The Impact of Seeking in Partial Match Retrieval

Athena Vakali and Yannis Manolopoulos
Department of Informatics
Aristotle University
54006 Thessaloniki, Greece
email: {avakali,manolopo}@athena.auth.gr

Abstract

We extend the problem of partial match query satisfaction by studying the impact of seeking. We study the physical location of the output pages by considering the number and the sparseness of cylinders holding the resulting pages. Lower and upper seek time bounds, as well as the average behavior of the seek time are calculated by assuming some real figures of specific modern disk system devices. The main conclusion is that the seek time is a factor affecting the partial match query response time and needs to be included in the overall performance measuring.

1 Introduction

A multi-attribute file is a collection of records characterized by more than one attribute, divided into pages and then stored in one or several disks. A partial match query (pm query) to a multi-attribute file is a request specifying certain values for one or several attributes. In such a case all pages containing at least one record having attribute values matching with the values specified by the pm query must be accessed. The problem of partial match retrieval on a multi-attribute file, residing on a single disk, has been examined in order to design an efficient allocation scheme minimizing the required number of disk accesses. More recently, a similar problem has been tackled by considering a set of parallel disk instead of a single one. The multi-disk file problem aims at grouping file records in pages and allocating pages in parallel disks so that a pm query is efficiently answered by maximizing concurrence of disk accesses.

Several data allocation methods have been studied lately in order to deal with the problem of assigning pages to one or several disks. The most frequent one is the Disk Modulo (DM) allocation method, which has been proven to be more effective because of its simplicity and its "strictly optimality" property (guaranteed under conditions). Variations of the DM method have been examined in [4,6] and optimal behavior is investigated under certain assumptions. In [3,7] Gray Codes are used in order to distribute binary Cartesian product files on multiple disks, while in [8] Error Correcting Codes have been introduced for file allocation. In all the previous works, the performance is measured in terms of the maximum number of pages that have to be accessed concurrently. Exceptionally, in [7] the number of clusters (i.e. a set of qualifying pages which are physically stored consecutively) is taken as a performance indicator. Summarizing, until now either the number of pages or the number of clusters have been taken as the crucial measure influencing performance's estimation.

In the present work we consider the seek time needed to respond to a query since any retrieval operation includes the time required to move to particular cylinders holding the required pages. Thus, the Partial Match Retrieval problem is studied under the new perspective of not having the pages consecutively stored on the disk, i.e. they are stored at different disk cylinders. Here, we assume that a SCAN (or CSCAN) disk scheduling policy is used, according to which the read/write heads move from the outer to the inner cylinder servicing requests that pend on each cylinder. In such a way, the cylinders where the pages lie are considered in our calculations for coming up with some performance estimates.

In the next section, we introduce some widely adopted equations used for modern disk system devices to describe the seek time as a function of the seek distance traveled. Then, we calculate lower and upper bounds for the seek time and perform calculations based on the sparseness of pages among cylinders by using some widely used equations for the seek time. Expected seek is also computed by using probability measures for the r requested cylinders (holding the b satisfying pages) spread among the data band of C

0-8186-7662-0/96 $05.00

cylinders which are occupied by the relation. Practically, seek time is proven to be an important factor needed to be taken in consideration and added to the overall response performance. Certain conclusions are summarized in the last section, as well as some areas are suggested for further research.

2 Expressions for Seek Time

Seek time is the time required to move to the appropriate cylinder which contains a requested page. Seek time has been examined extensively in the past since it is a major factor needed to be considered when studying disk performance. For quite long (since the early works for the hardware setup of the 1970s), it has been assumed that seek time is a linear function of seek distance traveled [16] :

$$T_{sk} = c_0 + c_1\, d$$

where d is the seek distance traveled and c_0, c_1 are two constant factors representing system's inertia and slope respectively.

Modern disk system devices though do not obey this linear equation. For example, in recent high performance disks with voice-coil actuators the fact of constantly accelerating and decelerating the access arm results in modeling the system performance by using the expression [1,5]:

$$T_{sk} = c_0 + c_1 \sqrt{d} \qquad (1)$$

Various values for the coefficients c_0, c_1 have been reported in the past (see [13] for a compilation of such values). More specifically, for the IBM 3380 model it has been proposed that c_0=2.1 and c_1=0.9 [11]. In addition, we note that a more general expression has been proposed in place of Relation (1):

$$T_{sk} = c_0 + c_1\, d^a$$

where $0 \leq a \leq 1$ [2].

Another approach to model seek time resulted in:

$$T_{sk} = c_0 + c_1 \sqrt{d} + c_2\, d \qquad (2)$$

Again, various values have been reported in the past with respect to the coefficients c_0, c_1 (see [13] for another compilation of such values). In [10], by using the IBM 3380 model, it has been taken that c_0=2.434, c_1=0.5555, c_2=0.01204 and $d < C$=885, where C is the total number of cylinders per disk surface.

Finally, a third approach to model seek time reached a more complex expression [15,14]:

$$T_{sk} = \begin{cases} c_0 + c_1 \sqrt{d} & \text{if } d < cutoff \\ c_2 + c_3\,(d - cutoff) & \text{if } d \geq cutoff \end{cases} \qquad (3)$$

where for model Tandem XL80 it has been accepted that c_0=5, c_1=0.64, c_2=14, c_3=0.02 and $cutoff$=0.2C [9].

Thus, based on the hardware requirements, a pm query responding might result in different overall response times than the ones calculated in the research carried out so far. As indicated in the introduction, most of the pm query problems have studied several optimality criteria and response time was measured in terms of the number of pages accessed from each disk. By expanding this aspect here, we question further the physical location of these requested pages. That is, we study the number of cylinders which contain the resulting pages, as well as whether these cylinders are consecutive and how they are distributed over the disk surface.

As explained earlier, the SCAN scheduling policy is considered to service the requests and we further assume (following the assumptions of reference [12]) that:

- The file occupies C consecutive cylinders.
- The pm query servicing is devoted each time to the satisfaction of a single query, so that there is no possibility that the heads might change servicing direction,
- The disk heads are initially positioned on top of the first (outer) cylinder and is ready to move towards the last (inner) disk file cylinder.

3 Upper and Lower Bounds in Seek Time

The purpose of this section is to derive upper and lower bounds on seek time for each one of Equations 1, 2 and 3, when a pm query posed on a multi-disk system is satisfied by a set of b pages, i.e. b is the maximum number of pages accessed from any disk. Furthermore, we accept that more than one requested pages might reside in the same cylinder. Thus, the total number of hit cylinders is r, where $1 \leq r \leq b$ and $r \leq C$. Here, we study the physical assignment of these b pages to r different cylinders.

In general when pages are spread among several $r > 1$ cylinders being apart, the total seek time for the different hardware devices described by Equations 1 and 2 is found by the following general formulae:

$$Total_{sk} = r\, c_0 + c_1 \sum_{i=1}^{r} \sqrt{d_i} \qquad (4)$$

$$Total_{sk} = r\,c_0 + c_1 \sum_{i=1}^{r} \sqrt{d_i} + c_2 \sum_{i=1}^{r} d_i \quad (5)$$

For the third case described by Equation 3 the total seek time is calculated by the formula:

$$Total_{sk} = r_s\,c_0 + c_1 \sum_{i=1}^{r_s} \sqrt{d_i} + r_l\,c_2 + c_3 \sum_{i=1}^{r_l} (d_i - cutoff) \quad (6)$$

where $0 \leq d_i \leq C - r$ are the cylinder distances to be crossed in order to reach the cylinder containing the required pages. Formula (6) holds since there are r_s short seeks of distances $d_i < cutoff$, whereas the rest r_l long seeks travel a distance $d_i \geq cutoff$. It is obvious that $r_s + r_l = r$, whereas for the case of Tandem XL80 $r_l \leq 5$ since $cutoff$=$0.2C$.

To obtain the lower bound of the seek time required to answer a pm query which is satisfied by a set of b pages residing in r cylinders we have to take in consideration the following facts:

- initially the r/w heads are positioned on top of the outer cylinder,
- the target group comprise a set of r consecutive cylinders,
- the first cylinder out of the r ones is the outer disk cylinder.

Under these assumptions, we understand that the seek time is a multiple of the system's inertia. This is based on the fact that in such a case there is no distance d needed to be traveled since cylinders (containing at least one pages each) are successive and start from the first cylinder on top of which the r/w head lies. Thus, the minimum seek time is:

$$Total_{sk}^{min} = (r-1)\,c_0$$

where $r-1$ is the actual number of stops in servicing the pm query. Therefore, the minimum seek time T_{sk}^{min} ranges in between the value of c_0 for $r = 2$ and the value of $(b-1)\,c_0$ for $r = b$. This result holds for Equation 6 model too, since the minimum cost will occur when the requested cylinders are consecutive so there is no distance $\geq cutoff$ that could be traversed.

In order to estimate the upper bound for the models of Equations 4 and 5, we have to consider that

- the last hit cylinder must be the last cylinder of the data band, i.e. the C-th one
- in [14] it has been proved that, if the r/w heads scan across a region of C cylinders and make $r-1$ stops, then the elapsed seek time is maximized when the $r-1$ stops are evenly apart by C/r cylinders (C is assumed a multiple of r).

Thus, for the model of Equation 4 we have that,

$$Total_{sk}^{max} = r\,c_0 + r\,c_1 \sqrt{\lceil C/r \rceil - 1}$$

and similarly for Equation 5 we have:

$$Total_{sk}^{max} = rc_0 + rc_1\sqrt{\lceil C/r \rceil - 1} + rc_2\,(\lceil C/r \rceil - 1)$$

In order to calculate the upper bound for the Equation 6 we question further the [14] tight upper bound estimation. Since the estimation of the maximum seek time is made by spreading the $r-1$ requested cylinders evenly apart by C/r, the last cylinder C is never reached, i.e. the whole of the data band area is never scanned completely. Also, in case that C is not a multiple of r, [14] suggests the use of $\lceil C/r \rceil$. This suggestion though, couldn't be used in all cases since it results in unrealistic figures. For example, for the disk HP-97560 there are C=1962, whereas $cutoff$=383. Therefore, when there are $r-1$=499 requested cylinders according to [14], they should be spread evenly apart by $\lceil C/r \rceil = \lceil 1962/500 \rceil = 4$, which can not hold in real since the 500 requests need 500×4=2000 cylinders totally. In order to overcome these facts we propose another approach here, aiming at a further maximization of the seek time by traversing all the disk area of C cylinders.

Thus, we propose a new estimation, in order to widen the scan cylinders area. We spread the r requested cylinders evenly apart by $\lfloor C/r \rfloor$ in order to reach cylinders being as close as possible to the last cylinder C. As indicated in the following example the upper bound is influenced in cases of relatively few requested cylinders.

Example: Suppose that the data band consists of C=1000 cylinders and there are r=4 requested cylinders. By considering the model of Equation 6 the cutoff value will be $0.2 \times C = 200$. In case of adopting the model of [14] approach the seek upper bound will be found by spreading r cylinders evenly apart by $d_i = C/5 = 200$ and the seek is equal to $T_{sk}^{max} = c_2 + c_3\,(d_i - cutoff) = 14$. In case of adopting the new presented here approach the seek upper bound will be found by spreading r cylinders evenly apart by $d_i = C/4 = 250$ and the seek is equal to $T_{sk}^{max} = c_2 + c_3\,(d_i - cutoff) = 15$. Thus, the new upper bound found here increases the seek time by a 6.67% rate.

Furthermore, suppose that there are r=19 requested cylinders. According to the model by Oyang [14] the seek upper bound is equal to $T_{sk}^{max} = 180.12$. On the other hand, according to the new approach presented here the seek upper bound is equal to $T_{sk}^{max} = 181.84$. Therefore, the new upper bound increases the seek time by a 1% rate. □

Several calculations have been carried out for several real modern disk devices and we noted that as the total cylinders number C is increased the upper bound is increased too. In case that we use C=2000 in the example above the seek time is increased by 7% and by 1.4%, for r equal to 4 and 19 respectively.

Thus, the generalized formula for the upper bound of Equation 6, becomes:

$$Total_{sk}^{max} = \begin{cases} c_0 + c_1\sqrt{d} & \text{if } \lfloor C/r \rfloor < cutoff \\ c_2 + c_3(d - cutoff) & \text{if } \lfloor C/r \rfloor \geq cutoff \end{cases}$$

Curves for the minimum and maximum of the three models are presented in Figures 1, 2, 3 for the three specific disk types by taking C=1000. In these figures minimum and maximum are pictured together with their respective expected seek (calculated and analyzed in the next section). The minimum results are independent of C so they remain the same despite the changes of C while maximum is affected since the formulae for the upper bound do include the number of cylinders C. In Table 1, the worst behavior is shown for C=1000. In all cases, we remark that seeking is an important factor and has to be taken in consideration.

Requests	Equation		
r	(1)	(2)	(3)
50	301.15	230.67	393.41
100	480.00	393.40	702.39
200	780.00	686.80	1206.22
300	1097.65	990.01	1832.55
400	1349.12	1256.44	2362.04

Table 1: Maximum seek time (msec) for C=1000 cylinders as a function of the number of requests r.

4 Expected Seek Time

In this section, we deal with the expected seek time of the three disk models described previously by the Equations 1, 2 and 3. Suppose that a pm query needs a retrieval of b pages in order to be satisfied. Assume that the b pages reside in r cylinders.

We calculate the expected seek based on probability measures. In order to satisfy the pm query r/w heads scan across a region of C cylinders and make r stops in order to visit the r satisfying cylinders. While traversing the disk, the r/w heads scan some specific sub-intervals of varying length d_i between successive requested cylinders. The length d_i of the sub-intervals varies between 0 for the case of consecutive cylinders and $C-1$ for the case of having r=1 located at the last (inner) cylinder. Therefore, the probability to have a distance $d = i$ to travel between two successive cylinders is given by the probability distribution function:

$$p(i) = \frac{\binom{C-i-1}{r-1}}{\binom{C}{r}}$$

Based on the above probability distribution function $p(i)$ we end up with the following formulae for the total expected seek. The first model of Equation 1 results in:

$$Total_{sk}^{exp} = r \sum_{i=0}^{C-1} p(i)\ (c_0 + c_1\ \sqrt{i}) \qquad (7)$$

whereas the second model of Equation 2 results in:

$$Total_{sk}^{exp} = r \sum_{i=0}^{C-1} p(i)\ (c_0 + c_1\ \sqrt{i} + c_2\ i) \qquad (8)$$

For the third case of the model described by Equation 3, the length i of the sub-intervals is either $\geq cutoff$ or $< cutoff$ producing two different quantities because of the different behavior of the models based on the $cutoff$. Thus, the expected seek becomes:

$$Total_{sk}^{exp} = r \left(\sum_{i=0}^{cutoff-1} p(i)\ (c_0 + c_1\ \sqrt{i}) + \sum_{i=cutoff}^{C-1} p(i)\ (c_2 + c_3\ (i - cutoff)) \right) \qquad (9)$$

In Table 2, the expected seek behavior is shown for C=1000. Again, we remark that even in the expected case, seeking is an important factor that is needed to be measured in the overall response to pm query process.

The resulted expected seek of Equations 7, 8 and 9 is depicted in Figures 1, 2 and 3. It is understood that the way of allocating pages to cylinders is quite important since even the minimum seek costs can delay the overall response time considerably.

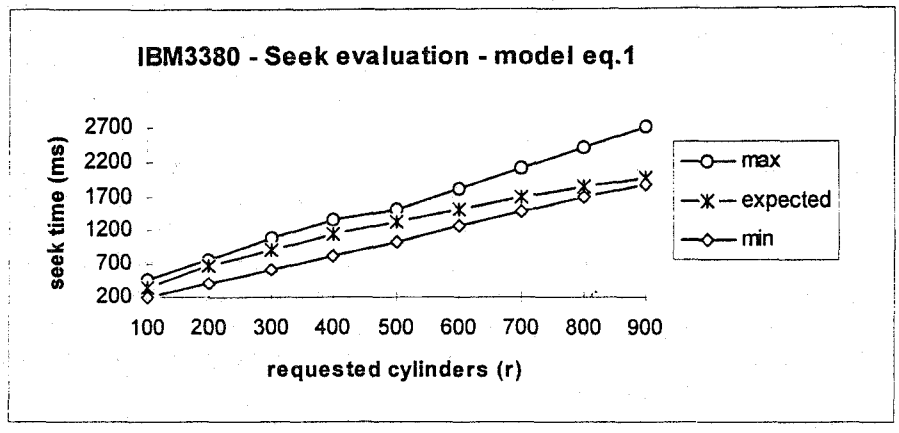

Figure 1: Minimum/expected/maximum seek time of device IBM 3380 (Equation 1).

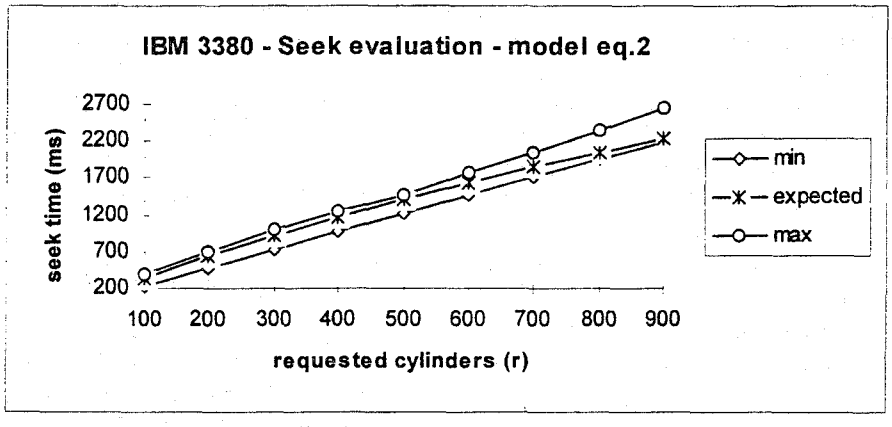

Figure 2: Minimum/expected/maximum seek time of device IBM 3380 (Equation 2).

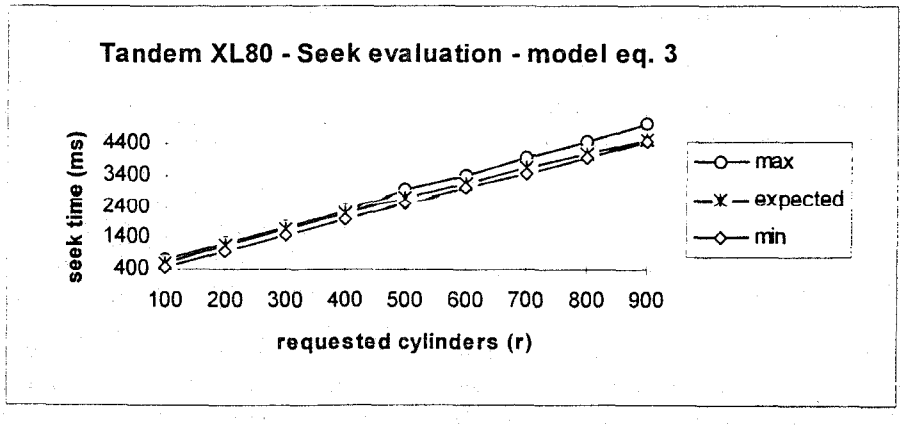

Figure 3: Minimum/expected/maximum seek time of device Tandem XL80 (Equation 3).

Requests	Equation		
r	(1)	(2)	(3)
50	163.03	144.70	238.25
100	360.83	329.17	562.40
200	674.22	643.12	1153.05
300	926.27	914.88	1692.00
400	1145.23	1164.54	2203.51

Table 2: Expected seek time (msec) for C=1000 cylinders as a function of the number of requests r.

5 Conclusions

Research conducted so far had examined several methods for allocation of records to pages as well as specific optimality properties for servicing pm queries. Here, partial match retrieval is extended by considering the seek distances needed to be traveled in order to find the satisfying pages in response to pm queries. Performance including seeking is calculated (in milliseconds) for certain disk devices, so that comparisons are made and the importance of considering cylinder positions where pages fit is emphasized. It is concluded that seek time is at least as important as block transfer time to answer pm queries. Therefore, allocation of pages to cylinders is evenly important as allocation of records to pages.

The research should expand in the area of declustering and data allocation, where new models need to be introduced together with the scheduling policies. Other query types (except pm queries) could also be examined and comparisons could be made in order to emphasize the efficiency of the new models.

References

[1] D. Bitton and J. Gray: Disk Shadowing, *Proceedings of the 14th VLDB Conference*, pp.331-338, 1988.

[2] A.R. Calderbank, E.G. Coffman and L. Flatto: A Note Extending the Analysis of Two-headed Systems to more General Seek Time Characteristics, *IEEE Transactions on Computers*, Vol.38, No.11, pp.1584-1586, 1989.

[3] C.C. Chang, H.Y. Chen and C.Y. Chen: Symbolic Gray Code as a Data Allocation Scheme for Two-Disk Systems, *The Computer Journal*, Vol.35. No.3, pp.299-305, 1992.

[4] C.C. Chen and H.F. Lin: Optimality Criteria of the Disk Modulo Allocation Method for Cartesian Product Files, *BIT*, Vol.31, pp.566-575, 1991.

[5] C.H. Chien: Seek Distances in Disks with Dual Arms and Mirrored Disks, *Performance Evaluation*, 1993.

[6] H.C. Du and J.S. Sobolewski: Disk Allocation for Cartesian Product Files on Multiple Disk Systems, *ACM Transactions on Database Systems*, Vol.7, pp.82-101, 1982.

[7] C. Faloutsos: Multiattribute Hashing using Gray Codes, *Proceedings of the 1986 ACM SIGMOD Conference*, pp.228-238, 1986.

[8] C. Faloutsos and D. Metaxas: Disk Allocation Methods using Error Correcting Codes, *IEEE Transactions on Computers*, Vol.40, No.8, pp.907-914, 1991.

[9] J. Gray, B. Horst and M. Walker: Parity Stripping of Disk Arrays: Low Cost Reliable Storage with Acceptable Throughput, *Proceedings of the 16th VLDB Conference*, pp.148-161, 1990.

[10] R.P. King: Disk Arm Movement in Anticipation of Future Requests, *ACM Transactions on Computer Systems*, Vol.8, No.3, pp.214-229, 1990.

[11] M.Y. Kim and A.N. Tantawi: Asynchronous Disk Interleaving - Approximating Access Delays, *IEEE Transactions on Computers*, Vol.40, No.7, pp.801-810, 1991.

[12] Y. Manolopoulos: Probability Distributions for Seek Time Evaluation, *Information Sciences*, Vol.60, No.1-2, pp.29-40, 1991.

[13] Y. Manolopoulos: Seek Time Evaluation, *Encyclopedia of Microcomputers*, Vol.15, pp.227-245, Marcel Dekker, NY, 1995.

[14] Y.J. Oyang: A Tight Upper Bound of the Lumped Disk Seek Time for the Scan Disk Scheduling Policy, *Information Processing Letters*, Vol.54, pp.355-358, 1995.

[15] C. Ruemmler and J. Wilkes: An Introduction to Disk Drive Modeling, *IEEE Computer*, Vol.27, No.3, pp.17-28, 1994.

[16] T.J. Teory and T.B. Pinkerton: A Comparative Analysis of Disk Scheduling Policies, *Communications of the ACM*, Vol.15, No.3, 1972.