M. C. Easton

# Model for Database Reference Strings Based on Behavior of Reference Clusters

**Abstract:** The observation that references to a particular page are clustered (in time) in typical database reference strings is used as the intuitive motivation for a model of page reference activity in an interactive database system. The model leads to a two-parameter form for the (Denning) working-set functions associated with a page. Methods for estimating parameter values from measurements or from logical descriptions of applications are discussed. Results from the model are shown to agree well with measurements from two database systems.

## Introduction

Any study of the performance of a storage hierarchy requires information concerning the stream of requests that it must service. We consider the case of a two-level hierarchy with data movement in pages of fixed size. We propose a stochastic model for the *reference string*, i.e., the sequence of page requests to the hierarchy.

An important observation that has been made concerning reference strings (except in the case of sequentially accessed files) is that once a page is referenced, there are often additional references to it within a relatively short time. This phenomenon is sometimes called *(time) clustering* of references. We call the first access to a page after a "long" period of inactivity a *primary* reference to the page. The references that follow, within a "short" time, are called *secondary* references. (These concepts can be defined precisely, as we show later.) If the replacement policy holds recently referenced pages in the first level and if the first-level buffer is sufficiently large, then *page faults* (references not found in the first level) will be caused only by primary references. Thus, the page fault rate can be determined without detailed knowledge of the behavior of the secondary references. In the next sections, we develop a model for the behavior of primary references and use it to determine page fault rates for large buffers.

## Working-set functions

We first discuss a method, based on Denning's working-set description [1], for representing the behavior of references to an individual page. The underlying model is that the reference string is a realization of a stationary discrete-time stochastic process [2]. For each positive integer $T$, we consider the probability that page $i$ is referenced at time $t$ and is distinct from the references at times $t - T, t - T + 1, \cdots, t - 1$. The stationarity assumption implies that this probability does not depend on the value of $t$, so we denote this probability by $M_i(T)$. We also consider the probability that page $i$ is referenced at least once in the set of references at times $t - T + 1, t - T + 2, \cdots, t$. Again, stationarity implies that this probability does not depend on $t$; it is denoted $S_i(T)$. The functions $M_i(T)$ and $S_i(T)$ are called the *working-set functions* of page $i$. (The number of working-set functions that must be dealt with can be reduced by grouping pages with similar behavior into classes and by then constructing a pair of functions for each class.)

One motivation for considering these probabilities is that they immediately yield the page fault probabilities for the replacement algorithm known as the *working-set policy* [1]. Although this policy is not used in practice in the exact form stated below, it is similar to many policies actually used. The working-set algorithm requires one to set a parameter $T$, the *window size*, which is implicitly related to the capacity of the first level. Under this policy, a page is removed from the first level if it has not been referenced in the previous $T$ successive references. Thus, the probability that a page fault occurs at time $t$ (under working-set management) is simply the sum over $i$ of the $M_i(T)$. This sum is denoted $M(T)$. The probability that page $i$ is resident in the first level at time $t$ is $S_i(T)$. If we assign a random variable to page $i$ that takes value 1 if page $i$ is in the first level at time $t$, and 0 otherwise, then

the expected value of the sum of these random variables, which is the expected number of pages resident in the first level at time $t$, is just the sum over $i$ of the $S_i(T)$. We denote this sum $S(T)$. In summary, for each value of parameter $T$, we have the (overall) page fault probability, $M(T)$, and the corresponding value of the expected number of pages in the first level, $S(T)$. Experiments with a database reference string from the Advanced Administrative System (AAS) [3] have shown that this relationship agrees closely with the page fault rate vs capacity relationship observed using the LRU (replace the least-recently used page) algorithm. Bennett's [4] evaluation of the AAS miss ratio using the CLOCK (replace the first page scanned whose use-bit is off [5]) algorithm also agrees closely with the results from the working-set analysis. Theoretical ties between working-set and LRU miss ratios have been explored by Fagin [6].

An important feature of the working-set description is that the contribution of each page to fault rate and to buffer space utilization can be examined separately. This allows one to consider the impact on performance of selected portions of the database, such as the set of pages of a particular logical file or the set of pages resident on a particular physical device. The working-set description also makes possible examination of storage management strategies other than the one in which a single buffer is managed by an overall policy. For example, one might wish to hold all of file A (statically) in first-level storage, while using a buffer with working-set management for the other pages. More generally, one might have a different window size for each file (see [7] for a related discussion).

To further facilitate the use of working-set descriptions, we describe a model that leads to a two-parameter form for the working-set functions associated with a particular page.

### Model for initiation of reference clusters

In this section, a model for database references is presented. Before giving the formal description, we give an intuitive description using the notions of primary and secondary reference and of reference cluster. A reference to page $i$ is defined to be *primary* if the time (measured in references) since the last reference to it exceeds a particular value ($\tau$). Otherwise, the reference is *secondary*. A cluster begins with a primary reference and ends when no further secondary references are possible (because no reference to page $i$ has occurred in the previous $\tau$ references).

The basic assumption of the model is that once a cluster has ended, the time of initiation of the next cluster is a random variable with geometric distribution. Thus, $\tau$ can be viewed as a "memory time."

This model resembles one proposed by Guimaraes [8]. He suggested that each page could be in one of two modes. The page references occur in continuous time, with exponentially distributed interreference intervals. The parameter of the exponential distribution takes one of two possible values, depending on the current mode of the page. (Each page has the same stochastic behavior). An important difference between his model and ours is that we take into account the differences in behavior exhibited by different pages and, thus, are able to isolate the contribution of each page to $M(T)$ and $S(T)$. (A model for program paging in which the working set behavior for each page is described by a separate four-parameter function related to the Weibull distribution has been studied by Opderbeck and Chu [9].)

Some notation is required for a precise description of the model. We define a *reference sequence* $\boldsymbol{R} = \langle \cdots R_{-1}, R_0, R_1, \cdots \rangle$ to be a sequence of random variables. These variables take values from the set of page names $X = \{x_1, x_2, \cdots, x_n\}$. A realization of $\boldsymbol{R}$ is called a *reference string*. (As indicated above, $\boldsymbol{R}$ is assumed to be stationary.) The "finite memory time" assumption is

P1 There exist finite $\tau$ and probabilities $\rho_i$ such that for all $T \geq \tau$,

$$P\{R_t = x_i | R_{t-T} \neq x_i, R_{t-T+1} \neq x_i, \cdots R_{t-1} \neq x_i\} = \rho_i, \qquad i = 1, \cdots, n.$$

(One may, equivalently, assume that for each $i$ there is a finite $\tau_i$ and a $\rho_i$ such that the above probability statement holds for all $T \geq \tau_i$. If this is true, then P1 holds with $\tau$ equal to the largest of the $\{\tau_i\}$.)

Intuitively, the value of $\rho_i$ in P1 is the reciprocal of the mean time from the end of one cluster of references to page $i$ to the start of the next.

Note that if P1 holds with $\{\rho_i\}$ and $\tau = \tau_1$, then P1 holds with the same set of values $\{\rho_i\}$ and with $\tau = \tau_2$ for every $\tau_2 > \tau_1$. Thus, the values of the $\{\rho_i\}$ do not depend on the value of $\tau$.

From P1, we can readily deduce expressions for $S_i(T)$ and $M_i(T)$ for $T \geq \tau$. Let $Q_i(t, T)$ denote the event $\{R_{t-T} \neq x_i, R_{t-T+1} \neq x_i, \cdots, R_{t-1} \neq x_i\}$. Let $q_i(T) = P\{Q_i(t, T)\}$. The definition of $M_i(T)$ is then

$$M_i(T) = P\{R_t = x_i \quad \text{and} \quad Q_i(t, T)\}. \tag{1}$$

The definition of $S_i(T)$ is

$$S_i(T) = 1 - P\{Q_i(t+1, T)\} = 1 - q_i(T). \tag{2}$$

We first find an expression for $q_i(T)$. Note that $P\{Q_i(t+1, T+1)\} = P\{R_t \neq x_i | Q_i(t, T)\} P\{Q_i(t, T)\}$. It follows from this and P1 that for $T \geq \tau$,

$$q_i(T+1) = (1 - \rho_i)\, q_i(T). \tag{3}$$

From (3), we have

$$q_i(T) = (1 - \rho_i)^{T-\tau} q_i(\tau), \qquad T \geq \tau. \tag{4}$$

Equations (2) and (4) imply that

$$S_i(T) = 1 - [1 - S_i(\tau)](1 - \rho_i)^{T-\tau}, \qquad T \geq \tau. \tag{5}$$

If we write (1) as $M_i(T) = P\{R_t = x_i | Q_i(t, T)\}P\{Q_i(t, T)\}$, then we obtain from P1 and (4)

$$M_i(T) = \rho_i q_i(T) = \rho_i(1-\rho_i)^{T-\tau} q_i(\tau)$$
$$= M_i(\tau)\,(1-\rho_i)^{T-\tau}, \qquad T \geq \tau. \tag{6}$$

The first step in the derivation of (6) shows that

$$\rho_i = M_i(\tau)/[1 - S_i(\tau)]. \tag{7}$$

Therefore, for each $i$ we need only determine the values of $M_i(\tau)$ and $S_i(\tau)$ to compute $M_i(T)$ and $S_i(T)$ for all $T \geq \tau$. The expected number of pages in the first level and the overall page fault probability are obtained from

$$S(T) = \sum_{i=1}^{n} S_i(T); \; M(T) = \sum_{i=1}^{n} M_i(T). \tag{8}$$

## Examples

An example of a stationary reference sequence that satisfies P1 has been previously proposed as a model for a database reference string [10]. This model is a Markov chain having $n$ states, one for each member of $X$ (i.e., one for each page). There are parameters $\lambda_1, \cdots, \lambda_n$ and $r$ that satisfy

$$0 \leq r < 1, \quad \sum_{i=1}^{n} \lambda_i = 1, \lambda_i > 0 \text{ for } i = 1, \cdots, n.$$

The transition probabilities are

$$p_{ii} = r + (1 - r)\lambda_i,$$
$$p_{ij} = (1 - r)\lambda_j, \qquad i \neq j. \tag{9}$$

If we choose $\tau = 1$, then P1 is satisfied and $\rho_i = (1 - r)\lambda_i$. Also, it is not hard to show that $S_i(\tau) = \lambda_i$, $M_i(\tau) = (1 - r)(1 - \lambda_i)\lambda_i$. (The reader may verify that P1 holds if $\tau$ is chosen to be larger than 1, with no change in the values of $\{\rho_i\}$.) The special case of $r = 0$ corresponds to the well-known "i.i.d." (independent-identically distributed) random variable model.

The second example is described below informally by giving a method for constructing a realization. It can be formally described by a second-order Markov chain. (Many other examples can be constructed from Markov chains of order $\gamma$, where $\gamma$ is any desired positive integer.)

The first reference is chosen at random, with uniform probability, from the set $X$ (which contains at least three members). The second reference is chosen at random, with uniform probability, from the $n - 1$ members of $X$ that are distinct from the first reference. On each remaining step, a reference is chosen at random, with equal probability, from the set of $n - 2$ members of $X$ that did not appear in the previous two steps. It is obvious that with $\tau = 2$, then P1 is satisfied with $\rho_i = 1/(n - 2)$, $i = 1, \cdots, n$.

The two examples show markedly different behavior. In the first, if $r$ is close to 1, then a cluster for page $i$ contains one (generally) long sequence of successive references to page $i$. In the second model, each cluster for page $i$ contains a single reference to page $i$.

## Experimental results

The model was tested against two reference strings. The first consisted of $2 \times 10^6$ references to 1693-byte pages in the AAS database system, the same string used in [10]. Although the model described here is a generalization of the model used in the previous paper, the results obtained by application of the two models need not be the same. In [10], there is a fixed relationship between $\lambda_i$ (the probability of occurrence of $x_i$) and the conditional probability of occurrence of $x_i$, given that it was not also the previous reference. In the model described here, this assumption is relaxed. In [10], the values of $\{\lambda_i\}$ were estimated from the reference string, but the value of the parameter relating $\lambda_i$ to the conditional probability mentioned above was estimated by an ad hoc procedure. When testing the model described here, as shown below, the values of parameters $M_i(\tau)$ and $S_i(\tau)$ were directly estimated from measurements, and, hence, so were the conditional probabilities that appear in P1.

The second reference string analyzed, also of length about $2 \times 10^6$ references, is from an IMS installation [11] used for engineering analysis in a manufacturing environment. The original data consisted of a map of the database at the start of the measurement and a sequence of DL/1 calls that was subsequently executed. Several analyses of these data have been carried out (e.g., [12, 13]). The work described here was based on a sequence that gives, for each reference, the origin of access and the number of bytes transferred. This sequence was constructed by J. Mommens, who developed a technique for expanding each DL/1 call. Mommens' sequence was then converted to a sequence of references to 4096-byte pages for use in this study.

As a first test of assumption P1, the following function was considered:

$$G(T) = \sum_{i=1}^{n} M_i(T)/[1 - S_i(T)]. \tag{10}$$

If P1 holds and $T \geq \tau$, then, by (5), (6), and (7), $G(T) = \Sigma_{i=1}^{n}\rho_i$. That is, $G(T)$ is constant for $T \geq \tau$. Estimates of values for $M_i(T)$ and $S_i(T)$, denoted, respectively, $\hat{m}_i(T)$ and $\hat{s}_i(T)$, were obtained for each $i$ and for a number of values of $T$ by use of the "transient-free" estimators described in [14] (see Appendix). These estimates were used in the evaluation of (10). The results, displayed for the

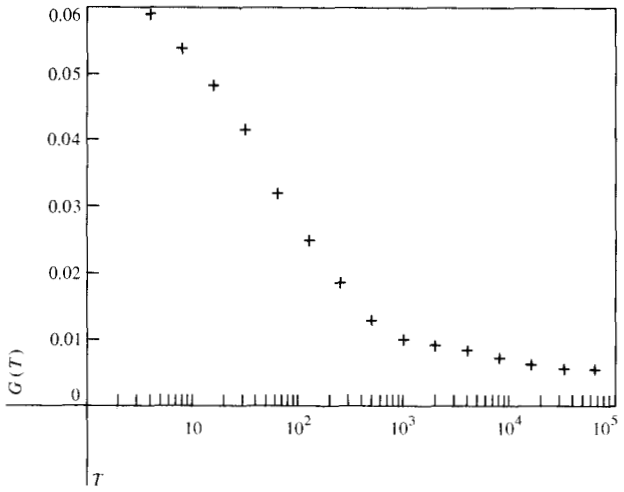


**Figure 1** Estimate of function $G(T)$ from IMS string.

IMS string in Fig. 1, show a leveling off of the measured values of $G(T)$ in the range $10^4 \leq T \leq 10^5$. For convenience, the values chosen for $T$ in preparing Fig. 1 were of the form $2^j$; the value $\tau = 2^{15} \approx 3.3 \times 10^4$ was selected for the comparison of observed working-set values with those resulting from the model. [If P1 holds with the value of $\tau$ selected, then any value larger than this could also be used, but with consequent restriction of the range of validity of Eqs. (5), (6), and (8).] Results similar to that of Fig. 1 were obtained for the AAS string. Coincidentally, $\tau = 2^{15}$ was also selected for the AAS comparison.

With the exceptions noted below, $\rho_i$ was estimated from (7), and then (5) and (6) were used in (8). As indicated in the Appendix, the estimates $\hat{s}_i(\tau)$ and $\hat{m}_i(\tau)$ are based on the lengths of and number of occurrences of intervals of length exceeding $\tau$ between successive appearances of $x_i$. However, for many pages, only one cluster of references occurred in the observed string. A long period of no activity was generally observed preceding and/or following each such cluster. For each such page, it was assumed that $\rho_i k << 1$, where $k$ is the number of references in the observed string. Under this assumption, and for $T - \tau < k$, the following approximate forms of (5) and (6) can be used:

$$S_i(T) \approx 1 - [1 - S_i(\tau)][1 - (T - \tau)\rho_i]$$
$$= (T - \tau)M_i(\tau) + S_i(\tau), \quad (11)$$

$$M_i(T) \approx M_i(\tau). \quad (12)$$

Let $L$ be the set of indices for which (11) and (12) are used in obtaining an estimate for $S(T)$ from (8). Then the part of each sum in (8) that involves indices in $L$ depends only on

$$\sum_{i \in L} \hat{m}_i(\tau) \quad \text{and} \quad \sum_{i \in L} \hat{s}_i(\tau).$$

It was assumed that the latter sums provided reasonable estimates for the contribution of low activity pages to the sums in (8). The rare case in which $\hat{s}_i(\tau) = 1$ (and hence $\hat{m}_i(\tau) = 0$) was also handled by using (11) and (12) since (7) could not be used to estimate $\rho_i$. In general, for pages having $\hat{s}_i(\tau) \approx 1$, inaccuracies in estimation may cause significant errors in the computation of $M_i(T)$. However, in the two strings examined here, there were few pages with this property, so the overall effect on $M(T)$ was not significant. [For the evaluation of (10) described previously, where the values of $\rho_i$ appear only in a simple sum, the one or two cases with $\hat{s}_i(\tau) = 1$ were omitted and all other values of $\rho_i$ were computed by (7).]

In Figs. 2 and 3, the results from the model are seen to agree well with estimates of the expected working-set values obtained from the sample strings. The latter estimates were computed by the methods described in [14].

Some additional insight into the referencing pattern can be gained by examining the values estimated for the $\{\rho_i\}$. For most values of $i$, $\rho_i$ was much smaller than $P\{R_t = x_i\}$. For each such $i$ it can be concluded that $P\{R_t = x_i | \sim Q_i(t, \tau)\}$ is larger than $P\{R_t = x_i\}$. In words, for most $i$ the conditional probability of referencing page $i$, given the page *was* referenced in the previous $\tau$ references, is greater than the unconditional probability of a reference to page $i$. This observation supports the intuitive notion that pages recently referenced are more likely than others to be referenced next.

### Miss ratios from logical descriptions of an application

In some instances, a description of a database system, its application programs, and information concerning "typical" transactions will be sufficient to allow as reasonable the assumption that P1 will hold for a particular value of $\tau$. In such cases, the information available may provide estimates for parameter values so that miss ratios can be computed.

As a simple example, suppose that a consumer credit company wants to provide on-line verification of credit card accounts. Suppose that the service is available 12 hours a day, 6 days a week, and processes requests at a uniform rate of 1000 per hour. There are $n = 18\,000$ credit card holders. Uses of the charge card are assumed to be clustered in time, with fairly long intervals between clusters. On the average, there is one period per week of use of the card, averaging four hours between first and final use. If the card has not been used for a duration of two hours, then it can be assumed that the customer has terminated this active period.

If we assume that P1 is a reasonable assumption for the stream of credit requests, then we can estimate the parameter values as follows. First we set $\tau = 2000$, since that corresponds to a two-hour period of inactivity. In each 72 000 references, a particular customer, on aver-

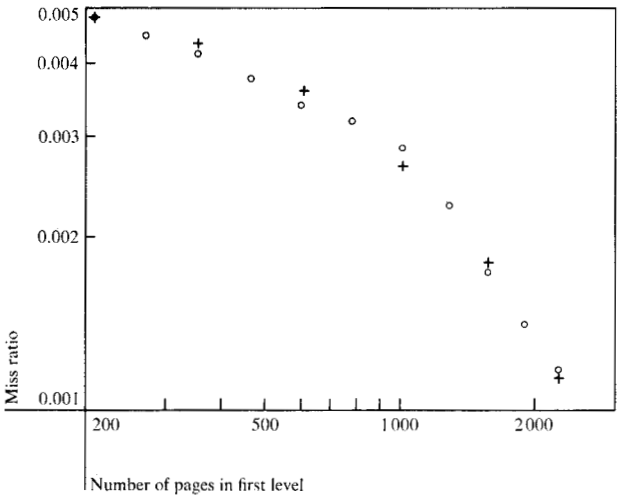


**Figure 2** IMS results in which o is estimate from sample string, + is expected value from model.

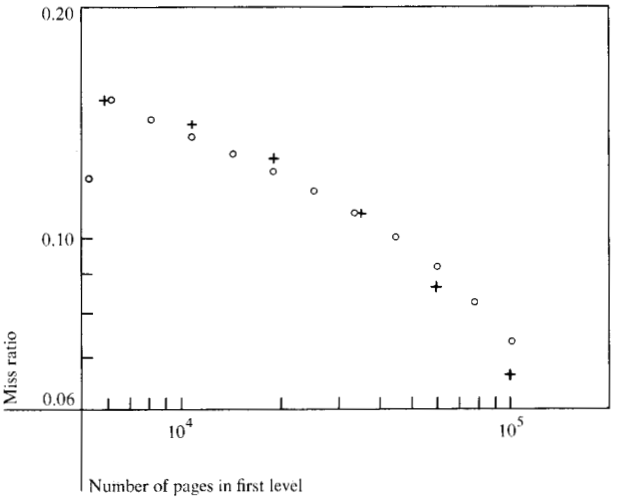


**Figure 3** AAS results in which o is estimate from sample string, + is expected value from model.

age, causes one "fault" to a window of size $\tau$ so $M_i(\tau) = 1/72\,000$, $i = 1, \cdots, n$. Since, on average, the system processes 4000 references between an individual's first and last use of his card, there will be, under working-set management with $\tau = 2000, 4000 + 2000$ references while the individual's record is in the first level. Therefore, our estimate for $S_i(\tau)$ is 6/72. The estimate for $\rho_i$ [by (7) or by a direct argument] is 1/66 000. (Note that a cluster, by definition, ends two hours after the last reference. Thus, 66 000 references occur on average from the end of one cluster for customer $i$ to the beginning of the next.) Then (5), (6), and (8) yield the values in Table 1.

**Table 1** Values from credit card model.

| $T$ | $M(T)$ | $S(T)$ |
|---|---|---|
| 2000 | 0.250 | 1500. |
| 16000 | 0.202 | 4654. |
| 32000 | 0.159 | 7527. |

## Applicability of the model

The results on the two test cases showed that the model yielded accurate predictions of page fault probability and mean storage utilization. However, the values used for $\tau$ may be so large that the restriction $T \geq \tau$ will limit the model to studies of buffers that are much larger than current main memory buffers. If the main memory buffer is extended, say by use of electronic disks [15], then the range of applicability of the model becomes suitable to the application.

Another consideration in applying the model is the possibility of significant fluctuations in the workload over extended periods of time. As one indication of this, the number of LRU page faults for a particular buffer size was examined over successive intervals of $5 \times 10^5$ references. In the case of AAS, the fluctuations from interval to interval were small. In the case of IMS, however, the number of faults in one such interval was less than 700, while in another it exceeded 5600. (However, the portion of the string used in the analysis described earlier was chosen in a period of less dramatic fluctuations.) Therefore, one must be cautious about assuming that behavior measured or modeled in one time period will be maintained in another. A detailed intuitive understanding of the nature of the workload at the time of a measurement can aid in attributing any degree of generality to the results obtained.

The model described here does not deal explicitly with correlations between references by one page and references by another. Results shown in [1] and [10] imply that the working-set functions for page $i$ are determined by the distribution of interreference time (time between successive references) for page $i$. Thus, it is not surprising that an approach that looks at behavior of individual pages can be successful in modeling working-set functions. However, the model described here is not suited to addressing such questions as the form of the distribution of time between successive page faults (which are generally to different pages) or the impact of grouping pages together into larger pages (blocking). In the case of blocking, however, the model can be re-applied to a particular reference string using a different blocking factor. The effect of the block size on the parameters can then be observed.

## Conclusions

Database referencing patterns are known to be difficult to describe analytically. In the case of large buffers, however, it becomes easier to predict miss ratios without a detailed model. The work described here concentrates on the pages that have not been referenced for a relatively long period of time. The assumption that the probability of referencing such a page is independent of the previous history of the reference string leads to predictions of miss ratios that agree closely with measurements from two interactive database systems.

## Appendix: Method for estimating parameter values

Let $\boldsymbol{r} = \langle r_1, r_2, \cdots, r_k \rangle$ be a reference string of length $k$. A $T$-substring of $\boldsymbol{r}$ is a sequence $\langle r_{t-T+1}, r_{t-T+2}, \cdots, r_t \rangle$. In $\boldsymbol{r}$, the first $T$-substring terminates at $t = T$ and the last at $t = k$. We estimate a value for $M_i(\tau)$ by finding $N_m$, the number of times that $r_{t+1} = x_i$ and $x_i$ does not appear in the $\tau$-substring that terminates at $t$, $t = \tau, \tau + 1, \cdots, k - 1$. We estimate a value for $S_i(\tau)$ by finding $N_s$, the number of times that $x_i$ appears in the $\tau$-substring that terminates at $t$, $t = \tau, \tau + 1, \cdots, k$. The estimated values, denoted, respectively, $\hat{m}_i(\tau)$ and $\hat{s}_i(\tau)$, are obtained as follows:

$$\hat{m}_i(\tau) = N_m/(k - \tau), \tag{A1}$$

$$\hat{s}_i(\tau) = N_s/(k - \tau + 1). \tag{A2}$$

As shown in [14], these computations can be carried out conveniently by examining the lengths of intervals between successive appearances of $x_i$ in $\boldsymbol{r}$.

## References


1. P. J. Denning and S. C. Schwartz, "Properties of the Working-Set Model," *Commun. ACM* **15,** 191 (March 1972).
2. A. M. Yaglom, *Stationary Random Functions*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1962.
3. J. H. Wimbrow, "A Large-Scale Interactive Administrative System," *IBM Syst. J.* **10,** 261 (1971).
4. B. T. Bennett, private communication.
5. M. C. Easton and P. A. Franaszek, "Use-bit Scanning in Replacement Decisions," *Proceedings of the Fifth Texas Conference on Computing Systems*, October 1976, p. 160.
6. R. Fagin, "Asymptotic Miss Ratios over Independent References," *J. Computer Syst. Sci.* **14,** 2 (April 1977).
7. M. Ghanem, "Dynamic Partitioning of the Main Memory Using the Working Set Concept," *IBM J. Res. Develop.* **19,** 445 (1975).
8. C. C. Guimaraes, *Queuing Models with Applications to Scheduling in Operating Systems*, Jennings Computer Center, Case Western Reserve University, Cleveland, OH, 1973.
9. H. Opderbeck and W. W. Chu, "The Renewal Model for Program Behavior," *SIAM J. Computing* **4,** 356 (September 1975).
10. M. C. Easton, "Model for Interactive Data Base Reference String," *IBM J. Res. Develop.* **19,** 550 (November 1975).
11. *IMS/360 General Information Manual*, GH20-0765, IBM Corporation, White Plains, NY, 1973.
12. W. G. Tuel, Jr. and J. Rodriguez-Rosell, "A Methodology for Evaluation of Data Base Systems," *Research Report RJ1668*, IBM Research Division, San Jose, CA, 1975.
13. D. P. Gaver, S. S. Lavenberg, and T. Price, "Exploratory Analysis of Access Path Length Data for an IMS Installation," *Research Report RJ1736*, IBM Research Division, San Jose, CA, 1976.
14. M. C. Easton and B. T. Bennett, "Transient-free Working Set Statistics," *Commun. ACM* **20,** 90 (February 1977).
15. J. H. Wensley, "The Impact of Electronic Disks on System Architecture," *Computer* **8,** 44 (February 1975).


*Received March 22, 1977; revised October 24, 1977*

*The author is located at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598.*