3, Am. Math. Soc. Providence, RI, 1966.
12. W. Panny and H. Prodinger, Bottom-up mergesort—a detailed analysis, *Algorithmica* **14** (1995), 340–354.
13. V. V. Petrov, ''Limit Theorems of Probability Theory. Sequences of Independent Random Variables,'' Clarendon, Oxford, 1995.
14. R. Sedgewick, ''Algorithms in C,'' 3rd ed., Addison-Wesley, Reading, MA, 1998.
15. M. Snir, Exact balancing is not always good, *Inform. Process. Lett.* **22** (1986), 97–102.
16. T. Walsh, How evenly should one divide to conquer quickly? *Inform. Process. Lett.* **19** (1984), 203–208.