Properties of the Working-Set Model

Peter J. Denning and Stuart C. Schwartz
Princeton University*

A program's working set $W(t, T)$ at time t is the set of distinct pages among the T most recently referenced pages. Relations between the average working-set size, the missing-page rate, and the interreference-interval distribution may be derived both from time-average definitions and from ensemble-average (statistical) definitions. An efficient algorithm for estimating these quantities is given. The relation to LRU (least recently used) paging is characterized. The independent-reference model, in which page references are statistically independent, is used to assess the effects of interpage dependencies on working-set size observations. Under general assumptions, working-set size is shown to be normally distributed.

Key Words and Phrases: working-set model, paging, paging algorithms, program behavior, program modeling

CR Categories: 4.3

Presented at the Third ACM Symposium on Operating Systems Principles, Palo Alto, California, October 18–20, 1971.

1. Introduction

Although computer memory systems have always used several levels of storage media, it was not until multiprogramming and virtual memory came into existence that the problems of managing and evaluating them efficiently became severe. The complexity of computer memory systems—besides the lack of reliable prior information about the memory demands of programs running in them—has stimulated interest in analytic program-behavior models from which adaptive memory management policies can be derived. The working-set model for program behavior [6, 7] has proved to be a useful starting point. A program's working set is, intuitively, the smalle stsubset of its pages that must reside in main memory in order that the program operate at some desired level of efficiency.[1] The working-set principle of memory management states that a program may use a processor only if its working set is in main memory, and that no working-set page of an active program may be considered for removal from main memory. Simulation results on the RCA Spectra 70/46 [16] and experimental observations of IBM TSS/360 [9] provide evidence that this principle is viable.

Programs, to one degree or another, obey the *principle of locality* which asserts: (1) during any interval of time, a program distributes its references nonuniformly over its pages; (2) taken as a function of time, the frequency with which a given page is referenced

* Department of Electrical Engineering, Brackett Hall, Engineering Quadrangle, Princeton, NJ 08540. The work reported herein was supported in part by NASA Grant NGR-31-001-170 and by NSF Grants GK-13193 and GJ-30126.

[1] The results of this paper are formulated in the context of paging; generalizations to nonpaging systems are straightforward and not considered here.

tends to change slowly, i.e. it is quasi-stationary; and (3) correlation between immediate past and immediate future reference patterns tends to be high, whereas the correlation between disjoint reference patterns tends to zero as the distance between them tends to infinity. This principle is an abstraction of at least three phenomena observed in practice. First, programs use sequential and looping control structures heavily, and they cluster references to given pages in short time intervals. Second, programmers tend to concentrate on small parts of large problems for moderately long intervals; simple alterations in algorithm strategy and data organization can magnify the manifestations of this effect manyfold [3, 14]. Third, programs may be run efficiently with only a subset of their pages in main memory [2, 11, 14].

According to the above, we may picture a program as making transitions from time to time among "localities," a locality being some subset of its pages. The pages in the "current locality" are referenced with high probability. Ideally, a program's working set should comprise the pages of its current locality. We define a program's working set $W(t, T)$ at time t to be the set of distinct pages it has referenced among the T most recent page references [6]. This definition uses the principle of locality to form an estimate of immediate future memory demand on the basis of immediate past reference patterns.

This paper presents some new analysis and results for program behavior. Informally stated, it will be shown that the interreference-interval density function is the negative slope of the missing-page-rate function, which, in turn, is the slope of the average-working-set-size function. In order to underscore the generality of the results and their applicability to practical measurement, the derivations emphasize time, rather than stochastic, analysis.

2. Definitions

Consider an n-page program whose pages constitute the set $N = \{1, 2, \ldots, n\}$. Neither the page size nor the manner in which address-space words are distributed among the pages are of concern here. The dynamic behavior of the program for given input data can be modeled in machine-independent terms by its *reference string*, which is a sequence $\rho = r_1 r_2 \cdots r_t \cdots$, each r_t being in N. If $r_t = i$ we understand that page i was referenced at the tth reference; thus t measures process time, which is discrete.

The analysis of this paper is based on three assumptions related to those of locality:

W1. Reference strings are unending.

W2. The stochastic mechanism underlying the generation of a reference string ρ is stationary, i.e. independent of absolute time origin.

W3. For all $t > 0$, r_t and r_{t+x} become uncorrelated as $x \to \infty$.

Since reference strings (or substrings of interest) generated by practical programs are long from a statistical standpoint—hundreds or thousands or more of references—the error introduced by assumption W1 is not significant. Assumption W3, which can be summarized by "references are asymptotically uncorrelated," is almost always met in practice. Assumption W2 does restrict the results somewhat, limiting the analysis to the context of a single program locality in the following sense. As mentioned above, a program passes through a sequence of localities as it generates references. One would expect that whatever nonstationarities exist depend only on the locality. In other words, we could approximate a reference string ρ as a sequence of substrings $\rho = \rho_1 \rho_2 \cdots \rho_i \cdots$, substring ρ_i being generated by a stationary model obeying W1–W3. Therefore the results are applicable locally in a given reference string, but not necessarily globally. Since our primary interest is understanding the behavior of the working-set model as an adaptive estimator for use in memory management, assumption W2 will not be severe as long as the measurement intervals are comparable to or less than the average interlocality transition time.

Following are definitions of working set, working-set size, missing-page rate, and interreference distributions. We have elected to present the definitions as time averages, rather than stochastic averages, in order to make their applicability in practical measurement more evident.

A program's *working set* $W(t, T)$ at time t is the set of distinct pages referenced in the time interval $[t - T + 1, t]$, i.e. among the $T \geq 1$ most recent references $r_{t-T+1} \cdots r_t$. If $t < T$, $W(t, T)$ contains only the distinct pages among $r_1 r_2 \cdots r_t$; if $t \leq 0$, $W(t, T)$ is empty. The parameter T is called the "window size," since $W(t, T)$ can be regarded as the contents of a window looking backward at the reference string. The *working-set size* $w(t, T)$ is the number of pages in $W(t, T)$. Let

$$s_k(T) = \frac{1}{k} \sum_{t=1}^{k} w(t, T) \tag{2.1}$$

denote the working-set size averaged over the first k references; we define the *average working-set size* to be

$$s(T) = \lim_{k \to \infty} s_k(T). \tag{2.2}$$

The existence of this limit is guaranteed by our assumption of stationarity (W2).

In Section 5, we shall define $s(T)$ as a stochastic average, i.e. $s^*(T) = \sum_j p_j$ where $p_j = \Pr[w(t, T) = j]$. In general, stationarity is not a sufficient condition for the time-average working-set size (2.1) to converge to the stochastic-average working-set size. According to the development in [15, pp. 16–22] however, W3 is a sufficient additional condition; it guarantees that the time average converges to the stochastic average in probability:

$$\lim_{k\to\infty} \Pr [| s_k(T) - s^*(T) | > \epsilon] = 0, \quad \text{any } \epsilon > 0. \tag{2.3}$$

The limits given below (eqs. (2.5), (2.6), (2.8)) converge under these same conditions. It should be noted that the following analysis depends on W1 and W2 only; W3 is required for convergence of time-average definitions to the corresponding stochastic quantities.

The *missing-page rate* $m(T)$ measures the number of pages per unit time returning to the working set. In systems using the working-set principle of memory management, a page may leave the working set and return without being removed from main memory in the meantime; therefore $m(T)$ will be an upper bound on the page-fault rate experienced by the program. Define the binary variable, for $t \geq 0$,

$$\Delta(t, T) = \begin{cases} 1 \text{ if } r_{t+1} \text{ is not in } W(t, T), \\ 0 \text{ otherwise.} \end{cases} \tag{2.4}$$

Note that $\Delta(0, T) = 1$ for all T since r_1 is never in $W(0, T)$, which is empty. The missing-page rate is defined to be

$$m(T) = \lim_{k\to\infty} \frac{1}{k} \sum_{t=0}^{k-1} \Delta(t, T). \tag{2.5}$$

In the next section we shall show that $m(T)$ can be regarded as the slope of $s(T)$.

Suppose that in reference string $r_1 r_2 \cdots r_t \cdots$ two successive references to page i occur at times t and $t + x_i$. We call x_i an *interreference interval* for page i. The interreference distribution for page i is defined to be

$$F_i(x) = \lim_{k\to\infty} \left[\frac{\text{no. } x_i \text{ in } r_1 \cdots r_k \text{ with } x_i \leq x}{\text{no. } x_i \text{ in } r_1 \cdots r_k} \right], \tag{2.6}$$

i.e. $F_i(x)$ is the fraction of x_i's for which $x_i \leq x$. The interreference density function for page i is defined to be

$$f_i(x) = F_i(x) - F_i(x - 1), \tag{2.7}$$

so that $f_i(x)$ is interpreted as the fraction of x_i's for which $x_i = x$. Define also the relative frequency of references to page i,

$$\lambda_i = \lim_{k\to\infty} \frac{1}{k} [\text{no. refs. to page } i \text{ in } r_1 \cdots r_k]. \tag{2.8}$$

Note that $\lambda_1 + \cdots + \lambda_n = 1$. The overall density and distribution functions are defined, respectively, to be

$$f(x) = \sum_{i=1}^{n} \lambda_i f_i(x), \quad F(x) = \sum_{i=1}^{n} \lambda_i F_i(x), \tag{2.9}$$

and the mean overall interreference interval is

$$\bar{x} = \sum_{i=1}^{n} \lambda_i \bar{x}_i, \quad \text{where} \quad \bar{x}_i = \sum_{x \geq 0} x f_i(x).$$

Page i will be called "recurrent" if $\lambda_i \neq 0$, and n_R will denote the number of recurrent pages in N. If $\lambda_i \neq 0$,

$$\bar{x}_i = 1/\lambda_i, \tag{2.10}$$

so that λ_i can be interpreted as the rate at which the program references page i. To see this, consider $r_1 r_2 \cdots r_k$: The average number of references to page i is $\lambda_i k$, and the average distance between two of them is $k/\lambda_i k$. If $\lambda_i = 0$, assumption W2 (stationarity) can be used to show that $\lambda_i \bar{x}_i = 0$; together with (2.11), this implies that the mean overall interreference interval $\bar{x}$ of (2.10) satisfies

$$\bar{x} = n_R. \tag{2.11}$$

One can show also that assumption W2 implies that a nonrecurrent page i is referenced at most a finite number of times in $r_1 r_2 \cdots r_t \cdots$. It follows that nonrecurrent pages make no contribution in the limit to the definitions of $s(T)$, $m(T)$, or $F_i(x)$. In the following analysis, therefore, we shall assume $n = n_R$ unless otherwise specified.

Equation (2.12) asserts that, the more recurrent pages a program has, the longer will be its mean overall interreference interval; since $W(t, T)$ can be defined as the set of pages i whose current interreference intervals satisfy $x_i \leq T$, this implies that, the more recurrent pages a program has, the larger will its working set tend to be.

Later on, we shall interpret the f_i and F_i as probability density and distribution functions, respectively, and λ_i as the long-run (i.e. over a locality) probability page i is referenced.

3. Time Analysis

The analysis in this section establishes the relationships between the functions $s(T)$, $m(T)$, and $f_i(x)$. We are referring to it as "time analysis" since the time-average definitions given earlier will be used, and with the exceptions of assumptions W1 and W2, nothing will be assumed about the statistical properties of reference strings. Even though the results are straightforward to derive, the lack of assumptions about reference-string structure suggests their generality.

Fig. 1

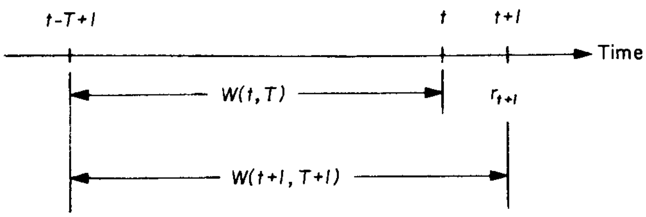
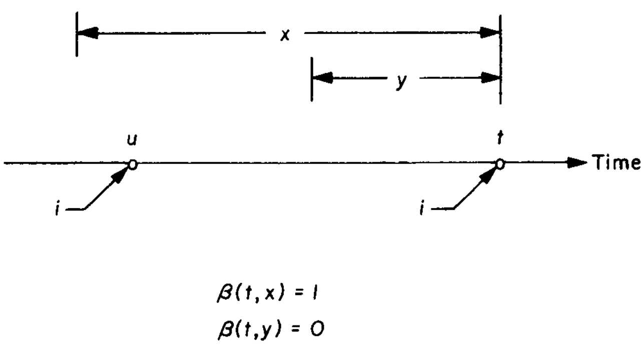


Fig. 2



The average working-set size $s(T)$, missing-page rate $m(T)$, overall interreference density $f(x)$, overall interreference distribution $F(x)$, and number of recurrent $n_R = n$ pages satisfy these properties:

P1. $1 = s(1) \leq s(T) \leq s(T+1) \leq \min\{n, T+1\}$.
P2. $s(T+1) - s(T) = m(T)$.
P3. $0 \leq m(T+1) \leq m(T) \leq m(0) = 1$.
P4. $m(T) = 1 - F(T) = \sum_{y>T} f(y)$.
P5. $m(T+1) - m(T) = -f(T+1)$.
P6. $s(T) = \sum_{z=0}^{T-1} m(z) = \sum_{z=0}^{T-1} (1 - F(z)) = \sum_{z=0}^{T-1} \sum_{y>z} f(y)$.
P7. $[s(T-1) + s(T+1)]/2 \leq s(T)$.
P8. $\lim_{T\to\infty} s(T) = n_R$.
P9. $\lim_{T\to\infty} m(T) = 0$.

Property P1 states that $x(T)$ is nondecreasing and is bounded above and below. It follows immediately from the definition and $W(t, T) \subseteq W(t, T+1)$.

Property P2 states that the "slope" of $s(T)$ is the missing-page rate. From Figure 1 and (2.4),

$$w(t+1, T+1) = w(t, T) + \Delta(t, T). \tag{3.1}$$

Summing both sides from $t = 0$ to $k - 1$, dividing by k, taking the limit as $k \to \infty$, and applying (2.2) and (2.5), we have $s(T+1) = s(T) + m(T)$, which establishes P2.

Property P3 states that $m(T)$ is nonincreasing in T. (The lower and upper bounds follow immediately from the definition of $m(T)$.) To prove P3, we shall show that

$$\Delta(t, T+1) \leq \Delta(t, T). \tag{3.2}$$

If this is true, we can sum both sides from $t = 0$ to $k - 1$, divide by k, take the limit as $k \to \infty$, and apply (2.5) to conclude $m(T+1) \leq m(T)$. If $\Delta(t, T+1) = 0$, relation (3.2) clearly holds. If $\Delta(t, T+1) = 1$, then by (2.4) r_{t+1} is not in $W(t, T+1)$; then $W(t, T) \subseteq W(t, T+1)$ implies $\Delta(t, T+1) = 1$ also. This establishes P3.

Property P4 states that $m(T)$ can be regarded as the probability $x > T$. Define the binary variable $\beta_i(t, x)$ to be 1 if and only if $r_t = r_u = i$ for some $u < t$ such that $t - u \leq x$ (cf. Figure 2). Define $n_i(k)$ to be the number of references to page i in $r_1 \cdots r_k$. The definition of $F(x)$ (eqs. (2.6) and (2.9)) can be expressed as

$$\begin{aligned} F(x) &= \sum_{i=1}^{n} \lambda_i F_i(x) \\ &= \sum_{i=1}^{n} \lim_{k\to\infty} \left(\frac{n_i(k)}{k}\right)\left(\frac{1}{n_i(k) - 1} \sum_{t=1}^{k} \beta_i(t, x)\right) \\ &= \lim_{k\to\infty} \frac{1}{k} \sum_{t=1}^{k} \sum_{i=1}^{n} \beta_i(t, x) \end{aligned} \tag{3.3}$$

where we have used the assumption that each page i is recurrent, so that $n_i(k) \to \infty$. Now, define $\beta(t, x) = \beta_1(t, x) + \cdots + \beta_n(t, x)$, and observe that $\beta(t, x) = 1 - \Delta(t - 1, x)$. Therefore

$$\begin{aligned} 1 - F(x) &= \lim_{k\to\infty} \frac{1}{k} \sum_{t=1}^{k} (1 - \beta(t, x)) \\ &= \lim_{k\to\infty} \frac{1}{k} \sum_{t=1}^{k} \Delta(t - 1, x) \\ &= m(x). \end{aligned} \tag{3.4}$$

This establishes P4.

Property P5 is a statement that the "slope" of $m(T)$—the "second slope" of $s(T)$—is the negative value of the overall interreference density f. It follows immediately from P4.

Property P6 gives formulas for calculating $s(T)$. It follows immediately from P2 and substitution of P4.

Property P7 states that the curve $s(T)$ is concave down. From P3, $m(T) \leq m(T-1)$; substituting the result of P2 on both sides, we obtain

$$s(T+1) - s(T) \leq s(T) - s(T-1), \tag{3.5}$$

which upon rearrangement of terms becomes P7.

Property P8 states that the limiting average working-set size is n_R. To see this, consider from P6

$$\lim_{T\to\infty} s(T) = \sum_{z\geq 0} \sum_{y>z} f(y). \tag{3.6}$$

If one expands the right side of (3.6), one finds that it can be rewritten in the form $\sum_z z f(z)$, which is the definition of $\bar{x}$. By (2.12), $\bar{x} = n_R$, and P8 is established.

Finally, Property P9 follows from P4 and the fact that $F(T) \to 1$ as $T \to \infty$.

The results of P1–P9 are summarized in Figures 3 and 4. In these figures T is represented as a real number, and

Fig. 3

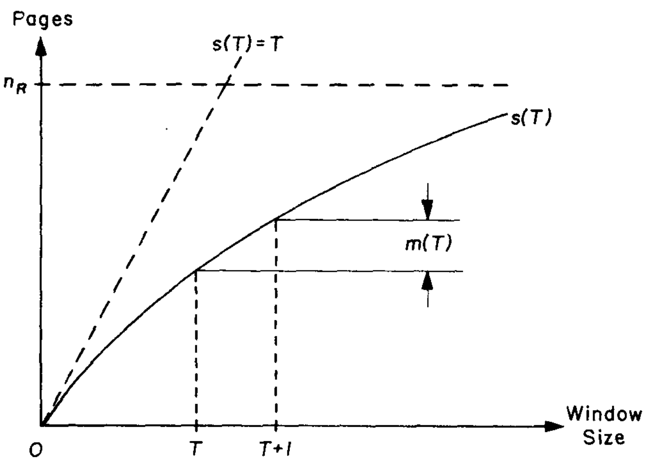
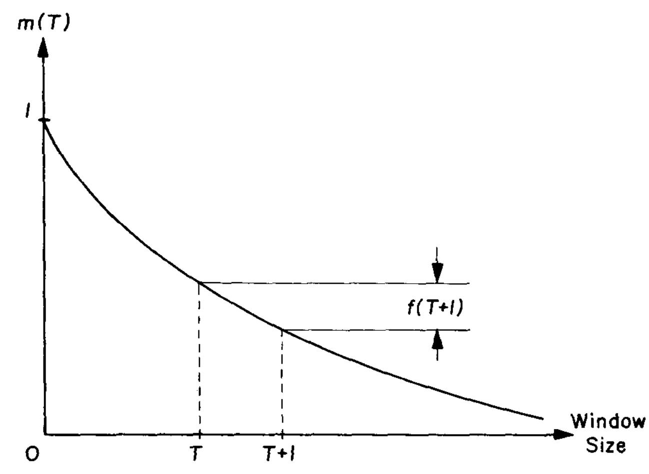


Fig. 4



$s(T)$ and $m(T)$ have been extended to piecewise linear functions.

The result P6 can be used as the basis for a one-pass algorithm that estimates the functions f, F, m, and s for a given reference string $r_1 r_2 \cdots r_k$. (This algorithm is analogous to one suggested by Mattson et al. for measuring page fault rates [12].) Let $(c_1, \cdots, c_{L+1})$ be a vector of integer variables, initially all 0; after the tth iteration of the algorithm $(1 \leq t \leq k)$, c_j $(1 \leq j \leq L)$ will count the number of interference intervals of length j in $r_1 r_2 \cdots r_t$, and c_{L+1} the number of interreference intervals of length $L + 1$ or greater. An array TIME[$1:n$] can be used to record the times of the most recent reference to pages. Consider each successive r_t. If $r_t = i$ is the first reference to i, set TIME[i] $= t$. If $r_t = i$ is not the first reference to i, let $j = t -$ TIME[i] and set TIME[i] $= t$; then add 1 to c_j if $j \leq L$ and to c_{L+1} otherwise. At the termination of this procedure (after $t = k$), one may estimate the interreference density function from

$$\hat{f}(x) = c_x/(c_1 + \cdots + c_{L+1}) \tag{3.7}$$

and apply P1–P9 to obtain estimates $\hat{F}$, $\hat{m} = 1 - \hat{F}$, and $\hat{s}$ of the functions F, m, and s, respectively. The estimates will be good if k is large and L large enough so that $c_{L+1} \ll c_1 + \cdots + c_L$.

Two guidelines for the choice of window size T are immediately deducible from P1–P9. First, a specified lower bound on efficiency would imply an upper bound on $m(T)$, and in turn a lower bound on T(property P3). Second, the concave-down property (P7) of $s(T)$ indicates that varying T need not be advantageous. Suppose T is varied in some manner, let $g_T(u)$ denote the fraction of time at which $T = u$, and let

$$\bar{T} = \sum_u u g_T(u)$$

be the average value of T. The equation of the straight-line tangent to $s(T)$ at $T = \bar{T}$ is $s(\bar{T}) + m([\bar{T}])(T - \bar{T})$, where $[\bar{T}]$ is the largest integer not exceeding $\bar{T}$; since $s(T)$ is concave down, this tangent is an upper bound on $s(T)$:

$$s(\bar{T}) + m([\bar{T}])(T - \bar{T}) \geq s(T), \quad T \geq 0. \tag{3.8}$$

Taking averages with respect to $g_T(u)$ on both sides of (3.8), we have

$$s(\bar{T}) \geq \overline{s(T)}. \tag{3.9}$$

In words, eq. (3.9) states that the variation on T with average value $\bar{T}$ produces an average working-set size smaller than when holding T fixed at $\bar{T}$. A similar argument can be applied to the curve $m(T)$ which (by P5) is concave up in every range of T over which $f(T)$ is nonincreasing:

$$\overline{m(T)} \geq m(\bar{T}). \tag{3.10}$$

In this case, varying T would have the additional effect of increasing the missing-page rate.

Interarrival distributions and densities encountered in practice frequently have nonincreasing tails [4, 10]. Assuming $f(x)$ is such a density, there exists an x_0 (relatively small compared to $\bar{x}$) such that $f(x)$ is nonincreasing for $x \geq x_0$. By P5, $m(T)$ would be concave up for $T > x_0$ and (3.10) would hold.

4. Relation to LRU Paging

There is an intimate relationship between LRU (least recently used) paging and the working-set model. The curve $m(T)$ can be used to estimate the page-fault rate of LRU paging, and the working-set model can be used to simulate LRU paging.

The LRU paging algorithm is a demand paging algorithm that operates in a fixed memory space of k pages ($1 \leq k \leq n$). At each page fault, LRU replaces from memory the page which has not been referenced for the longest time; thus the LRU memory always contains the k most recently used pages. By comparison, $W(t, T)$ always contains the $w(t, T)$ most recently used pages. Therefore the slope of $s(T)$ at $s(T) = k$ can be used to estimate the page-fault rate $L(k)$ of LRU in a k-page memory, as suggested in Figure 5, i.e. $L(k) \cong m(T_k)$.

To use the working-set model to simulate LRU paging, we vary T so that $W(t, T)$ always contains precisely k pages, in which case $W(t, T)$ will be precisely the contents of LRU's memory. Letting $T(t, k)$ denote the smallest value of T for which $w(t, T) = k$, we have

$$w(t, T(t, k)) = k. \tag{4.1}$$

Let $\bar{T}_k$ denote the average value of $T(t, k)$ over all t. The constraint (4.1) implies that $s(\bar{T}_k) = k$, whence $\bar{T}_k$ is precisely T_k, as shown in Figure 5. Now, suppose $m(T)$ is concave up in the interval of T over which $T(t, k)$ varies. Using (3.10), we have $\overline{m(T)} \geq m(T_k)$, where $\overline{m(T)}$ is the average value of $m(T)$ resulting from the variation of $T(t, k)$. Since $W(t, T(t, k))$ is simulating LRU, it follows that $L(k) = \overline{m(T)}$ and

$$L(k) \geq m(T_k). \tag{4.2}$$

In other words, when $m(T)$ is concave up in the range of T over which $T(t, k)$ varies, a working-set policy with $T = T_k$ will be at least as efficient as LRU.

5. Stochastic Analysis

It is useful to reconsider the analysis of the working-set model from a statistical standpoint. This will permit other analysis techniques not directly applicable in the time analysis and will lead to results not otherwise obtainable. We regard the program's reference string $r_1 r_2 \cdots r_t \cdots$ as a sequence of random variables. The basic transition probabilities of interest are[2]

$$g_{ij}(t, x) = \Pr[r_{t+x} = j \mid r_t = i], \quad i, j \text{ in } N,\ t \geq 1,\ x \geq 1. \tag{5.1}$$

In particular, the densities $g_{ii}(t, x)$ are nonstationary generalizations of the interreference densities $f_i(x)$. Under the assumptions that (5.1) is stationary within a locality (i.e. $g_{ii}(t, x) = f_i(x)$), that references are asymptotically uncorrelated, and that each page is recurrent (i.e. $\sum_{x \geq 1} g_{ii}(t, x) = 1$ for all $t > 0$), the results obtained from model (5.1) will converge to those obtained earlier. (Convergence will be in the sense discussed in connection with eq. (2.3).) Under these conditions, the interreference density functions f_i exist, $n = n_R$, and $\bar{x}_i = 1/\lambda_i$. It remains to show that P1–P9 follow.

Since the random process generating working-set size $w(t, T)$ is assumed stationary, we shall drop t and use $w(T)$ as the random variable "working-set size." Then $s(T) = \overline{w(T)}$ and $m(T) = \Pr$ [next ref. not in working set]. To show that P1–P9 follow, it is sufficient to show that

$$s(T) = \sum_{z=0}^{T-1} (1 - F(z)), \tag{5.2}$$

$$m(T) = 1 - F(T). \tag{5.3}$$

The remaining properties among P1–P9 are easily deduced from (5.2) and (5.3). To demonstrate (5.2), we define the binary random variables $\alpha_i(T)$ to be 1 if and only if page i is in the working set. We then may write $w(T) = \alpha_1(T) + \cdots + \alpha_n(T)$, whence

$$s(T) = \overline{w(T)} = \sum_{i=1}^{n} \overline{\alpha_i(T)}. \tag{5.4}$$

In the Appendix we show that

$$\overline{\alpha_i(T)} = \lambda_i \sum_{z=0}^{T-1} (1 - F_i(z)). \tag{5.5}$$

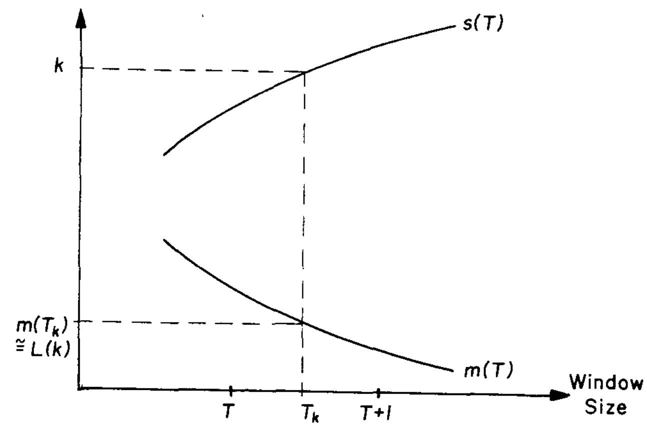
Substituting (5.5) for (5.4) and applying the definition $F(x) = \lambda_1 F_1(x) + \cdots + \lambda_n F_n(x)$, we obtain (5.2).

To demonstrate (5.3), we define the binary random variable $\Delta(T)$ to be 1 if and only if the next-referenced page is not in the working set; then $m(T) = \overline{\Delta(T)}$. From Figure 1, $w(T + 1) = w(T) + \Delta(T)$. Taking expectations on both sides, we find $m(T) = \overline{\Delta(T)} = s(T + 1) - s(T)$; from (5.2), $s(T + 1) - s(T) = 1 - F(T)$.

It is important to note that the random variables $\alpha_i(T)$

[2] To be precise, $r_1 r_2 \cdots r_t \cdots$ represents an ensemble of reference strings, and $g_{ij}(t, x)$ is defined across the ensemble at time t. Thus we can define nonstationary interreference densities, nonstationary mean interreference interval $\bar{x}_i(t) = \sum_x x f_i(t, x)$, page-reference probabilities $\lambda_i(t) = 1/\bar{x}_i(t)$, missing-page probabilite $m(t, T)$, and average working-set size $s(t, T)$.

Fig. 5



used above are dependent, but the linearity of expectation allows us to derive $s(T)$ without assumptions on the nature of this dependence. The same is not true for the variance of working-set size, given by

$$\sigma^2(T) = \overline{w^2(T)} - \overline{w(T)}^2, \qquad (5.6)$$
$$= \sum_{i=1}^{n}\sum_{j=1}^{n} \overline{\alpha_i(T)\alpha_j(T)} - s^2(T).$$

Since $\overline{\alpha_i(T)\alpha_j(T)} = \Pr[\alpha_i(T)\alpha_j(T) = 1]$, it is not possible to evaluate $\sigma^2(T)$ without assumptions about the distributions $g_{ij}(t, x)$ for all i and j in (5.1). The variance $\sigma^2(T)$ would be a complicated function of the g_{ij}. Because $s(1) = 1$ and $s(\infty) = n$, $\sigma^2(T)$ is constrained to be small when T is small and when T is large; therefore $\sigma^2(T)$ achieves a maximum for some $T > 1$. Since we are not prepared to offer assumptions about the g_{ij} here, we shall not pursue the problem of expressing $\sigma^2(T)$ further.

If the program size n is not too small and assumption W3 is satisfied, working-set size $w(T)$ will be normally distributed. Regarding $w(t, T)$ as a random process, we can write

$$w(t, T) = w(t-1, T) + \theta(t, T) \qquad (5.7)$$

where $\theta(t, T)$ assumes the values $+1$, 0, or -1 (the definition of working set implies that these are the only possible values). Therefore the working-set size at time t can be written

$$w(t, T) = \sum_{k=1}^{t} \theta(k, T). \qquad (5.8)$$

Now, the sequence $\theta(t, T)$ for $t = 1, 2, \ldots$ is a sequence of dependent random variables; but according to assumption W3, it is reasonable to assume that $\theta(t, T)$ and $\theta(t+x, T)$ become uncorrelated as $x \to \infty$. This in turn implies the conditions used in [13], from which we can conclude that the distribution of $w(t, T)$ as expressed in (5.8) converges to a normal distribution with mean $s(T)$ and standard deviation $\sigma(T)$ as given by (5.6). These conclusions are corroborated by experiments cited in [5].

6. The Independent Reference Model

According to the independent reference model [1], a program's reference string $r_1r_2 \cdots r_t \cdots$ is a sequence of independent random variables with stationary probabilities

$$\Pr[r_t = i] = \lambda_i, \quad 1 \le i \le n,\ t > 0. \qquad (6.1)$$

The reference string is "random" if $\lambda_i = 1/n$ for each i. Although it would be dangerous to use this model indiscriminately for practical programs, it does yield some additional insight into the nature of statistical dependence exhibited by practical programs.

It is possible to obtain expressions for the quantities $f_i(x)$, $F_i(x)$, $m(T)$, and $s(T)$ under the assumption (6.1). The interreference distributions are geometric:

$$1 - F_i(x) = \Pr[x_i > x] = (1-\lambda_i)^x, \qquad (6.2)$$

$$f_i(x) = F_i(x) - F_i(x-1) = \lambda_i(1-\lambda_i)^{x-1}, \qquad (6.3)$$

for which the mean is $\bar{x}_i = 1/\lambda_i$ and variance is $\lambda_i(1-\lambda_i)$. (To verify (6.2), note that $\Pr[x_i > x] = \Pr[r_{t+1} \ne i, \ldots, r_{t+x} \ne i]$.) The missing-page rate and average working-set size are, respectively,

$$m(T) = 1 - F(T) = \sum_{i=1}^{n} \lambda_i(1-\lambda_i)^T, \qquad (6.4)$$

$$s(T) = \sum_{z=0}^{T-1} (1 - F(z)) = n - \sum_{i=1}^{n} (1-\lambda_i)^T. \qquad (6.5)$$

An interesting aspect of the independent reference model concerns the constraints it imposes on $s(T)$. From property P1, we know that $s(T) \le \min\{n, T\}$. This bound can be achieved by the reference string $1, 2, \ldots, n, 1, 2, \ldots, n, \ldots$. Since programs which generate cyclic reference strings only cannot be modeled by independent references, this bound cannot be achieved within the independent reference assumption. Let $s(T \mid \lambda_1, \ldots, \lambda_n)$ denote the expression of (6.5). It can be shown that (6.5) is maximized when $\lambda_i = 1/n$ for all i,

$$s(T \mid \lambda_1, \ldots, \lambda_n) \le s(T \mid 1/n, \ldots, 1/n). \qquad (6.6)$$

That is, the expected working-set size is maximum for a purely random reference string.

Now, suppose $s^*(T)$ is an experimentally measured average working-set size curve, and $\lambda_i^*, \ldots, \lambda_n^*$ are the experimentally measured page reference rates. If

$$s^*(T) > s(T \mid 1/n, \ldots, 1/n), \qquad (6.7)$$

then in view of (6.6), the program cannot possibly be representable within the independent reference model. If (6.7) does not hold, then it would be fruitful to investigate whether

$$s^*(T) \cong s(T \mid \lambda_1^*, \dots, \lambda_n^*), \tag{6.8}$$

for if so, the independent reference model could be useful for investigating certain aspects of that program's behavior.

Still more can be deduced from comparisons between $s^*(T)$ and $s(T \mid \lambda_1^*, \dots, \lambda_n^*)$. If there exists a T_0 such that (6.7) holds for $T \leq T_0$ but not for $T > T_0$, we could conclude that the program includes a set of k pages, where $k \leq s^*(T_0)$, which contain a tight loop.

8. Conclusions

We have shown, in the context of the working-set model, that a large variety of results can be obtained with only minimal assumptions: reference substrings of interest are quasi-stationary; and references are asymptotically uncorrelated. Starting from this basis, we find that many extensions are possible. For example, reference strings can be decomposed into instruction and data references, and this decomposition can be used to decompose the working set into instruction and data working sets. Also, the effects of information sharing (overlapping working sets) can be studied with respect to memory demand and efficiency. Or interreference distributions known to exhibit clustering effects (e.g. the gamma distribution) can be used to obtain better approximations to locality within the framework of a stationary model.

Appendix

We wish to prove eq. (5.5). Since $\alpha_i(T)$ is 0 or 1, we need to determine $\Pr[\alpha_i(T) = 1]$. To simplify the discussion, we omit the use of subscript i and window size T.

Consider an interreference interval of length y; for convenience we suppose this interval spans the time interval $(0, y)$. Consider an arbitrarily chosen time point t in this interval, where $0 \leq t \leq y$. If $0 \leq t < T$ and $y > T$ or if $y < T$, then the back end of the window is at $t - T + 1 \leq 0$, and the reference to page i at time 0 is in the working set, i.e. $\alpha = 1$. It is therefore necessary to determine the probabilities that $y \leq T$, and $0 \leq t < T$ when $y > T$.

The distribution of intervals containing the arbitrarily chosen time point t is not the same as the interreference distribution $f(x)$, because the probability that t falls in an interval of given length is proportional to the fraction of the time axis occupied by intervals of that length. Let $g(y)$ denote the distribution of the interval $(0, y)$ containing t. Consider a long time interval defined by a large number k of successive interreference intervals. The expected number of these intervals which are of length y is $kf(y)$ and the expected total length occupied by intervals of length y is $kyf(y)$. Therefore the fraction of space occupied by intervals of length y is

$$g(y) = \frac{kyf(y)}{\sum_{x\geq 1} kxf(x)} = \frac{yf(y)}{\bar{x}} = \lambda y f(y) \tag{A.1}$$

The distribution function of y is

$$G(y) = \sum_{x=1}^{y} g(x) = \lambda \sum_{x=1}^{y} xf(x). \tag{A.2}$$

Using the fact that $f(x) = F(x) - F(x - 1)$, we can rearrange (A.2) so that

$$G(y) = \lambda y F(y) - \lambda \sum_{z=0}^{y-1} F(z). \tag{A.3}$$

From the remarks in the second paragraph above, we have $\alpha = 1$ if $y \leq T$ or if $0 \leq t < T$ and $y < T$. Therefore

$$\Pr[\alpha = 1] = \Pr[y \leq T] + \sum_{y>T} g(y) \Pr[0 \leq t < T < y]. \tag{A.4}$$

Given y, we may suppose t falls uniformly within $(0, y)$, whence $\Pr[0 \leq t < T < y]$ is T/y. Using this in eq. (A.4), we have

$$\begin{aligned} \Pr[\alpha = 1] &= G(T) + \sum_{y>T} \lambda y f(y) \frac{T}{y} \\ &= G(T) + \lambda T(1 - F(T)). \end{aligned} \tag{A.5}$$

Applying (A.3) to (A.5), we have

$$\begin{aligned} \Pr[\alpha = 1] &= \lambda T F(T) - \lambda \sum_{z=0}^{T-1} F(z) = \lambda T(1 - F(T)), \\ &= \lambda T - \lambda \sum_{z=0}^{T-1} F(z), \\ &= \lambda \sum_{z=0}^{T-1} (1 - F(z)), \end{aligned}$$

which was to be shown.

References

1. Aho, A., Denning, P., and Ullman, J. Principles of optimal page replacement. *J. ACM 18*, 1 (Jan. 1971), 80–93.
2. Belady, L. A study of replacement algorithms for virtual storage computers. *IBM Syst. J. 5*, 2 (1966), 78–101.
3. Brawn, B., and Gustavson, F. Program behavior in a paging environment. Proc. AFIPS 1968 FJCC Vol. 33, Pt. 2, AFIPS Press, Montvale, N.J., pp. 1019–1032.
4. Coffman, E.G. Jr., and Wood, R.C. Interarrival statistics for time sharing systems. *Comm. ACM 9*, 7 (July 1966), 500–503.
5. Coffman, E.G. Jr., and Ryan, T. A study of storage partitioning using a mathematical model of locality. *Comm. ACM 15*, 3 (Mar. 1972), 185–190.
6. Denning, P. The working set model for program behavior. *Comm. ACM 5*, 11 (May 1968), 323–333.
7. Denning, P. Resource allocation in multiprocess computer systems. MIT Project MAC Rep. MAC-TR-50 (May 1968), Ph.D. th.
8. Denning, P. Virtual memory. *Computing Surveys 2*, 3 (Sept. 1970), 153–189.
9. Doherty, W. Scheduling TSS/360 for responsiveness. Proc. AFIPS 1970 FJCC Vol. 37, AFIPS Press, Montvale, N.J., pp. 97–112.
10. Fuchs, E., and Jackson, P.E. Estimates of distributions of random variables for certain computer communications traffic models. *Comm. ACM 13*, 12 (Dec. 1970), 752–757.
11. Liptay, J. The cache. *IBM Syst. J. 7*, 1 (1968), 15–21.
12. Mattson, R., Gecsei, J., Slutz, D., and Traiger, I. Evaluation techniques for storage hierarchies. *IBM Syst. J. 9*, 2 (1970), 78–117.
13. Rosenblatt, M. A central limit theorem and a strong mixing condition. *Proc. Nat. Acad. Sci. 42* (1956), 43–47.
14. Sayre, D. Is automatic folding of programs efficient enough to displace manual? *Comm. ACM 12*, 12 (Dec. 1969), 656–660.
15. Yaglom, A.M. *Stationary Random Functions.* Prentice-Hall, Englewood Cliffs, N.J., 1962.
16. Weizer, N., and Oppenheimer, G. Virtual memory management in a paging environment. Proc. 1969 SJCC Vol. 34, AFIPS Press, Montvale, N.J., p. 234.