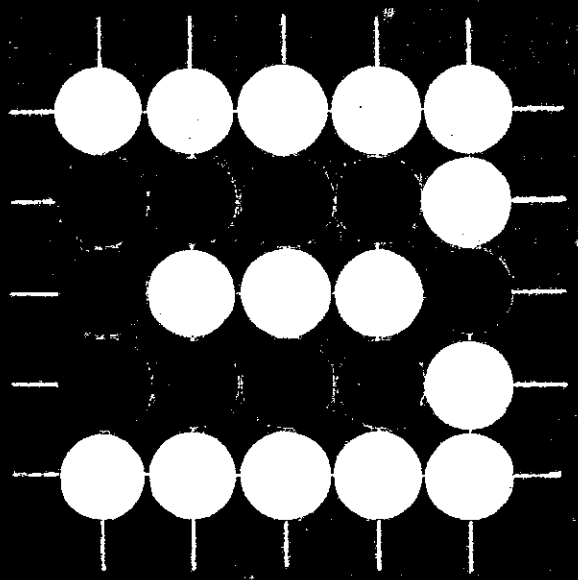
közlemények

18/1977



Felelős kiadó:
Dr. Vámos Tibor
igazgató

OPERÁCIÓS RENDSZEREK ELMÉLETE

Téli Iskola, Visegrád

MTA Számítástechnikai és Automatizálási Kutató Intézet
MTA Számítástudományi Bizottsága

Technikai szerkesztő:

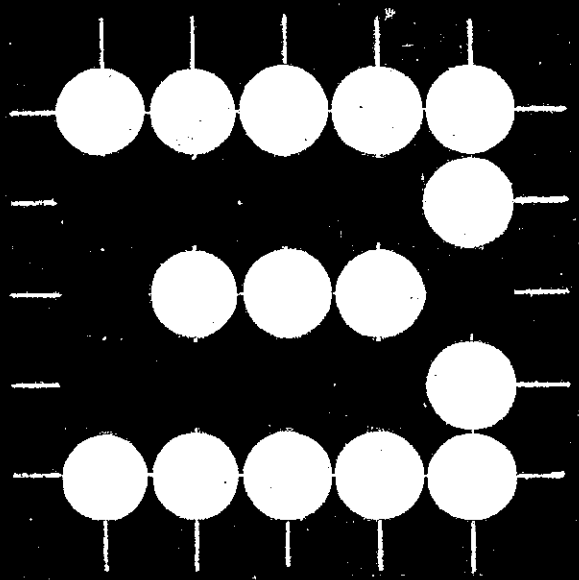
Solt Jánosné

MTA Számítástechnikai és Automatizálási Kutató Intézete

TARTALOMJEGYZÉK

közlemények

18/1977



MTA Számitástechnikai és Automatizálási Kutató Intézete, Közlemények, 18/1977.

PROGRAM BEHAVIOR IN MULTIPROGRAMMED COMPUTERS AND DIFFUSION APPROXIMATION

Mátyás Arató

One of the most important concept in analysis of computer program behavior is the *locality* property. In operating systems of computers the memory management problem – i.e., deciding how to distribute information among the levels and when to more it about – has of first importance. The virtual memory techniques for memory management are becoming widerspread. A virtual memory can be regarded as the main memory of a simulated computer. It is described in terms of the following abstractions: address space, memory space, and an address map. The address spece of a job is the set of addresses that can be generated by a processor as it executes the job. If the blocks, which are the units of allocation and transfer of information, are the same uniform size they are called *pages*. Most of the results about memory management are stated alwasy in terms of paging. A job's *reference string* $\eta_1, \eta_2, \ldots, \eta_t$, is defined such that η_t is the number of the page containing address x_t, $t \geq 1$. The reference string is used as the basis for the models of program behaviour. A reference string of pages satisfies the locality property if, during any interval of time, the program's pages are referenced nonuniformly, and the page reference densities change slowly in time.

One may define the *working set* of a program to be the smallest subset of its pages that must be in main memory at any given time in order to guarentee the program a specified level of processing efficiency. The *working set principle* of memory management asserts that a program (or job) may be designated as processable only if its working set is present in main memory. These definitions and practical measuring techniques were proposed by many authors (see e.g. Denning (1968) and Coffman, Denning's book (1974)).

Under the assumption of locality it is possible to demonstrate that the size of a job's working set tends to be normally distributed. This is a significant result, for it means that the theory of Gaussian processes can be applied to the study of memory management. This result is due to Denning and Schwartz (1972) and agrees with the experiments (see Coffman and Ryan (1972)).

When dealing with a large program it is impossible, in practice to predict the references deterministically, and it has been recognized that one has to resort to probabilistic models in this context. The choice of the correct probabilistic model is far from obvious. Here we want to give an improved model to understand the stochastic structure underlying the phenomene. Here we do not suggest methods for improvements in the decision algorithms (for such purposes we send the reader to Arató [1], [2] and Benczúr-Krámli-Pergel).

A program's working set $W(t,T)$ at time t is defined to be the set of distinct pages referenced in the time interval $[t - T + 1, t]$, i.e. among $\eta_{t-T+1}, \eta_{t-T+2}, \ldots \eta_t$. The parameter T is called the *window size,* it can be chosen large enough so that the probability of a currentlocality page's, being missing from the working set is small and, small enough so that the probability of more than one interlocality transition's being contained in the window is small.

A program's reference string $\eta_1, \eta_2, \ldots, \eta_t, \ldots$ we regard as a sequence of random variables. We assume that η_t is a simple Markov chain with stationary transition probabilities.

$$p_{ij}^{(n)} = P\{\eta_{t+n} = j \mid \eta_t = i\},$$

and

$$f_i(n) = \begin{cases} p_{ii}^{(1)}, & n = 1 \\ p_{ii}^{(n)} - \sum_{k=1}^{n-1} f_i(k)\, p_{ii}^{(n-k)}, & n > 1. \end{cases}$$

Further assumptions about reference string are that each page is recurrent, i.e.

$$\sum_{n \geqslant 1} f_i(n) = 1$$

and that η_t and η_{t+n} become uncorrelated in the limit $(n \to \infty)$.

The *working set size* $w(t,T)$ is the number of pages in $W(t,T)$. As

(1) $$w(t,T) = w(t-1,T) + \Theta(t,T)$$

where

$$\Theta(t,T) = \begin{cases} -1 \\ 0, \\ 1, \end{cases}$$

according to the assumptions on η_t it can be proved that $w(t,T)$ is asymptotically normally distributed as $t \to \infty$ and $T \to \infty$. The proof of this statement depends on the "mixing" property of homogeneous stochastic processes (see e.g. M. Rosenblatt, J. Rozanov or I. Ibragimov- J. Linnik).

In most applications it is more easy and simpler to deal with the sequence $w_n = w(t_n,T)$, where $t_0 < t_1 < 0 \ldots < t_n < t_{n+1} < \ldots$ and $t_n - t_{n-1} \gg 1$ (large enough).
In this case we may use the normal approximation for w_n and heuristically we get (with $Ew = m_n$, $w_n' = w_n - m_n$)

(2) $$w'_n = \rho w'_{n-1} + \epsilon_n, \qquad |\rho| < 1,$$

with independent sequence ϵ_n. The matter is that time can be measured in at least two ways: either with respect to the flow of instructions, or with respect to new page references. Because the first method is more informative when studying overhead caused by page faults, it is adopted in most cases. The second method is sufficiently information in almost all other cases, e.g. for dynamic partitioning strategy (see Coffman and Ryan (1972)).

Relation (2) means that the probability that w'_n increases at the next step is inversely proportional to w'_n; the process has a tendency to approach the mean that is a function of n.

When we are using the model (1) we cannot assume that $w(t)$ (T is fixed) is a stationary process, even when $\Theta(t,T)$ are not independent. But using the scaling factor $t_n - t_{n-1} = s_n$ and taking

$$w_{t_n} = w_n$$

and then the approximation that n is continuous it may be assumed that w_n is stationary. This last assumption means that we are taking a rared sequence of $w(t,T)$ and then (as $n \longrightarrow \infty$ too) we are looking the process w_n on a new axis, see Fig.1.

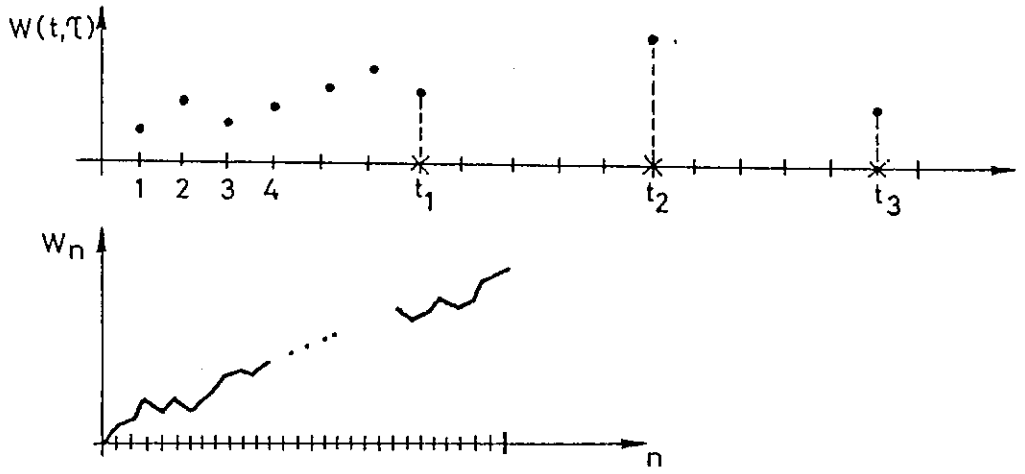


Figure 1.

The first order autoregressive model was used by Coffman and Ryan, where they tested the normal approximation with Monte-Carlo method (simulation). Empirical results also suggested the normal approximation for large "window" sizes, T and larger $t_n - t_{n-1}$. For smaller values of theses parameters the distribution is more closed by Poisson distribution.

When asymptotic uncorrelation assumption is not met for $\Theta(t,T)$ (as $t \to \infty$) the normal approximation to $w(t,T)$ may be poore (see e.g. Rodriguez–Rosell (1971)). Pages are assumed to be of constant size and the entire storega hierachy, main memory and auxiliary storage, in which paging takes place is taken to contain of $\{1,2,\ldots,n\}$ pages, where n is the total number of pages. Let k be the maximum number of pages that can reside in main memory. Under *demand paging*, a page that is referenced but is not resident in main memory will be brought in from auxiliary storage, usually in place of some other page which has to be pushed out, according to the particular paging algorithm implemented in the oparating system.

A common class of such algorithms is the so-called stack algorithm, which we define in the following way (see Denning [2]). The set $D = \{1,2,\ldots,n\}$ denotes the set of pages of a given program, so that the members of the reference string $\eta_t \epsilon D$. The program has been allocated a main memory space of m pages $m \leqslant n$ and $m \leqslant k$. We call subset S of D such that S contains m or fewer pages a possible *memory state*. Let $S(A,k,r)$ denote the memory state after A(the allocation algorithm) has processed r pages under demand paging in an initially empty main memory of size k. The A is a stack algorithm if $S(A,k-1,r)$ is a subset of $S(A,k,r)$. That is the contents of the $(k-1)$ page memory are always contained in the k-page memory, so that the memory states are "stacked up" on one another. The most popular algorithm is LRU, least recently used, because in some Markov models it is optimal. For stack algorithms the performance is crucially dependent on the behavior of the *distance string*, defined as follows. Suppose that at time t page $\eta_t = i$ has been referenced, and that the next reference to page i occurs at time $t + n_t + 1$; that is, between these two references to page i there have been n_t page references, but none to page i. Let d_t denote the number of distinct page references among these $\eta_t$references. The string $(d_1, d_2, \ldots)$ is called the *distance string* corresponding to the reference string η_t. A page exception will occur each time that $d_t = k$. If we assume that (see Freiberger, Grenander, Sampson 1975) at time t a page i is selected from $\{1,2,\ldots,n\}$ according to the probability distribution $p = \{p_1,\ldots,p_n\}$ and then subsequent references are made to this page i a number $\nu_i - 1$ times, where

$$P(\nu_i = k) = (1 - q_i)\, q_i^{k-1}, \quad k = 1,2,\ldots \quad (0 \leqslant q_i \leqslant 1),$$

then the lenght of the reference string (n_t) is not normally distributed. The distribution of the distence d_t under the above conditions, with $p_i = 1/n$ and $q_i = 0$, has the following surprisingly simple form

$$P(d_t = k) = \frac{1}{n},$$

which is far from Gaussian distribution.

Empirical results (see Freiberger, Grenander and Sampson) show that stochastic dependence between some pages must exist contradicting of the above assumptions in the analytic model. The empirical autocorrealation coefficient of order one indicates that the time series of working set size cannot be a white noise, it can be regarded as a first order autoregressive series with coefficient $\rho = 0{,}2 - 0{,}3$.

Spectral tests were used (see Lewis, Shedler 1973) to show the distance strings are correlated. That the d_t'–s are near a first order Markov chain (see Table 1.) they get empirically

Counts of one step transitions for LRU distances

j/k	1	2	3	4	5	6	7
1	2,943.817	840.912	210.914	78.002	41.500	24.281	16.792
2	1,048.310	2,151.371	146.163	35.192	26.012	13.591	7.931
3	130.957	271.850	176.386	16.570	5.693	2.804	2.318
4	22.878	70.630	35.013	10.338	3.721	1.516	1.643
5	18.512	36.744	16.366	5.258	4.017	393	235
6	10.685	20.180	7.959	1.650	812	1.914	223
7	11.549	11.324	3.596	846	280	233	271
8	7.957	9.934	4.405	1.261	182	128	61
9	9.451	8.248	3.457	834	184	145	56
10	7.274	8.657	2.641	769	639	268	76

Table 1.

From the sequence of address traced, the distance string was derived by stack processing techniques for a page size of 4K (4096) bytes. The data consisted of $t_0 = 8{,}802.464$ references to a total of 517 distinct pages.

For the exception process the results are in Table 2.

	Memory capacity c (pages)		
	76	197	512
N	1807	820	517
$\hat{\rho}_1$ – estimated serial correlation coefficient of order 1 for times between page exceptions	+ 0.188 (0.08)	+ 0.177	+ 0.130
$(N-1)^{\frac{1}{2}} \hat{\rho}_1$	+ 8.01 (1.7)	+ 5.11	+ 3.00
$\hat{\alpha}_{22}$ – estimated partial serial correlation of order 2	0.035 (0.002)	0.065	0.002

Table 2.

From this table 2. one can see that the estimated serial correlation of lag 1, $\hat{\rho}_1$, is too large to be consistent with a true value $\rho_1 = 0$ (the asymptotic variance is $\frac{1}{(N-1)^{1/2}}$, where N is the number of observation). Another outstanding feature is that the estimated partial serial correlations, $\hat{\alpha}_{22}$, are very small, suggesting first-order Markov dependence of intervals between exceptions. The partial correlation of order two is (see Arató, Benczúr, Krámli, Pergel (1976))

$$\alpha_{22} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2}$$

In case of first-order Markov dependence $\rho_k = \rho^k$ and $\alpha_{22} = 0$.

At last we give the empirical correlation function of the memory state at the second level on a CDC 3300 computer, the so-called scratch-pool requirement. The scratch-pool requirements of jobs depend on time stochastically and the benchmark measurements were executed during runtime (see Tőke, Tóth, (1975)). The samples are taken in every minutes for 5 hours. The empirical correlation function has an $e^{-\lambda t}$ (t – time) form, which indicates the first order autoregressive scheme, where after 15-20 minutes the correlation is practically equal to 0 (see Fig. 2.).

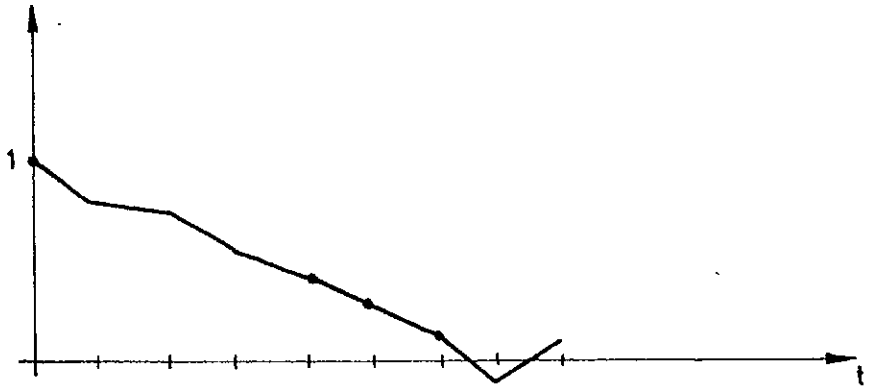


Figure 2.

References

[1] M. Arató, A. Benczúr, A. Krámli, J. Pergel, Statistical problems of the elementary Gaussian processes (1976.) SzTAKI Tanulmányok (1976) 41.

[2] M. Arató, [1] A note on optimal performance of page storage hierarchies Acta Cybernetica (in print) (1976)

[3] M. Arató, [2] Számitógépek hierarchikus laptárolási eljárásainak optimalizálásáról (in Hungarian) (1976) SzTAKI Közlemények 16

[4] A, Benczúr, A. Krámli, J. Pergel, On optimal algorithms of demand paging (1976). Acta Cybernetica (in print)

[5] E. Coffman, P. Denning, Operating Systems theory Prentice-Hall, New Yersey (1973)

[6] E. Coffman, T. Ryan, A study of storage partitioning using a mathematical model of locality (1972) Comm. A.C.M., 185-190.

[7] P. Denning [1] (1968) Resource allocation in multiprocess computer systems. MIT Project MAC report MAC-TR-50 Cambridge, Mass.

[8] P. Denning, [2] (1970) Virtual Memory Computing Surveys 2, 3 (1970).

[9] P. Denning, Schwartz, Properties of the working set model Comm. A.C.M. 191-198 (1972).

[10] W. Freiberger, U. Grenander, P. Sampson, Patterns in page references (1975) IBM J. Res. Dev. 230-243.

[11] I. Ibragimov, Ju. Limik, Independent and stationary related random variables (in Russian) Nauka, Moskwa.

[12] P. Lewis, G. Shedler, Empirically derived micromodels for sequences of page exceptions (1973). IBM J. Res. Dev.

[13] Rodriguez-J. Rosell, Experimental data on how program behaviour affects the choice of scheduler parameters, Proc. 3 rd ACM Symp. on Op. Syst. Princ. (1971)

[14] M. Rosenblatt, A central limit theorem and a stong mixing condition. Proc. Nat. Acad. Sci. USA 42, 43-47. (1956).

[15] Ju. Rozanov, Stationary stochastic processes (in Russian) Fizmatgis, Moskwa, (1963).

[16] P. Tőke, B. Tóth, A scratch-pool utilization study (in Hungarian) SzTAKI Közlemények 15, 103-110. (1975).

Összefoglaló

Multi programozású számitógépek program viselkedése és diffuziós közelités

Arató Mátyás

A dolgozatban a hivatkozási string sztochasztikus viselkedése alapján vizsgáljuk az u. n. munkamezők nagyságának Gauss eloszlással történő közelitését. Empirikus adatok alapján az időbeli változás egyszerü autoregressziós sémával irható le, amely diszkrét diffúzios folyamat.

Р Е З Ю М Е

Поведение программ в ЭВМ с многопрограмным режимом и диффузионные приближения.

Матяш Арато

В статье рассматриваются вопросы приближения с нормальным распределением так называемых рабочих полей когда поведение " reference string " является случайным. На основе эмпирических данных можно утверждать, что схема авторегрессии является хорошим приближением для описания процесса рабочего поля с дисвретным временем.