INFORMATION PROCESSING LETTERS 1 (1971) 59-60. NORTH-HOLLAND PUBLISHING COMPANY

THE TWO-THIRDS RULE FOR DYNAMIC STORAGE ALLOCATION UNDER EQUILIBRIUM

Erol GELENBE*

Computer, Information and Control Engineering, University of Michigan, Ann Arbor, Michigan, USA

Received 25 February 1971

dynamic storage allocation   fifty percent rule   equilibrium behavior

1. The two-thirds rule for dynamic storage allocation under equilibrium

In this note we consider a dynamically allocated storage system for variable-size records, as described by Knuth [1] (Section 2.5). We assume that storage is allocated according to a "best-fit" strategy. Free memory blocks are kept in the linked list AVAIL while USE is the set of reserved blocks. Let N and M be the average number of blocks in USE and AVAIL, respectively. As in [1] we assume that the system is in "equilibrium", so that N and M remain constant. Let $(1-p)$ be the probability that a request for a block of size x is satisfied exactly by one of the blocks in AVAIL. Knuth shows that under equilibrium implying that all requests of storage are satisfied (i.e. no overflow occurs),

$$M = \tfrac{1}{2}pN \tag{1}$$

and reports a simulation experiment verifying this relation when $p \approx 1$. Since usually $p \approx 1$, (1) is known as the "fifty-percent rule" [1,2].

The purpose of this communication is to derive a relationship between f, the average size of the blocks in AVAIL, and r the average block size in USE. With this information one can then determine the fraction of storage space which is reserved when the system operates under equilibrium

$$\rho = \frac{rN}{fM + rN} \tag{2}$$

(i.e., N, M are constant). It will be shown that for $p = 1$

$$\rho = \tfrac{2}{3}\,, \tag{3}$$

indicating that operation under equilibrium is inefficient in memory utilization. This value of ρ is supported by Knuth's simulation experiment [1] since he reports that a memory overflow occurred (equilibrium was destroyed) if the reserved portion of the total memory capacity exceeded $\frac{2}{3}$.

The blocks in USE will be of three types, according to their location in memory with respect to the blocks in AVAIL, assuming that the memory locations are numbered consecutively $0, 1, ..., K-1$ where K is the capacity of the memory. Each block of the first type, α, will have a free block at its "right" and "left"**. Each block of type β will have a free block either at its right or left, but not both. Blocks of type γ will have a block from USE at their right and left. There are A, B, C blocks of type α, β, γ respectively in USE. Obviously [1]

$$N = A + B + C$$

* Systems Engineering Laboratory, 2523 East Engineering. Ann Arbor, Michigan 48104.

** A block of memory is a set of consecutively numbered memory locations. If n is the largest location number a block occupies, then the block at its right begins at location $n+1$. The block at its left is defined analogously.

and approximately (neglecting blocks at "edges" of the memory)

$$M = \tfrac{1}{2}(2A+B)\,. \tag{4}$$

Let an *event* be an allocation of a block of free storage to a record. Since the system is under equilibrium, we may assume that on the average only one block in USE enters AVAIL (becomes "free") in the time between two consecutive events. If f is the average (over AVAIL) of free block sizes, after a block in USE enters AVAIL, f changes to

$$f' = \left[\frac{(M-2)f+2f+r}{M-1}\right]\frac{A}{N} + \left[\frac{(M-1)f+f+r}{M}\right]\frac{B}{N} + \left[\frac{Mf+r}{M+1}\right]\frac{C}{N} \tag{5}$$

since the newly freed block could have originated from α, β, γ, with probability (A/N), (B/N), (C/N) respectively*, and because if it is of type α the number of blocks in AVAIL will have decreased M by one, while if it was of type γ it would have increased M by one. A newly freed block of type β would leave M unchanged. Therefore

$$f' = f + r\left[\frac{A}{N(M-1)} + \frac{B}{NM} + \frac{C}{N(M+1)}\right] + f\left[\frac{A}{N(M-1)} - \frac{C}{N(M+1)}\right]$$

and approximately (for large M)

$$f' \approx f + \frac{r}{M} + \frac{(A-C)}{MN} f\,. \tag{6}$$

Knuth [1] shows that under equilibrium

$$C - A \approx (1-p)N\,, \tag{7}$$

therefore

$$f' = f + \frac{r}{M} - \frac{(1-p)}{M} f\,. \tag{8}$$

Immediately after an event, however, f' is changed to f'' as follows.

$$f'' = \left[\frac{(M-1)f'+(f'-r)}{M}\right]p + f'(1-p)\,. \tag{9}$$

This is because if a perfect fit [with probability $(1-p)$] is obtained M decreases by one and the average block in AVAIL remains of the same size, while if the fit is not perfect M remains unchanged and a new block of size $(f'-r)$ is added to AVAIL. Therefore

$$f'' = f' - \frac{rp}{M} = f + \frac{r}{M} - \frac{(1-p)}{M} f - \frac{rp}{M}\,.$$

But if f remains unchanged (in equilibrium) after each event we must have

$$\frac{r}{M} - \frac{(1-p)}{M} f - \frac{rp}{M} = 0\,,$$

hence

$$r = f\,, \tag{10}$$

independently of p. Thus from (2) we obtain

$$\rho = \frac{1}{1+\frac{1}{2}p}\,, \tag{11}$$

yielding (3) when $p \approx 1$.

* The block type and its lifetime are independent of each other.

References

[1] D.E.Knuth, The Art of Computer Programming, Volume 1, Fundamental Algorithms, Addison-Wesley, Reading, Mass. 1968.

[2] P.J.Denning, Virtual Memory, Computing Surveys, Vol. 2, No. 3, September 1970, pp. 161–182.