Estimating Block Transfers and Join Sizes

Stavros Christodoulakis

Computer Systems Research Group

University of Toronto

Toronto, Canada M5S 1A1

Abstract

In this paper we provide estimates of the number of sequential and random block accesses required for retrieving a number of records of a file when the distribution of records in blocks of secondary storage is not uniform. We show how these results apply to estimating sizes of joins and semi-joins. We prove that when the uniformity of placement assumption is not satisfied it often leads to pessimistic estimates of performance. Finally we show a recursive estimation of the probability distribution of the number of blocks containing a given number of records.

1. Introduction

In response to a query a number of blocks have to be transferred from secondary storage to main memory. Estimates of the number of blocks of secondary storage that have to be transferred in main memory are important in physical data base design ([Yao77a], [Batory81], [Tsichritzis and Christodoulakis83]), data base system performance evaluation ([Sevcik81], [Christodoulakis81]), and query optimization ([Schneiderman and Goodman76], [Yao and DeJong78]). In addition, the distribution of the number of blocks containing a given number of records is important in the analysis of performance of concurrency control ([Ries79], [Potier and Leblank80]), and the estimation of rotational delays [Siler76]. Finally estimates of sizes of joins and semi-joins are especially important in distributed query optimization ([Kerschberg et al.80], [Bernstein et al. 81]). It is our thesis that data base system performance analysis can not be based on inaccurate estimates of these parameters.

Estimates of these parameters are known for uniform distributions. However, in several occasions it is the case that distributions are not uniform. For example attribute value distributions in a given domain often follow Zipf's law ([Kerschberg et al.80], [Siler 76]). In the case of records placed in blocks of secondary storage, several factors may contribute to non-uniform distributions of the records of a file over the blocks of the secondary storage. Some of these are clustering of different record types, variable record lengths, insertion and deletion activity and file creation method (hashing for example). Finally, when text is placed in blocks of

This work was supported in part by the Natural Science and Engineering Council of Canada under Strategic Grant G0668.

secondary storage the number of distinct non-trivial words [Tsichritzis and Christodoulakis83] varies among blocks of secondary storage.

In this paper we generalize several existing theoretical results for estimating expected values of these parameters to include non-uniform distributions. Then we prove that in several occassions uniformity results to pessimistic estimates of these parameters. Finally we give iterative formulae for the calculation of the probability distributions of the blocks containing a given number of records.

2. Block Transfers for Given Placement

Let t_j be the number of records in a block j of a file. Since we can see non-trivial words of a text file as variable length records, the discussion of this section applies to text files as well. In this case t_j is the number of distinct non-trivial words in the block j of the text file.

We assume in this section that t_j is given for $j=1, \cdots M$, where M is the number of blocks of the file. In an existing file this statistics can be extracted with a sequential scan of the file, or by using a dense primary index. Let m be the maximum possible number of records per block ($0 \le t_j \le m, j=1, \cdots M$). The maximum possible number of records per block can be found from the smallest possible record size (in bytes), and the size of a block (in bytes). Thus our model allows for fixed and variable size records in a file, for files where the number of records per block is variable due to insertions and deletions, for files where the number of records per block is variable due to the process that was used for creating the file (hashing for example) as well as for the case that records from more than one files are intermixed with records of the file under consideration.

Random Accesses

We first examine a *non-replacement* model ([Yao77b], [Christodoulakis81]). Consider a query Q for which k records of the file qualify. We assume that any selection of k records from the n records of the file has the same probability $\frac{1}{C_k^n}$ where C_k^n stands for the combinations of n objects taken k at a time. Let X_j be a random variable associated with a block j. X_j assumes the value 1 if the block contains any of the k records selected by Q, and 0 otherwise. The number of blocks randomly accessed for selecting the k records from the file is $B_r = \sum_{j=1}^{M} X_j$. The expected number of random block accesses $\bar{B}_r$ required for retrieving k qualifying records from the file is therefore

$$\bar{B}_r = \sum_{j=1}^{M} \bar{X}_j = \sum_{j=1}^{M} P(X_j=1)$$

where $P(X_j=1)=P(X_j)$ is the probability that block j contains any of the k records.

$P(X_j)$ is function of the number t_j of the records of the file in the block j.

Thus

$$P(X_j) = 1 - \frac{C_k^{n-t_j}}{C_k^n}$$

and

$$\bar{B}_r = \sum_{j=1}^{M} (1 - \frac{C_k^{n-t_j}}{C_k^n}) \qquad (1)$$

or

$$\bar{B}_r = \sum_{i=0}^{m} l_i \, (1 - \frac{C_k^{n-i}}{C_k^n}) \qquad (2)$$

where l_i is the number of blocks in the file which contain exactly i records ($\sum_{i=0}^{m} l_i = M$, $\sum_{i=0}^{m} i l_i = n$).

Both (1) and (2) reduce to Yao's formula

$$\bar{B}_r = M(1 - \frac{C_k^{n-b}}{C_k^n}) \qquad (3)$$

when the number of records in every block is constant and equal to b [Yao77b].

We provide an iterative formula for (2):

Set

$$F_i = \frac{C_k^{n-i}}{C_k^n}, \quad F_0 = \frac{C_k^n}{C_k^n} = 1.$$

Then

$$F_{i+1} = \frac{C_k^{n-i-1}}{C_k^n} = \frac{n-k-i}{n-i} F_i$$

and

$$\bar{B}_r = M - \sum_{i=0}^{m} l_i F_i$$

Thus $O(m)$ operations are needed to calculate $\bar{B}_r$. Similar iterative evaluations can be given for the resulting closed form formulae involving combinations throughout the paper.

We consider next a retrieval with *replacement* model: records that are retrieved are placed back ([Cardenas75], [Christodoulakis81]). Another interpretation of this model may be that an ordered set of k possibly duplicate records is retrieved [Cheung82]. In addition it provides a good approximation of the non-replacement model when the number of records per block is large [Yao77b]. The expected number of block accesses in this model when the number of records per block is not constant is

$$\bar{B}_r = \sum_{j=1}^{M} (1-(1-\frac{t_j}{n})^k)$$

$$\bar{B}_r = \sum_{i=0}^{m} l_i (1-(1-\frac{i}{n})^k) \qquad (4)$$

When the number of records per block is constant (4) reduces to Cardena's formula [Cardenas75]:

$$\bar{B}_r = M(1-(1-\frac{1}{M})^k) \qquad (5)$$

It can be shown that the expected number of random block accesses of the non-replacement model is greater or equal to the expected number of block accesses of the replacement model. Indeed:

$$\frac{C_k^{n-i_j}}{C_k^n} = \frac{n-i_j}{n} * \frac{n-i_j-1}{n-1} * \cdots * \frac{n-i_j-k+1}{n-k+1} \le (\frac{n-i_j}{n})^k.$$

The result follows by comparing (2) and (4). The special case of uniform placement is discussed in [Yao77b].

Sequential Accesses

In this section we estimate the expected block accesses for finding k records in sequentially accessed files ([Yao 77a], [Schneiderman and Goodman76]). Let $t = \sum_{r=1}^{j} i_r$ be the total number of records in the first j blocks of the file. The probability that *exactly* j blocks have to be examined sequentially in order to find all the k qualifying records is

$$q_j = \frac{C_k^t - C_k^{t-1}}{C_k^n}$$

(e.g. all possible selections of k records from the j first blocks minus all possible selections of k records from the $j-1$ first blocks divided by the total number of selections of k records). Thus the expected number of blocks $\bar{B}_s$ that need be accessed sequentially in order to retrieve the k qualifying records is

$$\bar{B}_s = \sum_{j=1}^{M} j \frac{C_k^t - C_k^{t-1}}{C_k^n}$$

Or

$$\bar{B}_s = M - \frac{\sum_{j=1}^{M} C_k^{t-1}}{C_k^n} \qquad (6)$$

In the special case of uniform placement of exactly b records in each block, (5) reduces to Yao's formula [Yao77a]:

$$\bar{B}_s = M - \frac{\sum_{j=1}^{M} C_k^{(j-1)b}}{C_k^n} \qquad (7)$$

Next we derive estimates of the sequential block accesses using the replacement model. The probability that all k retrievals will be from the first t_j records is $\left[\frac{t_j}{n}\right]^k$. Thus the expected sequential accesses using the

replacement model is

$$\bar{B}_s = \sum_{j=1}^{M} j \left[\left\lceil \frac{t_j}{n} \right\rceil^k - \left\lceil \frac{t_{j-1}}{n} \right\rceil^k \right] \quad (8)$$

or

$$\bar{B}_s = M - \frac{\sum_{j=1}^{M-1} (t_j)^k}{n^k} \quad (9)$$

Again since $\frac{C_{t_j}^k}{C_t^k} \leq (\frac{t_j}{n})^k$ it follows that the expected accesses for the non-replacement model is greater or equal to the expected block accesses for the replacement model.

3. Block Transfers when Placement is not Known

When statistics of the distribution of the number of records per block does not exist (is not maintained or the file does not exist yet) it may be possible that it is derived by knowing the process that creates the file. Let P_i, $i=0,m$ describe the probability distribution of the number of records per block in the file. Then the expected random block accesses is given by:

$$\bar{B}_r = M \left[1 - \sum_{i=0}^{m} P_i \frac{C_k^{n-i}}{C_k^n} \right] \quad (10)$$

Let $P_j(t)$ be the probability that in the first j blocks of the file exist exactly t records, $q_{ji}(i)$ be the probability that the jth block contains exactly i records given that the j first blocks contain t records, and $R_{jsi}(k)$ be the probability that exactly j blocks of the file need be examined sequentially in order to find all the k qualifying records given t and i. Then

$$\bar{B}_s = \sum_{j=1}^{M} j \sum_{t=1}^{n} P_j(t) \sum_{i=1}^{m} q_{ji}(i) * R_{jsi}(k) \quad (11)$$

where

$$R_{jsi}(k) = \frac{C_k^t - C_k^{t-i}}{C_k^t} \quad (12)$$

Given P_i, $P_j(t)$ and $q_{ji}(i)$ equations (10), (11) and (12) can be used for calculating the expected random and sequential block accesses for retrieving k records.

When various record types involving records of different lengths are intermixed the number of records per block will vary with serious implications on system performance [Teorey and Das76], [Teorey and Oberlander76]. The same will be true when the file is formed from records of a single record type but the length of records follows a distribution. In [Teorey and Oberlander76] it is shown how to estimate the probability distributions and the number of blocks M of a file when the frequencies and the sizes in bytes of the various record types involved are known. Thus (10), (11), and (12) can be directly applied in this environment. Similar analysis can be done for a single record type but variable length records.

Another environment where these distributions can be easily derived is a hashed file environment where each block is selected with the same probability $(\frac{1}{M})$ each time that a record is inserted. Assuming no overflows, a binomial distribution may be used:

$$P_i = C_i^n (\frac{1}{M})^i (1 - \frac{1}{M})^{n-i}$$

where n is the number of records inserted in the file so far. Similarly

$$P_j(t) = C_t^n (\frac{j}{M})^t (1 - \frac{j}{M})^{n-t}$$

and

$$q_{ji}(i) = C_i^t (\frac{1}{j})^i (1 - \frac{1}{j})^{t-i}$$

In the following we will examine an environment where records of fixed size are randomly placed in the slots of the address space of the file. This for example may model the record placement in blocks of a file after a large number of insertions and deletions.

A Random Placement Model

The *address space* of a file is the ordered set $A=(1,2,\cdots N)$. A single element of A is called *linear address*. Let $n \le N$ be the number of records in the file. A *placement* of the n records of the file is a set of n distinct linear addresses. In a *random placement model* the n records of the file are randomly placed among the N linear addresses and each possible placement is equally probable. (We assume in this section fixed length records.)

In this model the probability distribution P_i can be calculated as

$$P_i = \frac{C_i^m \, C_{n-i}^{N-m}}{C_n^N}. \qquad (13)$$

Thus

$$B_r = M\left[1 - \sum_{i=0}^{m} \frac{C_i^m \, C_{n-i}^{N-m}}{C_n^N} * \frac{C_k^{n-i}}{C_k^n}\right] \qquad (14)$$

For the sequential access case we obtain

$$P_j(t) = \frac{C_i^{jm} \, C_{n-i}^{N-jm}}{C_n^N} \qquad (15)$$

and

$$q_{jt}(t) = \frac{C_i^{jm} \, C_i^{(j-1)m}}{C_i^{jm}} \qquad (16)$$

Thus

$$B_s = \qquad (17)$$

$$\frac{\sum_{j=\left\lfloor \frac{k}{m} \right\rfloor}^{M} j * \sum_{i=k}^{\min(n,jm)} C_{n-i}^{N-jm} \sum_{i=1}^{m} C_i^m \, C_i^{(j-1)m} \left(C_k^i - C_k^{i-1}\right)}{C_n^N \, C_k^n}$$

Example 1

Consider a file of $M=3$ blocks each having $m=2$ positions such that the size of the address space of the file is $N=M*m=6$. Let $n=3$ records be placed randomly in these 6 possible positions. Let $k=2$ records being selected. Using (13) and (14) we calculate $P_0=\frac{4}{20}$, $P_1=\frac{12}{20}$, $P_2=\frac{4}{20}$. Thus $\bar{B}_r=\frac{108}{60}$. These numbers are verified in figures (1) and (2) where all possible placements are shown. If a constant number of $b=1$ records existed in each block then using (3) we calculate $\bar{B}_r=\frac{120}{60}$.

For the sequential access case using (15), (16), and (17) we calculate $P_1(2)=\frac{4}{20}$, $P_2(2)=\frac{12}{20}$, $P_2(3)=\frac{4}{20}$, $P_3(3)=1$. Also $q_{12}(2)=1$, $q_{22}(1)=\frac{4}{6}$, $q_{22}(2)=\frac{1}{6}$ $q_{23}(1)=\frac{1}{2}$, $q_{23}(2)=\frac{1}{2}$, $q_{33}(1)=\frac{12}{20}$, $q_{33}(2)=\frac{4}{20}$. Thus $\bar{B}_s=\frac{152}{60}$. These numbers are verified in figures (3) and (4). If a constant number $b=1$ of records existed in each block then using (7) we calculate $\bar{B}_s=\frac{160}{60}$.

4. Sizes of Joins and Semi-joins

Let R be a relation and A an attribute of R which takes values on an ordered finite domain of values $D=(V_1,\cdots V_M)$. In this section we assume independence of attribute values of the attributes of a relation.

The *value vector* $V_A=(n_1,\cdots,n_M)$ is defined such that n_i is the number of tuples in R which have value V_i for attribute A $(i=1,\cdots,M, \sum_{i=1}^{M} n_i = n)$. The number of tuples in the equi-join on an attribute A of two relations $V'_A=(n'_1\cdots n'_M)$ on attribute A is calculated as

$$SJ = \sum_{i=1}^{M} n_i n'_i$$

[Kershberg et al.80]. If k and k' records are selected from each of the relations before the join, then the expected size of the join can be easily shown to be

$$SJ = \frac{kk'}{nn'} \sum_{i=1}^{M} n_i n'_i \qquad (18)$$

When semi-joins are used in the query evaluation method [Bernstein et al 81], in order to choose a good query evaluation strategy it is important to have good estimates of the number of distinct values remaining in

$N=6 \quad M=3 \quad m=2 \quad n=3 \quad k=2$

Placements of n records	# blocks with zero records	# blocks with 1 record	# blocks with 2 records
123	1	1	1
124	1	1	1
125	1	1	1
126	1	1	1
134	1	1	1
135	0	3	0
136	0	3	0
145	0	3	0
146	0	3	0
156	1	1	1
234	1	1	1
235	0	3	0
236	0	3	0
245	0	3	0
246	0	3	0
256	1	1	1
345	1	1	1
346	1	1	1
356	1	1	1
456	1	1	1
Total C_n^N=20 possible placements	$f_0=\frac{12}{20}$	$f_1=\frac{36}{20}$	$f_2=\frac{12}{20}$
	$P_0=\frac{4}{20}$	$P_1=\frac{12}{20}$	$P_2=\frac{4}{20}$

Figure 1

Random placement of $n=3$ records in $N=6$ locations. Expected number of blocks with 0, 1, and 2 records. Probabilities of blocks with 0, 1, 2 records.

$N=6 \quad M=3 \quad m=2 \quad n=3 \quad k=2$

Placements of n objects	Selections of $k=2$ objects for each placement	Blocks transfered for each selection
123	12 13 23	1+2+2
124	12 14 24	1+2+2
125	12 15 25	1+2+2
126	12 16 26	1+2+2
134	13 14 34	2+2+1
135	13 15 35	2+2+2
136	13 16 36	2+2+2
145	14 15 45	2+2+2
146	14 16 46	2+2+2
156	15 16 56	2+2+1
234	23 24 34	2+2+1
235	23 25 35	2+2+2
236	23 26 36	2+2+2
245	24 25 45	2+2+2
246	24 26 46	2+2+2
256	25 26 56	2+2+1
345	34 35 45	1+2+2
346	34 36 46	1+2+2
356	35 36 56	2+2+1
456	45 46 56	2+2+1
Total C_n^N=20 placements	$C_n^N * C_2^n$=60 selections	Total = 108 $= \hat{B}=\frac{108}{60}$

Figure 2

Selection of $k=2$ records from the $n=3$ records of the file. Average number of blocks selected.

$N=6 \quad M=3 \quad m=2 \quad n=3 \quad k=2$

Random plac. of n objects	$j=1$ Distr. of l	$j=2$ Distr. l	$j=3$ Distr. l	$j=2,l=2$ Distr. l_2	$j=2,l=3$ Distr. l_2	$j=3,l=3$ Distr. l_3
123	2	3	3	-	1	0
124	2	3	3	-	1	0
125	2	2	3	0	-	1
126	2	2	3	0	-	1
134	1	3	3	-	2	0
135	1	2	3	1	-	1
136	1	2	3	1	-	1
145	1	2	3	1	-	1
146	1	2	3	1	-	1
156	1	1	3	-	-	2
234	1	3	3	-	2	0
235	1	2	3	1	-	1
236	1	2	3	1	-	1
245	1	2	3	1	-	1
246	1	2	3	1	-	1
256	1	1	3	-	-	2
345	0	2	3	2	-	1
346	0	2	3	2	-	1
356	0	1	3	-	-	2
456	0	1	3	-	-	2
—	—	—	—	—	—	—
Total 20	$p_1(0)=\frac{4}{20}$	$p_2(0)=0$	$p_3(0)=0$	$q_2(0/2)=\frac{2}{12}$	$q_2(1/3)=\frac{2}{4}$	$q_3(0/3)=\frac{4}{20}$
	$p_1(1)=\frac{12}{20}$	$p_2(1)=\frac{4}{20}$	$p_3(1)=0$	$q_2(1/2)=\frac{8}{12}$	$q_2(2/3)=\frac{2}{4}$	$q_3(1/3)=\frac{12}{20}$
	$p_1(2)=\frac{4}{20}$	$p_2(2)=\frac{12}{20}$	$p_3(2)=0$	$q_2(2/2)=\frac{2}{12}$		$q_3(2/3)=\frac{4}{20}$
		$p_2(3)=\frac{4}{20}$	$p_3(3)=1$			

$$p_j(l) = \frac{C_l^{jm}\, C_n^{N-jm}}{C_n^N} \qquad q_j(i\,|\,l) = \frac{C_i^{jm}\, C_l^{(j-i)m}}{C_l^{jm}}$$

Figure 3

Probability distributions used in the estimation of the sequential access case.

the joining domain as well as good estimates of the tuples remaining in a relation after a semi-join is performed [Bernstein et al.81]. By semi-join $R'[A=A>R$ we mean the join of R' and R on attribute A followed by the projection on the attributes of R. It can be performed by sending the distinct values R'_A of R' in A to the sites containing R and eliminating those tuples of R which do not contain a value in R'_A.

The *value probability* vector $P_A^R = (P_1, \cdots, P_M)$ as its name indicates provides an estimate of the probability that a given value in the joining domain is non-zero at any point of the query evaluation process. It is created and maintained as follows:

Creation: If $n_i=0$ then $P_i=0$ else $P_i=1$

Update:

a) selections or semi-joins on other attributes of R such that k records remain in R:

$$P_i = 1 - \frac{C_k^{n-n_i}}{C_k^n} \tag{19}$$

b) projections on A or supersets of A do not affect P_i

c) selection on A of the form $A='V_i'$, $A\geq'V_i'$, $A\leq'V_i'$, $A<>'V_i'$ set $P_i=0$ for the components of P_A^R which do not qualify.

d) semi-join $R'[A=A>R$ with another relation R' with probability vector $P_A^{R'}$ $(P_1', \cdots, P_M')$ produces a new

$N=6 \quad M=3 \quad m=2 \quad n=3 \quad k=2$

Random Placement of n objects	Selections of $k=2$ objects	Prob of exactly j blocks needed	Blocks transferred for each selection
123	12 13 23	$P(2/3,1,2)=\frac{2}{3}$	1+2+2
124	12 14 24		1+2+2
125	12 15 25		1+3+3
126	12 16 26		1+3+3
134	13 14 34		2+2+2
135	13 15 35		2+3+3
136	13 16 36		2+3+3
145	14 15 45		2+3+3
146	14 16 46		2+3+3
156	15 16 56	$P(2/2,1,2)=\frac{1}{3}$	3+3+3
234	23 24 34		2+2+2
235	23 25 35		2+3+3
236	23 26 36		2+3+3
245	24 25 45		2+3+3
246	24 26 46		2+3+3
256	25 26 56		3+3+3
345	34 35 45	$P(2/2,2,2)=\frac{1}{3}$	2+3+3
346	34 36 46		2+3+3
356	35 36 56		3+3+3
456	45 46 56		3+3+3
Total	Total	$P(j/l,i,k)=\frac{C_i^k-C_{i-l}^k}{C_i^k}$	Total 152
$C_n^N=20$ placements	$C_n^N \cdot C_k^n=60$ selections		$\hat{B}=\frac{152}{60}$

Figure 4

Sequential access:

Selection of $k=2$ records from $n=3$ records of the file. Average number of blocks transferred.

vector

$$P_A^{RR'}(P_1P_1', \cdots, P_MP_M')$$

The expected number of distinct values remaining in the joining domain A after a semi-join (or join) is

$$\bar{D} = \sum_{i=1}^{M} P_iP_i' \tag{20}$$

The expected number of tuples of R remaining after a semi-join (or join) is

$$n_1 = \frac{k}{n} \sum_{i=1}^{M} P_i'n_i \tag{21}$$

where k is the number of records isolated from selections or semi-joins on other domains.

Figure 5 shows an example. The distribution of the number of tuples of three relations R_1, R_2 and R_3 on a common domain D are shown. We assume for this example a query evaluation strategy which performs a selection on R_1 (for which 3 records qualify) followed by a semi-join $R_1[A=A>R_2$ followed by a semi-join $R_2[A=A>R_3$. The estimated number of distinct values and semi-join sizes are shown in the figure for two different ways of estimating join sizes.

We note here that in order to reduce the number of calculations required, attribute values in a joining domain can be grouped into classes so that the members of each class have approximately the same number of records per value.

5. Implications of Non-Uniform Placement

In this section we compare different placements of a number of objects in a number of buckets and we draw some conclusions for the expected block accesses and semi-join sizes in a data base environment.

Distribution of tuples per value in domain A.

values	R_1	R_2	R_3
V_1	100	100	1
V_2	1	1	100
V_3	0	1	500

Estimated sizes with uniformity and independence

values	Probability after selection	Probability after semi-join	Probability after semi-join
V_1	18/30	18/30	18/30
V_2	18/30	18/30	18/30
V_3	18/30	18/30	18/30
	Total # of values $=3*\frac{18}{30}=1.8$	Total # of values=1.8 size of semi-join $=\frac{18}{30}*102$ $=54$	Total # of values=1.8 size of semi-join $=\frac{18}{30}*601=320$

Estimated sizes using probability vector

values	Prob. after selection	Prob. after semi-join	Prob. after semi-join
V_1	1	1	1
V_2	4/101	4/101	4/101
V_3	0	0	0
	Total # of values $=\frac{105}{101}$	Total # of values $=\frac{105}{101}$	Total # of values $=\frac{105}{101}$
	Size of semi-join = 100.4		Size of semi-join = 5

Figure 5

k=3 records are selected from R_1 then the semi-joins with R_2 and R_3 are performed.
Two different ways of estimating selectivities in semi-joins are indicated.

Definition: A vector $\mathbf{a}=(a_1,\cdots,a_M)$ with $a_1\geq a_2\cdots\geq a_M$ is said to majorize a vector $\mathbf{b}=(b_1,b_2,\cdots,b_M)$ with $b_1\geq b_2\geq\cdots\geq b_M$ if $\sum_{i=1}^{k} a_i \geq \sum_{i=1}^{k} b_i$ for k=1 to $M-1$ and $\sum_{i=1}^{M} a_i = \sum_{i=1}^{M} b_i$.

Intuitively if a vector $\mathbf{a}$ majorizes another vector $\mathbf{b}$, then the components of $\mathbf{a}$ deviate more from uniformity than the components of $\mathbf{b}$. In the following a_i and b_i will be non-negative integers for all i.

Theorem: Let $\mathbf{a}=(a_1,\cdots a_M)$ and $\mathbf{b}=(b_1,\cdots b_M)$ be two vectors satisfying the property that a_i's and b_i's are nonnegative integers and when the components of $\mathbf{a}$ and $\mathbf{b}$ are rearranged such that $a_1\geq a_2\geq\cdots\geq a_M\geq 0$ and $b_1\geq b_2\cdots\geq b_M\geq 0$ then $\mathbf{a}$ majorizes $\mathbf{b}$. Let $\bar{E}_r(\mathbf{x})$ be the cost function (1), where $\mathbf{x}$ is an M-dimensional vector. Then $\bar{E}_r(\mathbf{a})\leq\bar{E}_r(\mathbf{b})$.

Proof

We first observe that given a vector $\mathbf{d}$ we can permute its components without affecting $\bar{E}_r(\mathbf{d})$. Thus without loss of generality we will assume that the given vectors $\mathbf{a}$ and $\mathbf{b}$ are such that $a_1\geq a_2\geq\cdots a_M$ and $b_1\geq b_2\geq\cdots\geq b_M$. Let $\mathbf{a}$ and $\mathbf{b}$ satisfy $\mathbf{a}$ majorizes $\mathbf{b}$. The distance $\|\mathbf{a}-\mathbf{b}\|$ of the two vectors is defined to be $\|\mathbf{a}-\mathbf{b}\|$

$= \sum_{i=1}^{M} |x_i - y_i| = 2l$, where l is non-negative integer.

The following inequality holds for $n \geq m \geq k \geq 1$.

$$C_n^k + C_n^m \leq C_n^{k+1} + C_n^{m-1} \qquad (22)$$

Indeed

$$C_n^k - C_n^{k+1} + C_n^m - C_n^{m-1} =$$

$$\frac{\prod_{n-k+1}^{n}(i) - \prod_{n+1-k+1}^{n+1}(i) + \prod_{m-k+1}^{m}(i) - \prod_{m-k}^{m-1}(i)}{k!} =$$

$$\frac{-k\prod_{n+1-k+1}^{n}(i) + k\prod_{m-k+1}^{m-1}(i)}{k!} \leq 0 \quad \text{for } n \geq m \geq k \geq 1$$

We form successive vectors $d^0=b, d^1, \cdots d^l=a$ as follows: if $d^k=(i_1^k, \cdots i_M^k)$ for $k=0,l-1$, then $d^{k+1}=(i_1^{k+1}, \cdots, i_M^{k+1})$ is formed from d_k by decreasing by one a component with index p of d^k which satisfies $i_p^k > a_p$ and increasing by one a component q of d^k which satisfies $i_q^k < a_q$, and then rearranging the components such that $i_1^{k+1} \geq i_2^{k+1} \cdots \geq i_M^{k+1}$. The following relations hold:

$$||d^{k+1} - d^k|| = \sum_{j=1}^{M} |i_j^{k+1} - i_j^k| = 2$$

$$||a - d^{k+1}|| = \sum_{j=1}^{M} |a_j - i_j^{k+1}| = \sum_{j=1}^{M} |a_j - i_j^k| - 2 = ||a - d^k|| - 2,$$

a majorizes d^{k+1} majorizes d^k majorizes b,

and

$$B_r(d^k) \geq B_r(d^{k+1})$$

(using 22).

Thus $B_r(b) = B_r(d^0) \geq B_r(d_1) \geq \cdots \geq B_r(d_l) = B_r(a)$.

We will next show that a similar result holds for the non-replacement model.

Definition: A function of n real variables is called Schur concave if for every pair $i \neq j$, $(x_i - x_j)(\frac{\partial f}{\partial x_i} - \frac{\partial f}{\partial x_j}) \leq 0$

Theorem (Schur): If f is a Schur concave function and a majorizes b then $f(a) \leq f(b)$.

Proof: See for example [Marshall and Olkin 79].

A function which is known to be Schur concave is the entropy function of a probability distribution.

Theorem: Let $a=(a_1, \cdots, a_M)$ and $b=(b_1, \cdots b_M)$ be two vectors satisfying the property that a_i's and b_i's are non negative integers and when the components of a and b are rearranged such that $a_1 \geq a_2 \cdots \geq a_M$ and $b_1 \geq b_2 \cdots \geq b_M$ then a majorizes b. Let $B_r(x)$ be the cost function (4), where x is an M dimensional vector. Then $B_r(a) \leq B_r(b)$.

Proof: Without loss of generality we assume that the given vectors a and b are such that $a_1 \geq a_2 \geq \cdots \geq a_M$ and $b_1 \geq b_2 \geq \cdots \geq b_M$. The function $B_r(i_1, \cdots, i_M) = \sum_{j=1}^{M}(1 - (1 - \frac{i_j}{n})^k)$ is Schur concave because:

$$(i_j - i_i)\left[\frac{\partial B_r}{\partial i_j} - \frac{\partial B_r}{\partial i_i}\right] =$$

$$= (i_j - i_i)\left[+k(1-\frac{i_j}{n})^{k-1}\frac{1}{n} - k(1-\frac{i_i}{n})^{k-1}\frac{1}{n}\right]$$

$$= \frac{k}{n}(i_j - i_i)\left[(1-\frac{i_j}{n})^{k-1} - (1-\frac{i_i}{n})^{k-1}\right] \leq 0$$

Thus if a majorizes b then $B_r(a) \leq B_r(b)$.

The following set of corollaries are direct applications of the above theorems to the problem of estimating the number of random block accesses for retrieving k records. They apply to both the non-replacement and the replacement model.

Corollary1: Let $a=(a_1, \cdots a_M)$ and $b=(b_1, \cdots b_M)$ be two vectors describing two placements of the n tuples of a relation in M blocks of secondary storage. Without loss of generality $a_1 \geq a_2 \cdots \geq a_M$ and $b_1 \geq \cdots \geq b_M$. Let a and b be such that a majorizes b. Then the expected number of blocks containing k distinct records in placement a is less or equal to the expected number of blocks containing k distinct records in placement b.

In the limiting case that is possible to place the records uniformly in the blocks of the file, the following corollary holds.

Corollary2: From all possible placements of n records in M blocks of secondary storage a uniform one will result to a maximum expected number of block accesses for randomly retrieving k records from the file. (Uniform here means that either b or $b+1$ records are placed in each block.)

Corollary3: From all uniform placements of n records in secondary storage the one that utilizes the least number of blocks will result to a minimum expected number of random block accesses, and vice versa.

The above theorems can also be used to provide upper and lower approximations of the expected number of random block accesses when the maximum and minimum number of records in blocks of the file is known (in addition to the number of blocks and the number of records of the file). An upper approximation will be one that is derived assuming a uniform distribution of the records among the blocks of the file. A lower approximation will be one which is derived by assuming a distribution which follows the constrains of the maximum and minimum records per block, and also majorizes any other distribution.

The following corollaries are applications of the above theorems to the problem of estimating the expected number of distinct values remaining in an attribute after a selection of a subset of the records of a file.

Corollary4: Let $\mathbf{a} = (a_1, \cdots, a_M)$ and $\mathbf{b} = (b_1, \cdots b_M)$ be two vectors describing two distributions of n-tuples over the M distinct values of an attribute A. Without loss of generality $a_1 \geq a_2 \geq \cdots \geq a_M$ and $b_1 \geq b_2 \geq \cdots \geq b_M$. Let $\mathbf{a}$ and $\mathbf{b}$ be such that $\mathbf{a}$ majorizes $\mathbf{b}$, and let $S(\mathbf{a})$ and $S(\mathbf{b})$ be the expected number of distinct values of A remaining after the selection of $k<n$ tuples (respectively). Then $S(\mathbf{a}) \leq S(\mathbf{b})$.

Corollary5: Let $\mathbf{n}=(n_1, \cdots, n_M)$ be a vector describing the distribution of the attribute values of the tuples of a relation on the domain of values of an attribute A. Let k tuples be randomly selected from the file. From all possible distributions $\mathbf{n}$ the uniform distribution will result to a maximum expected number of distinct values found in the attribute A of the k selected tuples.

6. Distributions of the Number of Blocks Containing a set of Records

In this section we show how to estimate the probability that *exactly* P blocks contain a set of k distinct records. A *block type* is defined by the number of records that exist in a block at a point in time. Let c be the number of different block types and $\mathbf{n}=(n_1, \cdots n_c)$ with $n_1 > n_2 > \cdots > n_c$ be the *type characteristic vector*, where n_i is the number of records in a block of type i. Let $\mathbf{l}=(l_1, \cdots, l_c)$ be the distribution of blocks of a file over the block types (e.g. there are l_1 blocks containing n_1 records each, ... l_c blocks containing n_c records each). A *block selection vector* $\mathbf{i}=(i_1, \cdots, i_c)$ has components which represent the number of blocks from each block type examined at a point in time. (Component i_j corresponds to the type containning n_j records per block.)

Distributions for the non-replacement model

Consider p *given* blocks with distribution $\mathbf{i}=(i_1, \cdots, i_c)$ over the types of blocks ($|\mathbf{i}| = \sum_{j=1}^{c} i_j = p$). The number of distinct ways that *all* the p blocks are retrieved with the selection of k distinct records from the p blocks can be computed recursively as

$\mathbf{l} = (2, 2)$

$\mathbf{n} = (2, 1)$, $n = 2*2+1*2=6$

$k = 3$

Possible selection of k=3 records	P=2	P=#
123	v	
124	v	
125	v	
126	v	
134	v	
135		v
136		v
145		v
146		v
156		v
234	v	
235		v
236		v
245		v
246		v
256		v
345	v	
346	v	
356		v
456		v
	total=8	total=12
	$Q(P=2)=\frac{8}{20}$	$Q(P=3)=\frac{12}{20}$

Calculation

$\mathbf{i}=(1,0)$ $F=C_2^2=0$

$\mathbf{i}=(0,1)$ $F=C_3^2=0$

$Q(1)=0$

$\mathbf{i}=(0,2)$ $F=C_3^2=0$

$\mathbf{i}=(2,0)$ $F=C_3^2=4$

$\mathbf{i}=(1,1)$ $F=C_3^2=1$

$$Q(2) = \frac{C_2^2C_0^2*4+C_1^2C_1^2*1}{C_3^2} = \frac{8}{20}$$

$\mathbf{i}=(2,1)$ $F=C_3^2-C_2^2C_0^1*4-C_1^2C_1^1*1=4$

$\mathbf{i}=(1,2)$ $F=C_3^2-C_1^1C_1^2*1=2$

$$Q_3=\frac{C_2^2C_1^2 4+C_1^2C_2^2 2}{C_3^2}=\frac{12}{20}$$

Figure 6
Example of calculation of distributions in the non-replacement model.

$$F(\mathbf{i},\mathbf{n},k) = C_k^{\sum_{j=1}^{c} n_j i_j} - \sum_{\substack{|\mathbf{i}'|=0 \\ 0\le i'_j\le i_j}}^{|\mathbf{i}|-1} F(\mathbf{i}',\mathbf{n},k) * \prod_{j=1}^{c} C_{i_j}^{i'_j}$$

with boundary condition for $\mathbf{i}=(0,\cdots,0)$ $F(\mathbf{i},\mathbf{n},k)=0$.

The probability that exactly p blocks are selected when k records are retrieved without replacement from all the blocks of the file is

$$Q(p) = \frac{\sum_{\substack{|\mathbf{i}|=p \\ 0\le i_j\le l_j}} F(\mathbf{i},\mathbf{n},k) * \prod_{j=1}^{c} C_{i_j}^{l_j}}{C_k^n}$$

In the special case where there is only one block type (constant number of b records per block) we have: $c=1$, $\mathbf{n}=(M)$ and $\mathbf{l}=(M)$, where M is the number of blocks in the file. Then

$$Q(p) = \frac{F(p,b,k)*C_p^M}{C_k^n}$$

with

$$F(p,b,k) = C_k^{nb} - \sum_{i=0}^{p-1} F(i,b,k) C_i^p$$

This result appears in [Langer and Shum82].

Distributions for the replacement model

Consider p given blocks with distribution $\mathbf{i}=(i_1,\cdots,i_c)$ over the types of blocks ($\sum_{j=1}^{c} i_j=p$). The number of distinct possible outcomes is $(\sum_{j=1}^{c} n_j i_j)^k$.

Some of these outcomes will be from a proper subset of the p blocks. The number of possible outcomes that touch all p blocks is computed recursively as

$$F(\mathbf{i},\mathbf{n},k) = (\sum_{j=1}^{c} n_j\, i_j)^k - \sum_{\substack{|\mathbf{i}'|=0 \\ 0\le i'_j\le i_j}}^{|\mathbf{i}|-1} F(\mathbf{i}',\mathbf{n},k) \prod_{j=1}^{c} C_{i_j}^{i'_j}$$

with boundary condition for $\mathbf{i}=(0,0,\cdots 0)$ $F(\mathbf{i},\mathbf{n},k)=0$.

i=(1,1)

n=(2,1) $n=2*1+1*1=3$

Possible outcomes	$P=1$	$P=2$	Calculation
111	v		i=(1,0) $F=2^3=8$
112	v		
113		v	i=(0,1) $F=1^3=1$
121	v		
122	v		$Q(1)=\frac{2^3*C_1^1+1^3C_1^1}{3^3}=\frac{9}{27}$
123		v	
131		v	
132		v	
133		v	i=(1,1) $F=3^3-8-1=18$
211	v		
212	v		
213		v	$Q(2)=\frac{18*C_1^1*C_1^1}{3^3}=\frac{18}{27}$
221	v		
222	v		
223		v	
231		v	
232		v	
233		v	
311		v	
312		v	
313		v	
321		v	
322		v	
323		v	
331		v	
332		v	
333	v		
total $3^3=27$	total 9	total 18	
$P(1)=\frac{9}{27}$	$P(2)=\frac{18}{27}$		

Figure 7 Example of calculation of distributions in the replacement model.

The probability that exactly p distinct blocks are selected when k records are retrieved with replacement is

$$Q(p) = \frac{\sum_{\substack{|\mathbf{i}|=p \\ 0\le i_j\le l_j}} F(\mathbf{i},\mathbf{n},k) * \prod_{j=1}^{d} C_{l_j}^{i_j}}{n^k}$$

In the special case of a constant number of records per block this calculation becomes

$$Q(p) = \frac{F(p,b,k)*C_p^M}{n^k}$$

with

$$F(p,b,k)=(pb)^k-\sum_{i=0}^{p-1}F(i,b,k)C_i^p$$

7. Summary

In this paper we have derived estimates for some important parameters in data base performance evaluation when the distributions of objects into buckets are not uniform. Parallel research activity in this area is directed towards more accurate estimates of the size of projections ([Gelenbe and Cardy82], [Gelenbe and Cardy83]), more accurate estimates of join sizes ([Kerschberg et al.80], [Rosenthal81]), more accurate estimates of record selectivities in the presence of non-uniformity and correlations of attribute values [Christodoulakis83], more accurate estimates of block transfers

when the probabilities of records to be accessed are not uniform ([Zahorian et al.83], [Christodoulakis81]) and estimates of the number of blocks containing a number of qualifying records [Langer and Shum82].

We have shown in this paper that the assumption of uniform placement in certain cases results to pessimistic estimates. This result complements previous results ([Christodoulakis81], [Christodoulakis82]) indicating that uniformity and independence of attribute values, as well as uniformity of probabilities of records to be accessed are also often pessimistic assumptions. Understanding the implications of various assumptions made is an important issue in modeling the data base system performance.

Acnowledgements: The author would like to thank C. Faloutsos for careful reading of the paper and useful suggestions.

References

[Aho et al.74]

Aho, A.V., Hopcroft, J.E. and Ullman, J.D.: "*The Design and Analysis of Computer Algorithms*", Addison-Wesley. 1974.

[Batory81]

Batory, D.S.: "An Analytic Model of Physical Databases", Ph.D. Thesis, Technical Report CSRG-124, University of Toronto, 1981.

[Bernstein et al.81]

Bernstein, P.A., Goodman, N., Wong, E., Reeve, C., Rothnie, D.B.: "Query Processing in a System for Distributed Databases (SDD-1)", *ACM TODS 6*, 4, December 81, 602-625.

[Cardenas75]

Cardenas, A.F.: "Analysis and Performance of Inverted Database Structures", *CACM 18*, 5, May 1975, 253-263.

[Cheung82]

Cheung To-Yat: "Estimating Block Accesses and Number of Records in File Management", *CACM 25*, 7, 1982, 484-487.

[Christodoulakis81]

Christodoulakis, S.: "Estimating Selectivities in Data Bases", Ph.D. Thesis, Technical Report CSRG-136, University of Toronto, 1981.

[Christodoulakis82a]

Christodoulakis, S.: "Implications of Certain Assumptions in Data Base Performance Evaluation", submitted for publication, 1982.

[Christodoulakis82b]

Christodoulakis, S.: "Issues in Query Evaluation", *IEEE Database Engineering 5*, 3, 1982, 48-51.

[Christodoulakis83]

Christodoulakis, S.: "Estimating Record Selectivities", *Information Systems 8*, 2, 1983 (to appear).

[Christodoulakis and Faloutsos82]

Christodoulakis, S., and Faloutsos, C.: "Performance Considerations for a Message File Server", in *Alpha-Beta*, Report CSRG#143, University of Toronto, 1982 (F. Lochovsky editor).

[Gelenbe and Cardy82]

Gelenbe, E. and Cardy, D.: "The Size of Projections of Relations Satisfying a Functional Dependency", *VLDB 8*, Mexico City, September 1982, 325-333.

[Gelenbe and Cardy83]

Gelenbe, E. and Cardy, D.: "On the Size of Projections I", *Information Processing Letters 1983* (to appear).

[Kerschberg et al.80]

Kerschberg, L. Ting, P.D., and Yao, S.B.: "Optimal Distributed Query Processing", Technical Report, Bell Labs Holmdel, 1980.

[Kollias78]

Kollias, J. B. : "An estimate of Seek Time for Batched Searching of Random or Index Sequential Structured Files", *The Computer Journal 21*, 2, May 78, 132-133.

[Langer and Shum82]

Langer, A. and Shum, A.: "The Distribution of Granule Accesses Made by Database Transactions", *CACM 25*, 11, November 82, 831-832.

[Marshall and Olkin79]

Marshall, A. and Olkin, I.: "Inequalities" *Theory of Majorization and its Applications*", Academic Press 1979.

[Potier and Leblank80]

Potier, D., and Leblank, P.: "Analysis of Locking Policies in Database Management Systems", *CACM 29,10*, Oct. 80, 584-593.

[Rosenthal 81]

Rosenthal, A.: "Note on the Expected Size of a Join", *SIGMOD record*, July 81.

[Ries79]

Ries, D.: "The Effect of Concurrency Control on Database Management System Performance", Ph.D. Dissertation, Computer Science Department, University of California, Berkeley, April 1979.

[Riordan58]

Riordan, J.: "*An Introduction to Combinational Analysis*", Wiley, New York, 1958.

[Schneiderman and Goodman 76]

Schneiderman, B. and Goodman, V.: "Batched Searching of Sequential and Tree Structured Files", *ACM TODS 1*, 3, Sept 76, 268-275.

[Sevcik81]

Sevcik, K.: "Data Base System Performance Prediction Using an Analytic Model", *Proc. VLDB 9181*, 182-198.

[Siler76]

Siler, K.F.: "A Stochastic Evaluation Model for Database Organizations in Data Retrieval Systems", *CACM 19*, 2, February 76, 84-95.

[Teorey and Das76]

Teorey, T.J. and Das, K.S.: "Application of an Analytical Model to Evaluate Storage Structures", *Proc. ACM SIGMOD 1976*, 9-19.

[Teorey and Oberlander78]

Teorey, T.J. and Oberlander, L.B.: "Network Database Evaluation Using Analytical Modeling", *Proc. NCC 1978*, 833-842.

[Tsichritzis and Christodoulakis83]

Tsichritzis, D., and Christodoulakis, S.: "Message Files", *ACM Transactions on Office Information Systems 1*, Jan. 1983, 88-98.

[Yao77a]

Yao, S.B.: "An Attribute Based Model for Database Access Cost Analysis", *ACM TODS 2*, 1, March 77, 45-67.

[Yao77b]

Yao, S.B.: "Approximating Block Accesses in Database Organizations", *CACM 20*, 4, April 77, 260-261.

[Zahorian et al.83]

Zahorian, J., Bell, B., Sevcik, K.: "Estimating Block Transfers When Record Access Probabilities are non-uniform", *Information Processing Letters*, 1983, (to appear).