BIT 28 (1988), 27-36

ESTIMATING DISK HEAD MOVEMENT IN BATCHED SEARCHING

Y. P. MANOLOPOULOS and J. G. KOLLIAS

Division of Computer and Electronics Engineering
Department of Electrical Engineering,
University of Thessaloniki,
54006 Thessaloniki, Greece

Division of Computer Science,
Department of Electrical Engineering,
National Technical University of Athens,
15773 Athens, Greece

Abstract.

The study considers the problem of evaluating the expected disk head movement when the SCAN disk scheduling policy is used to answer a batch of queries. The two cases examined are: (a) the batched queries are based on primary key values, and (b) each query in the batch is based on secondary key values.

Earlier works assumed that hit cylinders are non-distinct and derived an exact (approximate) formula for the first (second) case. In this paper, both replacement and non-replacement models are examined and new exact (exact and approximate) formulae are derived for the first (second) case. It is shown that earlier and new approximate results may be used instead of the computationally expensive exact formulae.

Categories and Subject Descriptors: D.4.2, H.2.4.

General Terms: Disks, Algorithms, Performance.

Additional Keywords and Phrases: Disk head movement, SCAN scheduling algorithm, Batched search, Primary and secondary key retrieval, Replacement and non-replacement model.

1. Introduction.

A number of disk scheduling policies have been suggested in the past for satisfying requests for information which resides on a disk with movable heads. Among them we note the shortest-seek-time-first (SSTF), the SCAN access policy, the N-step scan and the Eschenbach scheme, as improvements over the first-come-first-served (FCFS) policy.

Scheduling policies have been broadly evaluated [1, 4, 11–14]. In [1, 4] it is stated that the cost of satisfying a batch of queries is a function of the total distance traveled by the disk heads and the number of the cylinder hits. Since the first factor may contribute much more than the second to the total cost,

Received September 1986. Revised December 1987.

we concentrate on its evaluation for the following two problems. First, we consider the case that a set of queries based on primary key values are to be answered from the file. For example, suppose that a set of transactions based on account numbers is to be processed using an account file. The required disk head movement may be drastically reduced if the transactions are considered as a batch (instead of performing them on a FCFS basis). The same case arises when we have an individual query based on a secondary key value. For example, the request may ask for all the employees holding a Ph.D. degree. A secondary index on the values of the attribute DEGREE normally provides the set of primary key values which hold each particular degree (e.g. B.Sc., M.Sc., Ph.D. etc. [12]). Second, we consider the case that a batch of queries based on secondary key values are to be answered. This is a more general case because each query in the batch forms another batch of records.

For the rest of this study we make exactly the same assumptions made in [1, 4]. In particular, we assume

(a) that the file occupies N consecutive cylinders,
(b) that the physical addresses on the records which satisfy a batch are sorted on ascending cylinder number and that the cost of sorting is negligible. A direct consequence of this assumption is that batching may well apply to any primary file structure,
(c) that the SCAN scheduling policy is applied to answer the requests. We note that according to SCAN the disk head travels alternatively from the outer to the inner cylinder,
(d) that the query processing program is dedicated to the satisfaction of the queries to the file. This excludes the possibility of changing the direction in which the head moves in order to serve some other system request, and
(e) that the disk head is initially positioned over the first cylinder of the file and is ready to move towards the inner cylinders of the disk.

We call first (last) record of a query the one residing closer to the outer (inner) cylinder. After processing all the queries there is no need for the head to return to the first cylinder.

In both these earlier works [1, 4] a replacement model was implicitly assumed and estimates for the expected cost were derived. The term replacement (non-replacement) model implies that the probability for locating a record in a specific cylinder in the file remains (does not remain) constant when the cylinder has already been accessed for locating another record in the batch. A direct consequence of this assumption is that hit cylinders are non-distinct (distinct). In Sections 2 and 3 we examine these two problems by assuming both the replacement and the non-replacement model and derive two exact formulae for each case. Besides, for the second problem a new approximate formula is derived. If the batch size is small (large) compared with the file size, the non-replacement (replacement) model assumption is more realistic. In practice, the two models determine the two bounds on the disk head movement. In the last section the results are discussed.

2. Batched search for primary key values.

In this section we assume that the batch of q queries based on primary key values are to be satisfied from the file. In [4] it is assumed that the SCAN disk scheduling policy is used to answer the batch. This implies that the queries are satisfied by moving the disk heads forward starting from the first cylinder of the file. In [4] it is also proved that the total expected distance traveled by the disk head is:

$$(N-1)q/(q+1) \tag{1}$$

i.e. $q/(q+1)$ of the portion of the file will be searched for satisfying the batch. It is noted that similar formulae for other environments exist in [6–8, 10, 12]. Formula (1) is exact although it is based on successive approximations. Here we will a give a Lemma by using combinatorial analysis (as suggested in [14]) and then (a) for the sake of completeness we will give another proof of formula (1) by assuming the replacement model, and (b) we will provide a new accurate value by assuming the non-replacement model.

LEMMA. *Under the replacement and the non-replacement model, the probability distribution of the length of the subintervals (a) between any two successively hit cylinders or (b) between the first cylinder and the first hit cylinder or (c) between the last hit cylinder and the last cylinder respectively is:*

$$P(N,n,q) = C(N+q-n-2, q-1)/C(N+q-1, q) \tag{2}$$

$$\bar{P}(N,n,q) = C(N-n-1, q-1)/C(N,q) \tag{3}$$

where N is the total number of cylinders, n is the length of the subinterval, q is the magnitude of the query and $C(a,b)$ is the number of the a-choose-b combinations.

PROOF. For the non-replacement model see [2] which proves (3).

Consider now the first case for the replacement model. If the records are assumed to be retrieved from the non-distinct cylinders then the number of ways that q records can be selected from N cylinders is $C(N+q-1, q)$ [3]. It follows that if the q records are retrieved from the first $(N-n)$ non-distinct cylinders exactly, where n is the last not visited cylinders, then the number of ways that this may happen is:

$$C(N+q-n-1, q) - C(N+q-(n+1)-1, q) = C(N+q-n-2, q-1).$$

The probability that this may happen is derived by dividing the above number by the total number of selections of q records from N non-distinct cylinders. The same result holds for the other two cases which proves (2). ■

Note that as expected $P(N, n, 0) = \bar{P}(N, n, 0) = 0$ and $P(N, n, 1) = \bar{P}(N, n, 1) = 1/n$.

THEOREM 1. *A batch of q distinct sorted records is to be satisfied from a file residing in N consecutive cylinders. If the records are retrieved from q non-distinct or distinct cylinders then the expected distance traveled by the disk head is respectively*:

$$(N-1)q/(q+1), \tag{4}$$

$$(Nq-1)/(q+1). \tag{5}$$

PROOF. The expected distances traveled are:

$$\sum_{n=0}^{N-1} (N-n-1)P(N, n, q) \quad \text{and} \quad \sum_{n=0}^{N-q} (N-n-1)\bar{P}(N, n, q).$$

Formulae (4, 5) follow easily by using the properties of combinations [3]. ■

3. Batched search for secondary key retrieval.

In this section we consider the case when m queries based on a secondary key value are to be satisfied. For example, if we assume that $m = 2$, then two possible queries are "Retrieve all employees where DEGREE = M.SC." and "Retrieve all employees where SALARY > 30000". In [1] it is observed that the total disk head movement may be reduced if instead of satisfying each query on a FCFS basis the sytem performs the index searches for all the m queries. The fact that the pointer part for each index is usually ordered in terms of the cylinder number, track number etc. to be accessed allows the system to perform the following variation of the SCAN policy: odd (even) numbered queries are served by accessing the cylinders on which the records satisfying the queries reside on an ascending (descending) sequence. (Note: The possibility of searching all records to all the queries in one scan has been excluded due to the complication involved in the software required.)

The analysis in [1] proceeds as follows. Let q_i be the number of records satisfying each of the m queries $(1 \leq i \leq m)$. It is argued that if the disk head covers a distance of:

$$(N-1)(q_1+q_2)/(q_1+q_2+1)$$

cylinders, then the first query has already been answered and the second query is ready to be searched. In this way for the ith step the disk head travels a distance of:

$$(N-1)\left(\frac{q_i+q_{i+1}}{q_i+q_{i+1}+1}-1+\frac{q_{i-1}+q_i}{q_{i-1}+q_i+1}\right)$$

cylinders. Finally, it is proved that the total expected distance traveled in order to answer all m queries, each retrieving records from q_i non-distinct cylinders $(1 \leq i \leq m)$, is:

$$\left(2\sum_{i=1}^{m-1}\frac{q_i+q_{i+1}}{q_i+q_{i+1}+1}-m+1+\frac{q_m}{q_m+1}\right)(N-1). \tag{6}$$

A theorem proved in [1] states that (6) is minimized if the queries are arranged in descending order of magnitude according to the q records they contain.

The above analysis is approximate. This can be proved by means of the Lemma. Without loss of generality, suppose that the ith query has been answered, where i is even. Then according to the analysis in [1], the disk head will be positioned on the top of the cylinder with number:

$$A = (N-1)(q_i+q_{i+1})/(q_i+q_{i+1}+1).$$

Then the probability that at least one record of the $(i+1)$th query will be retrieved from an inner cylinder is:

$$\text{Prob}(n < N-A) = \sum_{n=0}^{N-A-1} P(N,n,q_{i+1}) = \sum_{n=0}^{N-1} P(N,n,q_{i+1}) -$$

$$-\sum_{n=N-A}^{N-1} P(N,n,q_{i+1}) = 1-C(A+q_{i+1}-1,q_{i+1})/C(N+q_{i+1}-1,q_{i+1}).$$

For large values of q_i this relation is finally simplified to:

$$(N-1)/(N-1+q_{i+1}). \tag{7}$$

This formula gives the probability that after the ith query is answered at least one record of the $(i+1)$th query will be retrieved from an inner cylinder. Therefore, the approximation involved in [1] will give optimistic results. Note, also, that the simplification in the derivation of relation (7) underestimates this probability even more.

We overcome the deficiency of the previous analysis by using conditional probabilities. We proceed to exact analysis and conclude to two new formulae for the expected distance traveled by the disk head according to both replacement and non-replacement models.

THEOREM 2. *Given a set of m queries, each retrieving from q_i non-distinct cylinders (where $1 \leq i \leq m$), to be searched in a file residing in N consecutive cylinders, the expected distance traveled by the disk head is*:

$$\begin{aligned}(8)\quad &(N-1)\left(m-2\sum_{i=2}^{m}\frac{1}{q_i+1}-\frac{1}{q_m+1}\right)+\\ &+2\sum_{i=2}^{m}\sum_{n=0}^{N-2}P(N,n,q_{i-1})C(N+q_i-n-1,q_i+1)/C(N+q_i-1,q_i).\end{aligned}$$

PROOF. The expected distance traveled for answering the first query is $\sum_{n=0}^{N-1}(N-1-n)P(N,n,q_1)$. According to Theorem 1 this distance is $(N-1)q_1/(q_1+1)$.

The expected distance traveled for answering the second query is:

$$\begin{aligned}&\sum_{r=0}^{N-1}P(N,r,q_1)\sum_{s=0}^{r-1}P(N,s,q_2)\times\\ &\times\left((r-s)+\sum_{t=0}^{N-s-1}(N-1-s-t)P(N-s,t,q_2-1)\right)+\\ &+\sum_{r=0}^{N-1}P(N,r,q_1)\sum_{s=r}^{N-1}P(N,s,q_2)\times\\ &\times\left((s-r)+\sum_{t=0}^{N-s-1}(N-1-s-t)P(N-s,t,q_2-1)\right).\end{aligned}$$

This expression is explained as follows. Suppose that the last record of the second query lies in an inner cylinder compared with the last record of the first query. Then the distance between them has to be covered. This distance is expected to be

$$\sum_{r=0}^{N-1}P(N,r,q_1)\sum_{s=0}^{r-1}(r-s)P(N,s,q_2)$$

where $r(s)$ is the number of the inner cylinder in which the last record of the first (second) query is stored. Therefore, after having reached the innermost cylinder hit by the second query, the direction of the head movement is reversed. The expected distance traveled for answering the second query is:

$$\sum_{r=0}^{N-1}P(N,r,q_1)\sum_{s=0}^{r-1}P(N,s,q_2)\sum_{t=0}^{N-s-1}(N-1-s-t)P(N-s,t,q_2-1)$$

where t is the number of the outer cylinder in which the first record of the second query is stored. Thus, the first (second) part of the expression represents the expected distance traveled in case that the last record of the second query lies at an inner (outer) cylinder compared with the last record of the first query.

Finally, after summation over all the m queries formula (8) is derived by using the properties of combinations [3]. ■

THEOREM 3. *Given a set of m queries, each retrieving records from q_i distinct cylinders (where $1 \leq i \leq m$), to be searched in a file residing in N consecutive cylinders, the expected distance traveled by the disk head is*:

$$(9)\qquad \begin{aligned} &(N+1)\left(m-2\sum_{i=2}^{m}\frac{1}{q_i+1}-\frac{1}{q_m+1}\right)-1+ \\ &\qquad +2\sum_{i=2}^{m}\sum_{n=0}^{N-q_{i-1}}\bar{P}(N,n,q_{i-1})C(N-n,q_i+1)/C(N,q_i). \end{aligned}$$

PROOF. Similar to Theorem 2. ■

Formulae (8, 9), although exact, are computationally expensive. In the next section it will be shown that formula (6) is a very close approximation of formula (8). Now we will provide a new approximate formula to be used in place of formula (9).

COROLLARY. *Formula* (9) *can be approximated by*:

$$(10)\qquad 2\sum_{i=1}^{m-1}\frac{N(q_i+q_{i+1})-1}{q_i+q_{i+1}+1}+\frac{Nq_m-1}{q_m+1}-(m-1)(N-1).$$

PROOF. With a similar reasoning as in [1] and by using formula (3) instead of (1) the proof follows easily. ■

4. Concluding remarks.

The concept of batching refers to a means of scheduling the queries in order to achieve better utilization of computer resources. In fact Schneiderman and Goodman argued that the potential reduction of processor demand may well reduce the response time [10]. The present study estimated the expected disk head movements while batched searching is based on primary or secondary key values. The aim of this section is twofold. First, to compare the results derived in this paper with results reported in [1, 4]. Second, to list a number of other studies which relate to batching.

We start with batching in primary key values. The method assumes that the system collects some requests and orders them according to key values.

Table 1. *Expected number of cylinders traveled for one batched query.*

N	q	Replacement (formula 4)	Non-replacement (formula 5)
100	5	82.5	83.2
100	10	90.0	90.8
100	15	92.8	93.7
200	10	180.9	181.7
200	15	186.6	187.4
200	20	189.5	190.4

The search is performed by moving the disk head in only one direction. Section 2 derived two exact formulae, one of them new. Table 1 presents the expected value of cylinders traveled by the disk head by applying formulae (2, 3). As can be proved easily with simple algebra, the replacement model always leads to smaller values than the values of the non-replacement model.

Table 2. *Expected number of cylinders traveled for a set of batched queries.*

N	q_i	Replacement exact (formula 8)	Non-replacement exact (formula 9)	Replacement approximate (formula 6)	Non-replacement approximate (formula 10)
100	5, 5	163.9	165.3	163.5	165.8
100	10, 5	169.5	171.1	169.1	171.5
100	5, 10	177.0	178.8	176.5	179.2
100	10, 10	180.0	181.7	179.6	182.2
100	13, 6, 3	243.3	245.6	242.6	246.6
100	6, 13, 3	251.4	254.0	250.7	254.8
100	13, 3, 6	252.1	254.8	251.4	255.5
100	6, 3, 13	259.2	262.0	258.5	262.7
100	3, 6, 13	261.0	263.6	260.2	264.5
100	3, 13, 6	262.0	264.8	261.3	265.6

When batching in secondary key values, secondary indexes are first accessed to provide the actual record addresses. Addresses are ordered and searched by alternating directions in which the disk head moves. Table 2 presents the expected values of cylinders traveled when a set of queries is batched searched by applying formulae 6, 8, 9 and 10. The observation that the replacement model gives smaller values than the non-replacement models holds also here between the pairs of formulae .(8, 9) and (6, 10). It is worth noting that approximations are very close to the exact results. In fact we observe that for the values in table 2 the deviation is less than 0.5%. Therefore our analysis validates the previous work and the approximate formulae may be used instead of the computationally expensive exact ones. The results also obey the rule of the Theorem in [1], which is based on the approximate analysis, namely that queries ordered in descending magnitude are answered more efficiently.

We finish this section by placing our results in perspective with some other results on searching for a number of keys. Batching has also been studied in terms of physical block accesses. The early work in [10] and recently in [8] derive approximate and exact formulae for the expected number of block accesses for successful searching of sequential and hierarchical files. In [7] new expressions are for partly or completely unsuccessful search of sequential or hierarchical files. In this study we did not concern ourselves with the problem of evaluating the number of disk blocks transferred. This problem has been extensively studied in the past. In [15] a non-replacement model is assumed and a formula is derived giving the expected number of block hits when blocks are randomly selected. In [9] a combination of multi-key hashing and inverted indices is proposed to achieve better clustering and less block transfers. In [16] it is assumed that the records do not have equal probabilities to be accessed and formulae are derived giving the expected number of blocks transferred when searching a sequential or random file. One possible extension of the study is to consider a batch of queries based on secondary key values and to try to determine in advance the optimum query satisfaction for more general environments than those in [1]. A result along this direction is reported in [5]. If this is done then it is worth evaluating the distance traveled by the disk head when the optimum sequence is applied. Finally, the determination of the distance traveled by the disk head when the probabilities of visiting a cylinder are not equal is also worth investigating.

Acknowledgement.

The authors would like to thank the referees for their helpful comments. Many thanks are also due to G. Poulakas for his programming effort.

REFERENCES

1. F. W. Burton and J. G. Kollias, *Optimising disk head movements in secondary key retrievals*, The Computer Journal, Vol. 22, No. 3, pp. 206–208, 1979.
2. S. Christodoulakis, *Analysis of retrieval performance for records and objects using optical disk technology*, ACM Transactions on Database Systems, Vol. 12, No. 2, pp. 137–169, 1987.
3. W. Feller, *An Introduction to Probability Theory and its Applications*, John Wiley, 3rd edition, 1968.
4. J. G. Kollias, *An estimate of the seek time for batched searching of random and index sequential files*, The Computer Journal, Vol. 21, No. 2, pp. 132–133, 1978.

5. J. G. Kollias and C. H. Papadimitriou, *The optimum execution order of queries in linear storage*, submitted to International Conference Extending Database Technology (EDBT 88), Venice, Italy, 1988.
6. Y. Manolopoulos, J. G. Kollias and M. Hatzopoulos, *Binary vs. sequential batched search*, The Computer Journal, Vol. 29, No. 4, pp. 368–372, 1986.
7. Y. Manolopoulos and J. G. Kollias, *Expressions for partly and completely unsuccessful search of sequential and tree-structured files*, submitted to IEEE Transactions on Software Engineering, 1987.
8. P. Palvia, *Expressions for batched searching of sequential and hierarchical files*, ACM Transactions on Database Systems, Vol. 10, No. 1, pp. 97–106, 1985.
9. J. B. Rothnie and T. Lozano, *Attribute based file organization in a paged memory environment*, Communications of the ACM, Vol. 17, No. 2, pp. 63–69, 1974.
10. B. Schneiderman and V. Goodman, *Batched searching of sequential and tree-structured files*, ACM Transactions on Database Systems, Vol. 1, No. 3, pp. 268–275, 1976.
11. T. J. Teory and T. B. Pinkerton, *A comparative analysis of disk scheduling policies*, Communications of the ACM, Vol. 15, No. 3, pp. 177–184, 1972.
12. T. J. Teorey and J. P. Fry, *Design of Database Structures*, Prentice Hall, Englewood Cliffs, N.J., 1982.
13. S. J. Waters, *Estimating magnetic disk seeks*, The Computer Journal, Vol. 18, No. 1, pp. 12–19, 1975.
14. C. K. Wong, *Minimizing expected head movement in one-dimensional and two-dimensional mass storage systems*, Computing Surveys of the ACM, Vol. 12, No. 2, pp. 167–178, 1980.
15. S. B. Yao, *Approximating block accesses in database organizations*, Communications of the ACM, Vol. 20, No. 4, pp. 260–261, 1977. .
16. J. Zahorian, B. Bell and C. Cevcik, *Estimating block transfers when record access probabilities are non-uniform*, Information Processing Letters, Vol. 16, No. 6, pp. 249–252, 1983.