Analysis of Disc Fragmentation using Markov Chains

C. H. C. Leung
Department of Computer Science, University of Reading, Whiteknights Park, Reading RG6 2AX, UK

The fragmented storage map is represented by a Markov model which incorporates correlation between locations not necessarily adjacent to one another. Formulae for the average access distance for contiguous and non-contiguous allocations are given. For large storage requirements, the performance penalty of the former can be substantially higher than that of the latter: they are shown to be $O(k^2)$ and $O(n)$ respectively, where $n$ is the storage requirement and $k > 1$. The Markov model is also able to achieve close agreement with published measurements.

1. INTRODUCTION

The problem of disc storage fragmentation is studied in Ref. 1, where the fragmented storage map is represented as a binary stochastic sequence $\{C_n\}_{n \geq 0}$ with $C_n$ representing the condition—either free or occupied—of the $k$th location measured from an appropriate starting point. The sequence $\{C_n\}$ in Ref. 1 is taken to be statistically independent, and empirical evidence suggests that, under severe fragmentation, this appears to be acceptable as a first approximation. The present paper is a sequel to Ref. 1 and generalizes on the findings there. Both the results and method of derivation are new and the present model subsumes that in Ref. 1 as a special case. In particular, the assumption of independence for the sequence $\{C_n\}$ is abandoned, for unless the disc storage is very severely fragmented, the resultant approximation may not be adequate. In this study, we allow Markov dependency among different locations, i.e. the statistical characteristics of the condition of subsequent locations can be determined once that of the present one is known. Although the present model is not sufficiently general to describe memory maps with arbitrary statistical characteristics, it offers realistic improvement over the former model and should considerably extend the scope of applicability of the results to cover not only severe fragmentation conditions, but also less extreme conditions. Under moderate to severe fragmentation, correlation between locations separated by more than a few positions is mostly 'washed out' by the random action of unstructured fragmentation so that the Markov model—with a geometrically decaying autocorrelation function (see Section 2)—should provide a good approximation. In fact, by approximating the disc in Ref. 1 by a Markov model, it is possible to obtain much closer agreement between actual measurements and model results.

2. MODEL FORMULATION

If we use $0$ to signify that a given location is free, and $1$ to signify that it is occupied, our model incorporates dependency of the form, for any $k \geq 0$,

$$P(C_{k+1} = 0|C_k = 0) = \alpha, \quad 0 < \alpha < 1$$

and

$$P(C_{k+1} = 1|C_k = 1) = \alpha', \quad 0 < \alpha' < 1$$

so that we have the Markov transition matrix

$$P = \begin{pmatrix} \alpha & \beta \\ \beta' & \alpha' \end{pmatrix}$$

with $\alpha + \beta = \alpha' + \beta' = 1$. The probability $\alpha$ measures the ‘affinity’ between free locations, i.e. their tendency of occurring together which, as we shall see, has an important bearing on storage allocation performance. The correlation between locations $\tau$ positions apart for this model can be shown to be (Ref. 2, p. 278).

$$\rho(\tau) = \frac{\text{Cov}(C_0, C_\tau)}{\text{Var}(C_0)} = (1 - \beta - \beta')^\tau, \quad \tau \geq 0$$

which indicates that correlation is present between locations not adjacent to one another, but it decays geometrically with respect to their separation.† The previous model corresponds to the special case $\alpha = \beta$ so that the autocorrelation function degenerates to

$$\rho(\tau) = \begin{cases} 1, & \tau = 0 \\ 0, & \tau > 0 \end{cases}$$

which permits no correlation between any two distinct locations, adjacent or otherwise.

As in Ref 1, we distinguish between two main types of storage operations: non-contiguous and contiguous allocations. In the former, a given storage request does not need to be met by a single uninterrupted sequence of free locations whereas in the latter, the allocation of storage must be made in a single continuous block. The latter scheme is more restrictive but is required for certain file structures. We shall first examine non-contiguous storage allocation.

We denote by $v_n$ the average access distance that the read/write head has to cover to accumulate $n \geq 0$ further free locations given that we start from an occupied one. Similarly, we let $v_n$ signify the corresponding access distance given that we start from a free location. In either situation, there are two possibilities: if the next location encountered is free, then the average residual access distance is $v_{n-1}$; otherwise, the average residual access

†Indeed, autocorrelation function of this form is reported in Ref. 3, Section 13 to be very common and is able to provide close approximation to many stationary series encountered in practice.
distance is \( v_n \). Hence combining both possibilities, we have the following simultaneous equations for \( n > 0 \):

\[
v_n = 1 + \alpha v_{n-1} + \beta v_n
\]
\[
v_n = 1 + \beta v_{n-1} + \alpha v_n
\]

(1)

(2)

Substituting (2) into (1) gives

\[v_n = v_{n-1} + (1 + \beta / \beta), \quad n > 0\]

But \( v_0 = 0 \); this therefore yields

\[v_n = n(1 + \beta / \beta)\]

which also implies

\[v_n = [(n - 1)(\beta + \beta) + 1] / \beta\]

It is shown in Ref. 2, p. 80, that the equilibrium probability vector associated with the Markov chain \( P \) is \( (\omega = \beta / (\beta + \beta), \omega = \beta / (\beta + \beta)) \). In the present situation, \( \omega \) can be interpreted as the probability that a chosen location is free, and \( \omega' \), that it is occupied. Since \( \omega \) gives the likelihood that a given location is free, it will for convenience be referred to as the availability of the disc. Hence, starting from an arbitrary point on the disc with no information concerning the condition of the current location, the allocation distance the read/write head is expected to cover in order to accumulate \( n \) units of free space, not necessarily in a contiguous manner, is the unconditional average

\[d_n = \omega v_n + \omega' v_n' = [n \beta + (\omega')^2 - \beta \omega'] / (\omega \beta)\]

The average number of fragments over which the file is scattered is sometimes of interest as frequently there is a limit to it; for example, in the IBM System 370, this limit is 16. If \( f_n \) denotes the average number of fragments resulting from allocating a file of size \( n \), then for \( n > 0 \), \( f_n \) would be the same as \( f_{n-1} \), if the location immediately following the \((n - 1)\)th location allocated to the file is free so that there is no interruption between the \((n - 1)\)th location and the \(n\)th location—this happens with probability \( \omega \); otherwise one additional fragment to house the \(n\)th unit is required. Thus we have

\[f_n = \alpha f_{n-1} + \beta (f_{n-1} + 1)\]

Since \( f_1 = 1 \), we obtain

\[f_n = 1 + (n - 1) \beta, \quad n > 0\]

We next turn our attention to contiguous allocation. Here, we shall replace \( v_n \) and \( v_n' \) by \( s_n \) and \( s_n' \), respectively. The difference between the \( v \)'s and the \( s \)'s being that, in the latter, the additional restriction of single fragment allocation must be observed; accordingly \( v_n \leq s_n \) and \( v_n' \leq s_n' \). Now, \( s_n \) would simply be \( s_{n-1} + 1 \) if the location immediately following the last location otherwise allocated to a file of size \((n - 1)\) is free. However, if this is not so, the whole allocation process will need to start from scratch again except that an access distance of \((s_{n-1} + 1)\) units has already been incurred. Hence, we have for \( n \geq 2 \)

\[s_n = \alpha (s_{n-1} + 1) + \beta (s_{n-1} + 1 + s_n')\]

or

\[\alpha s_n' = s_{n-1} + 1, \quad n \geq 2\]

(3)

Likewise, the equation for \( s_n \) is

\[s_n = \alpha (s_{n-1} + 1) + \beta (s_{n-1} + 1 + s_n')\]

or

\[s_n = \beta s_n' + s_{n-1} + 1, \quad n \geq 2\]

(4)

Now, (3) is a first-order difference equation and from Ref. 5, p. 37, its solution is given by

\[s_n' = \alpha (\omega' - 1) / (\beta (\omega' - 1)), \quad n \geq 2\]

(5)

where \( C \) is an arbitrary constant depending on the initial condition of the equation. We know that \( s_1 \) is the amount of access distance required to allocate the first unit of space, and as we are dealing with single unit allocation, there is no difference between the contiguous and non-contiguous schemes and so we have \( s_1 = v_1 = 1 / \beta \). Substituting this into (3), we have

\[s_2 = 1 / (\alpha \beta) + 1 / \alpha\]

Substituting this last expression into (5) yields

\[1 / (\alpha \beta) + 1 / \alpha = C / \alpha + 1 / \alpha\]

giving \( C = 1 / \beta \) and the complete solution of \( s_n' \) is, on simplification,

\[s_n' = (\beta + \beta' - \beta' \alpha' - 1) / (\beta \beta' \alpha' - 1), \quad n > 0\]

Next, the solution of (4) can be seen to be (by repeated substitution),

\[s_n = s_1 + \sum_{k=2}^{n} (1 + \beta s_k)\]

and again, in establishing the correspondence \( s_1 = v_1 \), we obtain, after simplification,

\[s_n = (\beta + \beta') (1 - \alpha' / (\beta \beta' \alpha' - 1)), \quad n > 0\]

Figure 1. Mean allocation distance under fragmentation: —— contiguous allocation; —— non-contiguous allocation; \( \omega \): storage availability.
Hence, the distance \( d^n \) the read/write head is expected to cover in accumulating \( n \) contiguous units of free space commencing from an arbitrary point is

\[
d^n = \omega s_n + \omega' s_n
\]

which, on simplification, yields

\[
d^n = (\alpha^{-(n-1)} - \alpha \omega - \omega \omega')(\omega \beta)
\]

If \( n \gg 1 \), the first term in the above expression is the dominant one and we have \( d^n = O(\alpha^{-n}) \). If there is strong 'affinity' among the free locations, i.e. \( \alpha \approx 1 \), then \( \alpha^{-1} \approx 1 \) so that \( d^n \) grows with respect to the storage requirement \( n \); on the other hand, a rapid growth would result if the 'affinity' among free locations is weak. In the case of non-contiguous allocation, of course, we have \( d^n = O(n) \). In fact, it can be formally demonstrated (see Appendix 1) that for \( n > 0 \),

\[
d^n \geq d_n
\]

with equality holds if and only if \( n = 1 \). Figure 1 plots the mean access distance against the storage requirement for \( \alpha = 80 \% \). We see that the access distance required for contiguous allocation is substantially higher than that for non-contiguous allocation, and that they diverge as the storage requirement increases. We also observe that the storage availability has a strong bearing on allocation performance: the access distance under different storage availability shows considerable difference for the same storage requirement.

3. PRACTICAL CONSIDERATIONS

Since the model in Ref. 1 is a special case of the present model, the latter's accuracy when compared with empirical measurements is not lower than that of the former. Using the measurements published in Ref. 1, errors of below 0.1\% are observed for \( r_n \) and \( f_n \). The details of the comparison are given in Appendix 2. In a practical sense, the present model is also more versatile than that in Ref. 1 because it has two independent parameters \( \beta \) and \( \beta' \); the model in Ref. 1, however, merely has a single parameter \( q \). The parameters \( \beta \) and \( \beta' \) may be estimated as follows. The distribution of free space length \( X \) is, counting from the beginning of the block, \( P[X = n] = \alpha^{n-1} \beta \), which implies \( E(X) = 1/\beta \). Likewise, if \( Y \) denotes the occupied space length, then we have \( E(Y) = 1/\beta \). Hence, \( \beta \) and \( \beta' \) may be estimated by \( \beta = 1/E(X) \) and \( \beta' = 1/E(Y) \), respectively. This estimation procedure is illustrated in Appendix 2.

4. SUMMARY AND CONCLUSIONS

A Markov model is employed to study the performance of fragmented disc storage. It represents a realistic improvement over the model in Ref. 1 and allows correlation between different storage locations. It should provide adequate approximation for moderate to severe fragmentation conditions. Expressions have been derived for the average access distance incurred for allocating contiguous and non-contiguous space: the former is found to be an exponential function of the storage requirement, whereas the latter is a linear function. Furthermore, the Markov model is able to fit empirical measurements significantly better than the previous model which, coupled with its simplicity, should provide a useful tool for the practical assessment of system performance.

Acknowledgements

The author is grateful to the Data Administration and Hardware Evaluation & Capacity Planning Departments of Kodak Limited for drawing his attention to the problem, and the present solution owes much to discussions with Mr A. Price of the latter department.

REFERENCES


Received February 1982

APPENDIX 1

To show that \( d^n \geq d_n \) for \( n > 0 \) with equality holding if and only if \( n = 1 \).

From the expressions for \( d^n \) and \( d_n \), it is evident that \( d^n \geq d_n \) iff

\[
\alpha^{-(n-1)} - \alpha \omega - \omega \omega' \geq n \beta + (\omega)^2 - \beta \omega
\]

which is equivalent to

\[
\alpha^{-(n-1)} \geq n(1-\alpha) + \alpha
\]

Since \( \alpha < 1 \), we can write \( \alpha^{-1} = 1 + x \), with \( x > 0 \), and so the above becomes

\[
(1 + x)^n \geq 1 + nx
\]

which is clearly true since the right hand side is merely the first two terms in the binomial expansion of the left. If \( n = 1 \), then evidently we have equality. If \( n > 1 \), then since we have discarded at least one positive term in the binomial expansion, we thus have strict inequality.
APPENDIX 2

Empirical assessment

Using the measurements in Ref. 1, we have from Section 3 the estimates \( \hat{\beta} = 1/E(X) = 1/1.30 \) and \( \hat{\beta}' = 1/E(Y) = 1/69.29 \). Since the mean file size is \( n = 9 \), therefore having allocated the first free track to the file, the computed mean allocation distance starting from a free location is \( v_{n-1} = 8(1 + 69.29/1.30) = 434.4 \). The corresponding measured value is \( (160 + 8523)/20 = 434.15 \), which gives an error of 0.06%. Next, the computed average number of fragments per file is \( f_0 = 1 + 8/1.3 = 7.15 \). The corresponding measured value is 143/20 = 7.15, which exhibits complete agreement with the computed value.