SIAM J. COMPUT.
Vol. 7, No. 3, August 1978

0097-5397/78/0703-0004 $01.00/0

EFFICIENT CALCULATION OF EXPECTED MISS RATIOS IN THE INDEPENDENT REFERENCE MODEL*

RONALD FAGIN† AND THOMAS G. PRICE†

Abstract. In the independent reference model of program behavior, King's formulas for the expected FIFO ("first-in-first-out") and expected LRU ("least-recently-used") miss ratios each contain an exponential number of terms (very roughly n^{CAP}, where n is the number of pages and CAP is the capacity of main memory). Hence, under the straightforward algorithms, these formulas are computationally intractable. We present an algorithm which is both efficient (there are $O(n \cdot \mathrm{CAP})$ additions, multiplications, and divisions) and provably numerically stable, for calculating the expected FIFO miss ratio. In the case of LRU, we present an efficient method, based on an urn model, for obtaining an unbiased estimate of the expected LRU miss ratio (the method requires $O(n \cdot \mathrm{CAP})$ additions and comparisons, and $O(\mathrm{CAP})$ divisions and random number generations).

Key words. independent reference model, miss ratio, FIFO, first-in-first-out, LRU, least-recently-used, storage management, page replacement, numerically stable

1. Introduction. The *independent reference model* (IRM) is a simple, widely studied model of page reference behavior in a paged computer system (Aho, Denning and Ullman [1]; Aven, Boguslavskii and Kogan [2]; Fagin [7], [8]; Fagin and Easton [9]; Franaszek and Wagner [10]; Gelenbe [11]; King [13]; Yue and Wong [24]). In this model, at each point in discrete time, exactly one page is referenced, where page i is referenced with probability p_i, independent of past history. We present an efficient, numerically stable algorithm for obtaining the expected FIFO ("first-in-first-out") miss ratio, and an efficient algorithm, based on an urn model, for obtaining an unbiased estimate of the expected LRU ("least-recently-used") miss ratio.

It is known that actual program page reference and data base segment or page reference patterns in a paging environment are quite intricate (Lewis and Shedler [15]; Lewis and Yue [16]; Madison and Batson [17]; Spirn and Denning [23]; Rodriguez-Rosell [21]). In particular, sequences of page references may be nonstationary. The assumption of independent references is not only intuitively suspect, but inconsistent with observed reference patterns. Why, therefore, should the IRM be investigated?

In the study of computer system performance, it is sometimes helpful to experiment with overly simple models, in order to gain insight into system behavior. In particular, since paging is a complex phenomenon, it is useful to study the effects of paging in conjunction with simple models of page reference patterns. From a mathematical point of view, the IRM is the simplest model in which pages retain their identity (as opposed, for example, to the independent LRU stack model of Oden and Shedler [19], and related models, in which all pages are treated identically). We remark that the formulation of other simple models that capture salient aspects of the referencing behavior of programs remains an important problem. The IRM is simple enough so as to be tractable, yet complex enough in the context of paging that there are nontrivial, surprising results. Sometimes these results generalize to realistic situations. Thus, in Fagin and Easton [9], it is shown that from the approximate independence of miss ratio on page size in the IRM, it follows that the miss ratio is approximately independent of page size in certain more realistic models, in which there is a

* Received by the editors November 1, 1976, and in final revised form October 25, 1977.
† IBM Research Laboratory, San Jose, California 95193.

"random" component and a "sequential" component. In fact, this independence of miss ratio on page size has been observed in an actual database system (see [9] for details).

Beyond easing the task of experimenting with the IRM, there is a further justification for obtaining algorithms that efficiently calculate miss ratios for the IRM. The IRM has been used as a component of more complex models that have accurately predicted miss ratio behavior. Thus, the miss ratios for Easton's model of data base references [6] and for Baskett and Rafii's model of program references [3] can be obtained directly from miss ratios in an IRM. The Easton and the Baskett–Rafii miss ratios, in turn, are supposedly fairly accurate predictors of genuine systems miss ratios.

Throughout, we assume that there are n pages, and that the probability that page i is referenced at time t is p_i $(i=1,\cdots,n)$, independent of past history. Of course, $\sum p_i = 1$. Denote the *capacity*, or size of first-level memory, by CAP $(1 \leqq \mathrm{CAP} \leqq n)$.

We deal with two page replacement algorithms, both of which are *demand* policies (Aho et al. [1]); that is, a page is brought into main (first-level) memory if and only if it is referenced but not present in main memory. The choice of which page is removed from main memory to make room for the newly-referenced but nonpresent page is determined by the page replacement algorithm. The first page replacement algorithm which we study in this paper is FIFO (Belady [4]), which replaces the page that has spent the longest time in memory. The second page replacement algorithm which we study is LRU (Mattson, Gecsei, Slutz, and Traiger [18]), which replaces the page that has been least recently referenced.

Define the *expected miss ratio* (in the independent reference model) to be the limit (as $t \to \infty$) of the probability that the page referenced at time t was not present in main memory at time t. King [13] showed that the expected FIFO and expected LRU miss ratios exist and are independent of the initial configuration of main memory. He showed that the expected FIFO miss ratio is

$$\frac{\sum p_{i_1} p_{i_2} \cdots p_{i_{\mathrm{CAP}}}(1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}}})}{\sum p_{i_1}\cdots p_{i_{\mathrm{CAP}}}}, \tag{1.1}$$

where the sums are each taken over all CAP-tuples $(i_1,\cdots,i_{\mathrm{CAP}})$ such that $i_j \neq i_k$ if $j \neq k$. Further, he showed that the expected LRU miss ratio is

$$\sum \frac{p_{i_1} p_{i_2}\cdots p_{i_{\mathrm{CAP}}}(1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}}})}{(1-p_{i_1})(1-p_{i_1}-p_{i_2})\cdots(1-p_{i_1}-p_{i_2}-\cdots-p_{i_{\mathrm{CAP}-1}})}, \tag{1.2}$$

where again, the sum is taken over all CAP-tuples $(i_1,\cdots,i_{\mathrm{CAP}})$ such that $i_j \neq i_k$ if $j \neq k$. We note that Gelenbe [11] showed that under the RAND ("random") page replacement algorithm (Belady [4]), in which the page to be removed from main memory in the event of a page fault is selected randomly, the expected miss ratio is the same as that of FIFO, that is, formula (1.1).

Each of the sums appearing in (1.1) and (1.2) contain very roughly n^{CAP} terms (actually $n(n-1)\cdots(n-\mathrm{CAP}+1)$ terms). Hence, for moderate values of n and CAP, formulas (1.1) and (1.2) cannot be evaluated numerically under the straightforward algorithm. For example, if $n=100$ and $\mathrm{CAP}=30$, then each of the sums contain over 10^{57} terms. The purpose of this paper is to provide fast, stable methods for evaluating the expected FIFO miss ratio (1.1) and for approximating the expected LRU miss ratio (1.2).

2. An efficient algorithm for the expected FIFO miss ratio. In this section we present an efficient, provably stable algorithm for evaluating King's formula

$$\frac{\sum p_{i_1}p_{i_2}\cdots p_{i_{\mathrm{CAP}}}(1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}}})}{\sum p_{i_1}p_{i_2}\cdots p_{i_{\mathrm{CAP}}}} \tag{2.1}$$

for the expected FIFO miss ratio in the independent reference model. (The sums are taken over all CAP-tuples $(i_1, \cdots, i_{\mathrm{CAP}})$ such that $i_j \neq i_k$ if $j \neq k$.)

Let $p_1, \cdots, p_n$ be a fixed but arbitrary ordering of the n page probabilities. For each positive integer $m \leqq n$ and each positive integer r, define

$$E(r, m) = \sum p_{i_1} \cdots p_{i_r},$$

where the sum is taken over all r-element *subsets* $\{i_1, \cdots, i_r\}$ of $\{1, \cdots, m\}$.

In other words, for each r-element subset of the first m probabilities $p_1, \cdots, p_m$, there is a term of $E(r, m)$ which is the product of these r probabilities. We make the usual convention that an empty sum is 0; hence $E(r, m) = 0$ if $r > m$.

We now express the expected FIFO miss ratio (2.1) as a function of the terms $E(r, m)$. Note that the numerator of formula (2.1) can be rewritten as

$$\sum p_{i_1}p_{i_2}\cdots p_{i_{\mathrm{CAP}}} \sum_{i \neq i_1, \cdots, i_{\mathrm{CAP}}} p_i,$$

which equals

$$\sum p_{i_1}p_{i_2}\cdots p_{i_{\mathrm{CAP}+1}}, \tag{2.2}$$

where the sum in (2.2) is taken over all (CAP+1)-tuples $(i_1, \cdots, i_{\mathrm{CAP}+1})$ such that $i_j \neq i_k$ if $j \neq k$. But (2.2) is simply $(\mathrm{CAP}+1)!E(\mathrm{CAP}+1, n)$, since $E(\mathrm{CAP}+1, n)$ is a sum over sets while (2.2) is the corresponding sum over tuples. Likewise, the denominator of (2.1) equals $\mathrm{CAP}!E(\mathrm{CAP}, n)$. Since we just showed that the numerator of (2.1) equals $(\mathrm{CAP}+1)!E(\mathrm{CAP}+1, n)$ and the denominator is $\mathrm{CAP}!E(\mathrm{CAP}, n)$, it follows that the expected FIFO miss ratio (2.1) equals

$$\frac{(\mathrm{CAP}+1)E(\mathrm{CAP}+1, n)}{E(\mathrm{CAP}, n)}. \tag{2.3}$$

We now show how to obtain an efficient, numerically stable algorithm for computing (2.3). Along the way, we also derive an efficient but *unstable* algorithm for computing (2.3).

We first verify the following recurrence equation for $E(r, m)$ when $r > 1$ and $m > 1$:

$$E(r, m) = E(r, m-1) + p_m E(r-1, m-1). \tag{2.4}$$

The first term $E(r, m-1)$ of (2.4) is the sum of those terms in $E(r, m)$ which do not have p_m as a factor, and the second term $p_m E(r-1, m-1)$ is the sum of the terms which do have p_m as a factor.

Recurrence equation (2.4) can be used recursively to compute the matrix of values $E(r, m)$ with $1 \leqq r \leqq \mathrm{CAP}+1$ and $1 \leqq m \leqq n$. Using this approach, we can calculate (2.3), the expected FIFO miss ratio, with approximately $2n \cdot \mathrm{CAP}$ additions and multiplications. Unfortunately, this method suffers from numerical instability, because for interesting values of n and CAP, the entries of the E matrix vary by enough orders of magnitude that they exceed the range of typical floating-point hardware (and so underflow occurs). For example, assume that all page reference probabilities are equal. Then $E(r, m) = \binom{m}{r}(1/n)^r$. So if $n = 1000$, then $E(1, 1000) = 1$ and $E(1000, 1000) = 10^{-3000}$. The reason why underflow causes large errors for us is that if we add two terms x and y which are nearly equal (such as $x = E(r, m-1)$ and $y = p_m E(r-1, m-1)$ on the right-hand side of (2.4)), and if the y term has underflowed to zero but the x term has not, then a large relative error is introduced and then propagated. In the example given, the actual value of $x + y$ would be almost twice the calculated value.

We now show how to calculate (2.3) both efficiently and stably. In order to calculate (2.3), we need only calculate the ratio $E(\mathrm{CAP}+1, n)/E(\mathrm{CAP}, n)$ (but not $E(\mathrm{CAP}+1, n)$ and $E(\mathrm{CAP}, n)$ separately). Let

$$F(r, m) = E(r, m)/E(r-1, m)$$

for $1 \leqq r \leqq \mathrm{CAP}+1$ and $1 \leqq m \leqq n$. Under the usual convention that empty products are 1, we define $E(0, m)$ to be 1 for each m; then $F(1, m) = E(1, m) = p_1 + \cdots + p_m$. Formula (2.3), and hence the expected FIFO miss ratio, equals $(\mathrm{CAP}+1)F(\mathrm{CAP}+1, n)$. We can derive a recurrence equation for F directly and avoid our earlier numerical difficulties. We first note that if $r > 1$ and $m > 1$, then

$$\begin{aligned} F(r, m) &= \frac{E(r, m)}{E(r-1, m)} \\ &= \frac{E(r, m)/E(r-1, m-1)}{E(r-1, m)/E(r-2, m-1)} E(r-1, m-1)/E(r-2, m-1). \end{aligned} \tag{2.5}$$

If we divide both sides of (2.4) by $E(r-1, m-1)$, then we obtain

$$\begin{aligned} E(r, m)/E(r-1, m-1) &= (E(r, m-1)/E(r-1, m-1)) + p_m \\ &= F(r, m-1) + p_m. \end{aligned} \tag{2.6}$$

If we use (2.6) to replace $E(r, m)/E(r-1, m-1)$ in (2.5) by $F(r, m-1) + p_m$, and if similarly, we replace $E(r-1, m)/E(r-2, m-1)$ in (2.5) by $F(r-1, m-1) + p_m$, then we obtain the recurrence equation

$$F(r, m) = \frac{F(r, m-1) + p_m}{F(r-1, m-1) + p_m} F(r-1, m-1), \tag{2.7}$$

which holds for $1 \leqq r \leqq \mathrm{CAP}+1$ and $1 \leqq m \leqq n$. This recurrence equation can be used recursively to compute the matrix of values $F(r, m)$ starting from the boundary conditions

$$\begin{aligned} F(r, 1) &= \begin{cases} p_1, & r = 1, \\ 0, & 2 \leq r \leq \mathrm{CAP}+1, \end{cases} \\ F(1, m) &= p_1 + \cdots + p_m, \quad 1 \leqq m \leqq n. \end{aligned} \tag{2.8}$$

One way to calculate F is to initialize the first column and row using the equations for $F(r, 1)$ and $F(1, m)$ in (2.8) and then to calculate the entries for $2 \leqq r \leqq \mathrm{CAP}+1$ and $2 \leqq m \leqq n$, column by column, by using equation (2.7). Then the expected FIFO miss ratio with capacity CAP is $(\mathrm{CAP}+1)F(\mathrm{CAP}+1, n)$.

This algorithm requires approximately $4n \cdot \mathrm{CAP}$ additions, multiplications, and divisions. We note that this algorithm has the interesting property that in calculating the expected FIFO miss ratio with capacity CAP, we automatically calculate the expected FIFO miss ratios with capacities $1, \cdots, \mathrm{CAP}-1$.

We sketch a proof of the numerical stability of this FIFO algorithm in the Appendix.

3. An unbiased estimate of the expected LRU miss ratio. In this section we present an efficient method for obtaining an unbiased estimate of the expected LRU miss ratio, which, we recall, is given by

$$\sum \frac{p_{i_1} p_{i_2} \cdots p_{i_{\mathrm{CAP}}}(1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}}})}{(1-p_{i_1})(1-p_{i_1}-p_{i_2})\cdots(1-p_{i_1}-p_{i_2}-\cdots-p_{i_{\mathrm{CAP}-1}})}, \tag{3.1}$$

in the independent reference model.

Consider the following experiment, which involves drawing balls from an urn without replacement. Assume that an urn contains n balls numbered $1, \cdots, n$ (which correspond to our n pages). We say that ball i has *weight* p_i $(i = 1, \cdots, n)$, where $\{p_1, \cdots, p_n\}$ is the page probability distribution. Select one ball from the urn, in such a way that a given ball is selected with probability equal to its weight. Thus, ball i is selected with probability p_i $(i = 1, \cdots, n)$. Assume that ball i_1 was selected. Now renormalize the weights of the remaining $(n-1)$ balls so that the sum of their weights is 1. Thus, the weight of ball j is now $p_j/(1-p_{i_1})$, for $j \neq i_1$. Select a second ball from the urn, where once again a given ball is selected with probability equal to its new weight. Assume that ball i_2 was selected. Now renormalize the weights of the remaining $(n-2)$ balls so that the sum of their weights is 1: thus, the weight of ball j is now $p_j/(1-p_{i_1}-p_{i_2})$, for $j \neq i_1, i_2$. Continue the process until CAP balls have been selected. Let A (an estimate of (3.1)) be the value $1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}}}$. Note that with probability

$$p_{i_1} \frac{p_{i_2}}{1-p_{i_1}} \frac{p_{i_3}}{1-p_{i_1}-p_{i_2}} \cdots \frac{p_{i_{\mathrm{CAP}}}}{1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}-1}}}, \tag{3.2}$$

ball i_1 was selected first, ball i_2 was selected second, $\cdots$, and ball i_{CAP} was selected last; in this case A took on the value $1-p_{i_1}-\cdots-p_{i_{\mathrm{CAP}}}$. Therefore, the expected value of the random variable A is given by (3.1); that is, A is an unbiased estimate of (3.1).

The experiment we just described is faithfully mimicked by the algorithm in Figure 1 for obtaining a value for A (in the first line, P is our probability vector of page probabilities).

```
LET P1 = P;
LET A = 1;
DO I = 1 TO CAP;
    SELECT A RANDOM NUMBER R BETWEEN 0 AND 1;
    FIND THE FIRST J BETWEEN 1 AND N SUCH THAT
      P1(1)+···+P1(J)≧R;
    LET A = A − P(J);
    LET S = 1 − P1(J);
    DO K = 1 TO N;
     LET P1(K) = P1(K)/S;
    END;
    LET P1(J) = 0;
END;
```

FIG. 1

If there are, say, 100 independent replications of the experiment (that is, if the program is run 100 times, with different seeds to the pseudo-random number generator), then the average of the 100 values of A which are obtained also, of course, give an unbiased estimate of (3.1), and we can use the central limit theorem to obtain approximate confidence intervals for our estimate.

We now give a numerical example, using "Zipf's Law" (Zipf [25]; Knuth [14, vol. 3, p. 397]), in which the probability p_i of referencing the ith most frequently referenced page is

$$p_i = \frac{k}{i^\theta}, \qquad 1 \leqq i \leqq n,$$

where θ is a positive constant (the "skewness"), and k is a normalizing constant chosen so that $\sum p_i = 1$.

In our example, the skewness θ is 0.5, the number n of pages is 100, and the capacity CAP is 30. When we ran a version of the program in Fig. 1 100 times, we obtained 100 results $A_1, \cdots, A_{100}$. The average value

$$\bar{A} = (A_1 + \cdots + A_{100})/100$$

turned out to be 0.6119 (rounded to 4 decimal places). This value is our unbiased estimate of the expected LRU miss ratio (3.1). How much confidence should be placed in this estimate? To answer this question, we calculated several other statistical quantities of interest. Let

$$D = \left(\left(\sum_{i=1}^{100} (A_i - \bar{A})^2 \right) \Big/ 99 \right)^{1/2}.$$

In general, instead of 99, we would use $(L-1)$, where L is the number of independent runs of the program. Then D is an unbiased estimate of the standard deviation, and $X = D/\sqrt{L}$ of the standard deviation of the mean. In this case, D turned out to be 0.0301, and so X was 0.0030. Under the normal approximation, which is valid in large samples by the central limit theorem (here the sample size is 100), we know that an approximate 95% confidence interval for the sample mean is given by $\bar{A} \pm 2X$. (The normal approximation was justified in this case by the Kolmogorov–Smirnov test [14, vol. 2, p. 41]). So with approximately 95% confidence, we can say that the expected LRU miss ratio for this probability distribution (Zipf's Law, skewness 0.5, number of pages 100) with capacity CAP = 30 is 0.6119 ± 0.0060.

4. Examples. As a demonstration of the power of current techniques (including those developed in this paper) for obtaining expected miss ratios in the independent reference model, we present a family of examples (Table 1). In each case, we consider a Zipf's law probability distribution with skewness $\theta = 0.5$. We vary the number n of pages, and we also vary the capacity CAP in such a way that the "normalized capacity" CAP/n is 0.3. All values are rounded to four decimal places. We include not only the expected FIFO and LRU miss ratios, but also the expected WS, or working-set miss ratios (Denning and Schwartz [5]), the expected A_0 miss ratios (Aho et al. [1]), and the expected VMIN miss ratios (Prieve and Fabry [20]; Slutz [22]). Here A_0 is the optimal page replacement algorithm with no knowledge of the future in the independent reference model, and VMIN is the optimal variable-space page replacement algorithm under demand paging (with lookahead).

In the case of WS and VMIN, which are variable-space page-replacement algorithms, the quantity CAP is the *expected* number of pages in main memory. For example, in the WS case, the window size T is chosen in such a way that CAP is the expected working-set size.

The fact that the expected LRU, WS, A_0, and VMIN miss ratios have limiting values (as in Table 1) is proven in Fagin [8], where closed-form formulas are exhibited for these limits. Further, it is shown there that the limits in the LRU and WS cases are the same. (In the case of Table 1, the common limit is 0.5701.) It is an open problem as to whether there is a limiting value for the expected FIFO miss ratio, and how to find the limit.

We close this section with a minor technical comment on the LRU calculations in Table 1. Except for the $n = 10$ case, for which we used King's LRU formula, the given interval in the LRU column is approximately a 95% confidence interval. For the $n = 100$ and $n = 1000$ cases, the experiment described in § 3 was performed 100 times (that is, the L of § 3 is 100). For the $n = 10000$ case, the experiment was performed only 30 times, because of the great amount of paging which takes place when dealing with very large vectors.

TABLE 1
Expected miss ratios. (Zipf's Law, skewness $\theta = 0.5$, normalized capacity 0.3)

	FIFO	LRU	WS	A_0	VMIN
$n = 10$, CAP = 3	0.6660	0.6607	0.6599	0.5741	0.3601
$n = 100$, CAP = 30	0.6304	0.6119 ± 0.0060	0.6096	0.4870	0.2858
$n = 1{,}000$, CAP = 300	0.6091	0.5827 ± 0.0017	0.5831	0.4629	0.2706
$n = 10{,}000$, CAP = 3,000	0.6007	0.5748 ± 0.0010	0.5742	0.4556	0.2663
Limiting value	?	0.5701	0.5701	0.4523	0.2643

Appendix. The numerical stability of the FIFO algorithm. We sketch a proof that the FIFO algorithm described at the end of § 2 is numerically stable. We first show that there is not a large range in the matrix of values $F(r, m)$, where $1 \leqq r \leqq \mathrm{CAP}+1$ and $1 \leqq m \leqq n$; that is, we show that there are not many orders of magnitude between the smallest positive entry and the largest positive entry (note that there are no negative entries, although there are zero entries). In fact, we show that the largest entry is 1, and the smallest positive entry is at least $(\min p_i)/(\mathrm{CAP}+1)$. Therefore, there is no underflow in cases of interest. We then sketch a relative error analysis which shows that the maximum relative error in the entries $F(m, r)$ grows linearly with n (the number of pages).

Pick r_0 and m_0 so that $1 \leqq r_0 \leqq \mathrm{CAP}+1$ and $1 \leqq m_0 \leqq n$. If $r_0 > m_0$ then $F(r_0, m_0) = 0$. So assume that $r_0 \leqq m_0$. We now show that

$$(\min\{p_i: i = 1, \cdots, n\})/(\mathrm{CAP}+1) \leqq F(r_0, m_0) \leqq 1. \tag{A.1}$$

Let $S = p_1 + \cdots + p_{m_0}$, and define $q_i = p_i/S$ for $1 \leqq i \leqq m_0$; that is, we take the first m_0 probabilities $p_1, \cdots, p_{m_0}$, we normalize them so that their sum is 1, and we call the normalized probabilities $q_1, \cdots, q_{m_0}$. Define $E_1(r_0, m_0)$ and $E_1(r_0 - 1, m_0)$ in the same way as we defined $E(r_0, m_0)$ and $E(r_0 - 1, m_0)$ in § 2, except that we use the probabilities $q_1, \cdots, q_{m_0}$ instead of the probabilities $p_1, \cdots, p_n$. Hence, by analogy with (2.3) we know that the expected FIFO miss ratio, using the probability distribution $q_1, \cdots, q_{m_0}$, and with capacity $r_0 - 1$ is

$$\frac{r_0 E_1(r_0, m_0)}{E_1(r_0 - 1, m_0)}. \tag{A.2}$$

We now obtain upper and lower bounds for (A.2). Since (A.2) is the expected value of a miss ratio, it is bounded above by 1. Furthermore, (A.2) is at least as big as the expected value of the A_0 miss ratio (Aho et al. [1]) with the same capacity, where A_0 is the optimal page replacement algorithm with no knowledge of the future in the independent reference model. This expected A_0 miss ratio (with capacity r_0-1) is in turn at least as big as the expected A_0 miss ratio with capacity m_0-1 (since $(r_0-1)\leqq(m_0-1)$ by assumption). And, it is easy to check that the expected A_0 miss ratio with capacity m_0-1 is at least as big as $\min\{q_i: i=1,\cdots,m_0\}$. (Recall that these miss ratios are over the probability distribution $\{q_1,\cdots,q_{m_0}\}$.) To summarize the bounds we have obtained for (A.2), we have shown that

$$\min\{q_i: i=1,\cdots,m_0\}\leqq\frac{r_0E_1(r_0,m_0)}{E_1(r_0-1,m_0)}\leqq 1. \tag{A.3}$$

It is easy to see that $E(r_0,m_0)=S^{r_0}E_1(r_0,m_0)$, since $E(r_0,m_0)$ is a sum of products of r_0-element subsets of $\{p_1,\cdots,p_{m_0}\}$, while $E_1(r_0,m_0)$ is the corresponding sum of products of r_0-element subsets of $\{q_1,\cdots,q_{m_0}\}=\{p_1/S,\cdots,p_{m_0}/S\}$. Similarly, $E(r_0-1,m_0)=S^{r_0-1}E_1(r_0-1,m_0)$. Hence

$$\begin{aligned} F(r_0,m_0)&=\frac{E(r_0,m_0)}{E(r_0-1,m_0)}\\ &=\frac{S^{r_0}E_1(r_0,m_0)}{S^{r_0-1}E_1(r_0-1,m_0)}\\ &=\frac{SE_1(r_0,m_0)}{E_1(r_0-1,m_0)}. \end{aligned} \tag{A.4}$$

If, using (A.4), we substitute $F(r_0,m_0)/S$ for $E_1(r_0,m_0)/E_1(r_0-1,m_0)$ in (A.3), then after multiplying all parts of the resulting inequality by S/r_0, we obtain

$$(S/r_0)\min\{q_i: i=1,\cdots,m_0\}\leqq F(r_0,m_0)\leqq S/r_0. \tag{A.5}$$

Now $(S/r_0)\leqq 1$ since $S\leqq 1$ and $r_0\geqq 1$, and so it follows from (A.5) that $F(r_0,m_0)\leqq 1$, which establishes our upper bound on F. (In fact, this upper bound is attained, since $F(1,n)=1$.) As for the lower bound: we know that $\min\{q_i: i=1,\cdots,m_0\}=\min\{p_i: i=1,\cdots,m_0\}/S\geqq\min\{p_i: i=1,\cdots,n\}/S$. Furthermore, $r_0\leqq\mathrm{CAP}+1$, and so from (A.5) we obtain

$$\min\{p_i: i=1,\cdots,n\}/(\mathrm{CAP}+1)\leqq F(r_0,m_0), \tag{A.6}$$

which gives our lower bound on F. (This lower bound is actually attained when $p_i=1/n$ for each i, when $r_0=\mathrm{CAP}+1$, and when $m_0=n$.)

Having shown that there is no underflow (in cases of interest), we can now analyze the propagation of relative error. (If A is a quantity and A' is the calculated value of A, then the *relative error* is $(A'-A)/A$; note that the relative error can be positive, negative, or zero.) The key to stability for our algorithm is the fact that if the relative error in $F(r,m-1)$ is ε_1, and if the relative error in $F(r-1,m-1)$ is ε_2, then the relative error in

$$F(r,m)=\frac{F(r,m-1)+p_m}{F(r-1,m-1)+p_m}F(r-1,m-1)$$

is smaller in magnitude than the maximum of the magnitudes of ε_1 and ε_2. Why is this? For notational convenience, write A for $F(r,m-1)$, B for $F(r-1,m-1)$, and C

for p_m. We assume for now that C has no relative error. It is an important combinatorial fact that $F(r-1, m-1) \geqq F(r, m-1)$; this is Theorem 53 in Hardy, Littlewood and Polya [12, p. 52]. Therefore, $B \geqq A$. Let A' and B' be the calculated values of A and B respectively; thus, $A' = A(1+\varepsilon_1)$, and $B' = B(1+\varepsilon_2)$. Then the relative error in $F(r, m)$, that is, the relative error in $(A+C)B/(B+C)$, is given by

$$\left(\frac{(A'+C)B'}{(B'+C)} \Big/ \frac{(A+C)B}{(B+C)}\right) - 1. \tag{A.7}$$

Since (A.7) is the relative error in $F(r, m)$, our goal is to show that the absolute value of (A.7) is bounded above by max $(|\varepsilon_1|, |\varepsilon_2|)$.

If we replace A' by $A(1+\varepsilon_1)$ and B' by $B(1+\varepsilon_2)$ in (A.7), it is easily verified that the resulting expression equals

$$\left(\frac{\varepsilon_1(1+\varepsilon_2)(B+C)A}{(A+C)} + \varepsilon_2 C\right) \Big/ ((1+\varepsilon_2)B + C). \tag{A.8}$$

The absolute value of (A.8) is bounded above by

$$\left(\frac{|\varepsilon_1|(1+\varepsilon_2)(B+C)A}{(A+C)} + |\varepsilon_2| C\right) \Big/ ((1+\varepsilon_2)B + C), \tag{A.9}$$

where we have assumed that $|\varepsilon_2| < 1$ (so that $1+\varepsilon_2$ is positive). It follows immediately from $B \geqq A$ that $(B+C)A/(A+C) \leqq B$. Therefore, (A.9) is bounded above by the expression obtained by replacing $(B+C)A/(A+C)$ in (A.9) by B. That is, (A.9) is bounded above by

$$(c_1|\varepsilon_1| + c_2|\varepsilon_2|)/(c_1 + c_2), \tag{A.10}$$

where $c_1 = (1+\varepsilon_2)B$ and $c_2 = C$. But (A.10) is a weighted average of $|\varepsilon_1|$ and $|\varepsilon_2|$, and so is bounded above by their maximum. This is what we wanted to show.

If we now take into consideration the effect of roundoff and the fact that the constants p_m may also be in error due to the fact that they may not be represented exactly, then we find that the maximum relative error in the entries $F(r, m)$ is a small constant times the number n of columns of F, times ε, where ε is the inherent roundoff error in a floating-point number. (Thus, ε is 2^{-t}, where t is the number of bits in the representation of the mantissa of a floating-point number.)

Acknowledgment. The authors are grateful to Gerry Shedler for useful comments that led to improved exposition.

REFERENCES

[1] A. V. AHO, P. J. DENNING AND J. D. ULLMAN, *Principles of optimal page replacement*, J. Assoc. Comput. Mach., 18 (1971), pp. 80–93.

[2] O. I. AVEN, L. B. BOGUSLAVSKII AND YA. A. KOGAN, *Some results on distribution-free analysis of paging algorithms*, IEEE Trans. Computers, C-25 (1976), pp. 737–745.

[3] F. BASKETT AND A. RAFII, *The A_0 inversion model of program paging behavior*, Computer Science Rep. CS-76-579, Stanford University, Nov. 1976, Comm. ACM, to appear.

[4] L. A. BELADY, *A study of replacement algorithms for a virtual-storage computer*, IBM Systems J., 5 (1966), pp. 78–101.

[5] P. J. DENNING AND S. C. SCHWARTZ, *Properties of the working-set model*, Comm. ACM, 15 (1972), pp. 191–198.

[6] M. C. EASTON, *Model for interactive data base reference string*, IBM J. Res. and Devel., 19 (Nov. 1975), pp. 550–556.

[7] R. FAGIN, *A counterintuitive example of computer paging*, Comm. ACM, 19 (1976), pp. 96–97. (*Corrigendum*: Comm. ACM, 19 (1976), p. 187.)

[8] ———, *Asymptotic miss ratios over independent references*, J. Comput. System Sci., 14 (1977), pp. 222–250.

[9] R. FAGIN AND M. C. EASTON, *The independence of miss ratio on page size*, J. Assoc. Comput. Mach., 23 (1976), pp. 128–146.

[10] P. A. FRANASZEK AND T. J. WAGNER, *Some distribution-free aspects of paging algorithm performance*, Ibid., 21 (1974), pp. 31–39.

[11] E. GELENBE, *A unified approach to the evaluation of a class of replacement algorithms*, IEEE Trans. Computers, C-22 (1973), pp. 611–618.

[12] G. H. HARDY, J. E. LITTLEWOOD AND G. PÓLYA, *Inequalities*, Cambridge University Press, London, 1964.

[13] W. F. KING, III, *Analysis of paging algorithms*, IFIP Conf. Proc. Ljubljana, Yugoslavia (Aug. 1971).

[14] D. E. KNUTH, *The Art of Computer Programming*, vol. 2, 1969, and vol. 3, 1973, Addison-Wesley, Reading, MA.

[15] P. A. W. LEWIS AND G. S. SHEDLER, *Empirically derived micromodels for sequences of page exceptions*, IBM J. Res. Devel., 17 (1973), pp. 86–100.

[16] P. A. LEWIS AND P. C. YUE, *Statistical analysis of program reference patterns in a paging environment*, Proc. IEEE Conf. Computers, Boston, 1971.

[17] A. W. MADISON AND A. P. BATSON, *Characteristics of program localities*, Comm. ACM, 19 (1976), pp. 285–294.

[18] R. MATTSON, J. GECSEI, D. SLUTZ AND I. TRAIGER, *Evaluation techniques for storage hierarchies*, IBM Systems J., 9 (1970), pp. 78–117.

[19] P. H. ODEN AND G. S. SHEDLER, *A model of memory contention in a paging machine*, Comm. ACM, 15 (1972), pp. 761–771.

[20] B. G. PRIEVE AND R. S. FABRY, *VMIN—an optimal variable-space page replacement algorithm*, Ibid., 19 (1976), pp. 295–297.

[21] J. RODRIGUEZ-ROSELL, *Empirical data reference behavior in data base systems*, IEEE Computer, 9 (1976), no. 11, pp. 9–13.

[22] D. R. SLUTZ, *A relation between working set and optimal algorithms for segment reference strings*, IBM Research Rep. RJ 1623, July 1975, San Jose, California.

[23] J. R. SPIRN AND P. J. DENNING, *Experiments with program locality*, AFIPS Conf. Proc. 41, (1972), pp. 611–621.

[24] P. C. YUE AND C. K. WONG, *On the optimality of the probability ranking scheme in storage applications*, J. Assoc. Comput. Mach., 20 (1973), pp. 624–633.

[25] G. K. ZIPF, *Human Behavior and the Principle of Least Effort*, Addison-Wesley, Cambridge, MA, 1949.