Applications: Operations and Management

Alan Merten Editor

Estimating Block Accesses in Database Organizations: A Closed Noniterative Formula

KYU-YOUNG WHANG and GIO WIEDERHOLD *Stanford University*

DANIEL SAGALOWICZ *SRI International*

ABSTRACT: ***A closed, noniterative formula is introduced for estimating the number of block accesses in a database organization, and the error analyzed. This formula, an approximation of Yao's exact formula, has a maximum error of 3.7 percent, and significantly reduces the computation time by eliminating the iterative loop. It also achieves a much higher accuracy than an approximation proposed by Cardenas.***

1. INTRODUCTION

In evaluating the access cost of a query for a database organization in which records are grouped into blocks in secondary storage [16], one must often estimate the number of block assesses required to retrieve the records selected by the query. Various formulas have been proposed for this purpose [2, 3, 8–13, 17, 19]. In particular, Yao [17] presented the following theorem:[1]

THEOREM 1: [Yao] *Let n records be grouped into m blocks* ($1 \le m \le n$), *each containing* $p = n/m$ *records. If k records*[2] *are randomly selected from the n records, the expected number of blocks hit (blocks with at least one record selected) is given by*

$$b(m, p, k) = m\left[1 - \binom{n-p}{k} \Big/ \binom{n}{k}\right] \quad (1)$$

$$= m[1 - ((n-p)!(n-k)!)/((n-p-k)!n!)] \quad (2)$$

$$= m\left[1 - \prod_{i=1}^{k} (n-p-i+1)/(n-i+1)\right] \quad (3)$$

when $k \le n - p$, and

$$b(m, p, k) = m \quad \text{when } k > n - p. \quad (4)$$

[1] The notation and some of the conditions have been slightly modified.

[2] Cases in which k or m are random variables with various distributions have been studied [7].

Authors' Present Addresses: Kyu-Young Whang and Gio Wiederhold, Computer Systems Laboratory, Stanford University, Stanford, CA 94305; and Daniel Sagalowicz, Artificial Intelligence Center, SRI International, Menlo Park, CA 94025.

This work was supported by the Defense Advanced Research Project Agency under the KBMS Project, Contract N39-80-G-0132.

Earlier Cardenas [3] suggested the formula

$$b_c(m, p, k) = m[1 - (1 - 1/m)^k], \qquad (5)$$

assuming that there are n records divided into m blocks and that the k records are randomly selected from the n records. It is interesting to note that Eq. (5) is independent of the blocking factor p.

Yao [17] showed that Eq. (5) is based on the assumption that records are selected *with replacement*, i.e., a record can be selected more than once. But this assumption does not hold in practice, since records selected by a query simultaneously must be distinct from one another. Yao eliminated this assumption and proved Theorem 1 under the assumption that records are actually selected *without replacement*, i.e., a record cannot be selected more than once at one time.

Theorem 1 gives the exact formula under the given assumptions. However, we notice that Eq. (3) has an iterative form, which will take excessive time to evaluate if k becomes large. Another way of evaluating Yao's formula is by using the Gamma function (in practice a Log Gamma (LGAM) [6] function should be used, since the Gamma function grows very steeply). By modifying Eq. (2) slightly, we obtain

$$\begin{aligned} &b(m, p, k) \\ &= m[1 - \exp(\mathrm{LGAM}(n - p + 1) + \mathrm{LGAM}(n - k + 1) \qquad (6) \\ &\qquad - \mathrm{LGAM}(n - p - k + 1) - \mathrm{LGAM}(n + 1))]. \end{aligned}$$

Evaluation of this formula poses a problem in practice, especially when k is small. Since, in the evaluation of the argument of the exponential function, we are subtracting big numbers from equally big numbers to get a very small number, the roundoff error of the computation can become intolerable. For example, when Eq. (6) is calculated by using single-precision variables on a 36-bit machine having the resolution of 2^{-27} ($\simeq 10^{-8}$) [4], it has a 46% error at $p = 10$, $m = 1000$, $n = 10000$, and $k = 2$. The roundoff error is 310% when $p = 10$, $m = 3162$, $n = 31620$, and $k = 3$. But these values of parameters are well within the range of relevant databases.

We propose below a closed, noniterative formula that approximates Yao's exact formula with reasonable accuracy, as well as reducing considerably the computation error caused by limited precision.

2. A NONITERATIVE FORMULA

In this section, we introduce the following formula and discuss how it was obtained. Errors of this formula will be discussed in Section 3. We assume throughout that m and k have only integer values.

$$\begin{aligned} b_{w1}(m, p, k)/m = &[1 - (1 - 1/m)^k] \\ &+ [1/m^2p \times k(k - 1)/2 \\ &\quad \times (1 - 1/m)^{k-1}] \qquad (7) \\ &+ [1.5/m^3p^4 \times k(k - 1)(2k - 1)/6 \\ &\quad \times (1 - 1/m)^{k-1}] \\ &\qquad \text{when } k \le n - p, \text{ and} \end{aligned}$$

$$b_{w1}(m, p, k)/m = 1 \qquad \text{when } k > n - p \qquad (8)$$

Let us see how Eq. (7) has been derived. When $k > n - p$, we always have $b_{w1}(m, p, k)/m = 1$ from Eq. (4). If we use $n = mp$, Eq. (3) can be transformed to an equivalent form

$$b(m, p, k)/m = 1 - \prod_{i=0}^{k-1} (1 - 1/m(1 - i/mp)) \qquad (9)$$

If we perform a series expansion on $1/m(1 - i/mp)$ and take only the first three terms, we obtain

$$b(m, p, k)/m \simeq 1 - \prod_{i=0}^{k-1} ((1 - 1/m) - i/m^2p - i^2/m^3p^2)$$

If we expand the multiplication and keep the first three terms, we get

$$\begin{aligned} b(m, p, k)/m \simeq &[1 - (1 - 1/m)^k] \\ &+ [1/m^2p \times k(k - 1)/2 \\ &\quad \times (1 - 1/m)^{k-1}] \qquad (10) \\ &+ [1/m^3p^2 \times k(k - 1)(2k - 1)/6 \\ &\quad \times (1 - 1/m)^{k-1}]. \end{aligned}$$

Eq. (10) is only an approximation of Eq. (9), since we took only a few terms from the expansions. Two factors were added to Eq. (10) to derive Eq. (7). The factor 1.5 has been introduced empirically to compensate for the errors at small values of p, i.e., $p \simeq 1$. It was chosen especially to reduce the error to zero when $p = 1$, $k = [n - p]$, as n goes to infinity ($n \to \infty$), in which case Eq. (10) has the most significant error. The factor $1/p^2$ has been introduced empirically to reduce the effect of the third term for higher values of p, for adding the third term at these values of p increases the error (although it reduces the error at lower values of p). Let us note that the first term is identical to Cardenas' formula, with the remaining two terms compensating for its error. We shall show later that the approximation formula derived here constitutes a practically negligible deviation from the exact formula.

3. ERROR ANALYSIS

We note that the first term of Eq. (7) is identical to Cardenas' formula, Eq. (5). The second term compensates for the major

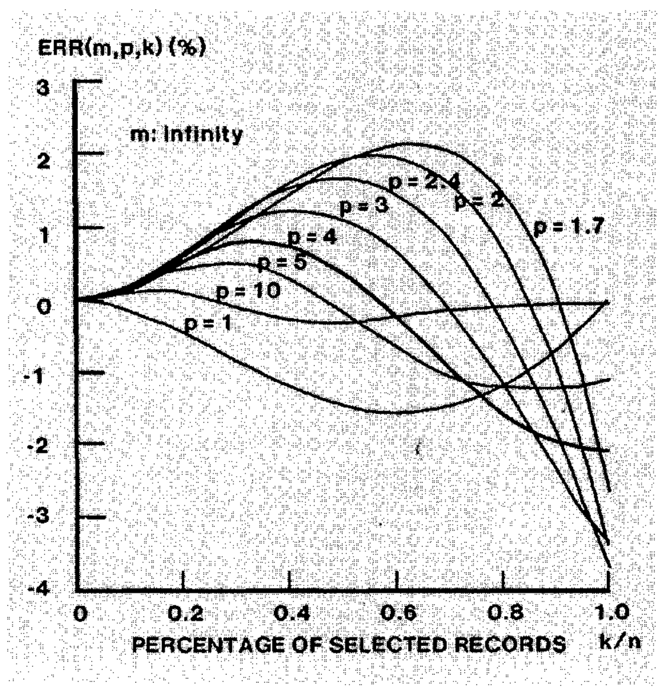

FIGURE 1. Error of Eq. (7) as m Goes to Infinity.

error of Eq. (5), while the third term provides a finer adjustment to further reduce the error. The third term has been empirically modified to get a better approximation.

Derived in Theorem 2 and plotted in Figure 1 for various values of p and $\kappa = k/n$ is a formula that gives the limiting values of error $ERR(m, p, k) = (b(m, p, k) - b_{w1}(m, p, k)/b(m, p, k)$ as the total number of blocks m (and, accordingly, the total number of records n) goes to infinity.

THEOREM 2:

$$\lim_{m\to\infty} ERR(m, p, k) = 1 - (1 - e^{-p\kappa}(1 - p\kappa^2/2 - \kappa^3/2p))/(1 - (1 - \kappa)^p) \quad (11)$$

where $ERR(m, p, k) = (b(m, p, k) - b_{w1}(m, p, k))/b(m, p, k)$, and p and κ have fixed values.

PROOF: To derive this formula, we need the following form of Yao's formula, which has the iteration on the blocking factor p rather than on the number of selected records k.

$$b(m, p, k) = 1 - \prod_{i=1}^{p} ((n - k - i + 1)/(n - i + 1)) \quad (12)$$

This formula is easily derivable from Eq. (2). If we subtract Eq. (7) from Eq. (12), and divide the result by Eq. (12), we can obtain Eq. (11), by taking the limit as $m \to \infty$ (accordingly $n \to \infty$) and by using the identity $\lim_{m\to\infty} (1 - 1/m)^m = e^{-1}$. Q.E.D.

Eq. (12) is also a convenient formula for evaluating the exact value when we have integer blocking factors. In fact, all computed values for integer blocking factors that we shall employ later in this section were produced by using Eq. (12). The limiting values, as the blocking factor p goes to infinity with m and κ fixed, are proved to be zero in the following theorem.

THEOREM 3: $\lim_{p\to\infty} ERR(m, p, k) = 0$, where m and $\kappa = k/n$ have fixed values. Here $0 \le k \le (m - 1)p$, and k is an integer.

PROOF: If $\kappa = 0$, both $b(m, p, k)/m$ and $b_{w1}(m, p, k)/m$ simply become 1, so $ERR(m, p, k)$ must be zero. If $\kappa > 0$, we know that $\lim_{p\to\infty} b(m, p, k) \ge 1$, since at least one block must be hit. Therefore, the denominator of $ERR(m, p, k)$ is always at least 1 and cannot be 0. To study the behavior of the numerator, let us look at Eq. (12). In Eq. (12), $(n - k - i + 1)/(n - i + 1) \le (n - k)/n = 1 - \kappa < 1$. Therefore, $\lim_{p\to\infty} \prod_{i=1}^{p} ((n - k - i + 1)/(n - i + 1)) \le \lim_{p\to\infty} (1 - \kappa)^p = 0$. Thus, $\lim_{p\to\infty} b(m, p, k) = 1$. But it is clear from Eq. (7) that $\lim_{p\to\infty} b_{w1}(m, p, k) = 1$ also, since $\lim_{p\to\infty} (1 - 1/m)^{\kappa mp-1} = \lim_{p\to\infty} (1 - 1/m)^{\kappa mp} = \lim_{p\to\infty} e^{-\kappa p} = 0$, and an exponential order can suppress any polynomial order of p. Hence, $\lim_{p\to\infty} ERR(m, p, k) = 0$. Q.E.D.

The errors that occur when both n and p are finite were investigated by performing an exhaustive computer calculation. These analyses show that Eq. (7) yields at most 3.7% (−3.7% if the sign is considered) of deviation from the exact formula, Eq. (3), over the entire range of $p \ge 1$, $m \ge 1$, $0 \le k \le n - p$, where m and k are integers. This maximum error occurs at $p = 1 + \sqrt{2}$, $k = \lfloor n - p \rfloor$ as $m \to \infty$ (This can be observed in Figure 1. The maximum error and the value of p at which this error occurs can in fact be derived from Eq. (11) once we know that this occurs at $\kappa = 1$, as $m \to \infty$.) The maximum positive error (2.5%) occurs at $p = 1.5$, $k = 3$, and $m = 3$. The maximum positive error when $m \to \infty$ is 2.1% at $p = 1.7$ and $k = 0.65n$.

The dependence of the error on the values of $n/p = m$ is shown in Figure 2, where k is set to be equal to $\lfloor n - p \rfloor$. (Note that the maximum error occured at this k value.) At low values of m and p, there is a short range within which errors are changing by a large amount, since at these values of m and p, $k = \lfloor n - p \rfloor = \lfloor (m - 1)p \rfloor$ in the range where high positive errors occur, as we see in Figure 1 (see the value when $p = 2$, $m = 3$, $n = 6$, and $k = 4$, for example). The dependence of the error on m is otherwise very flat, as in

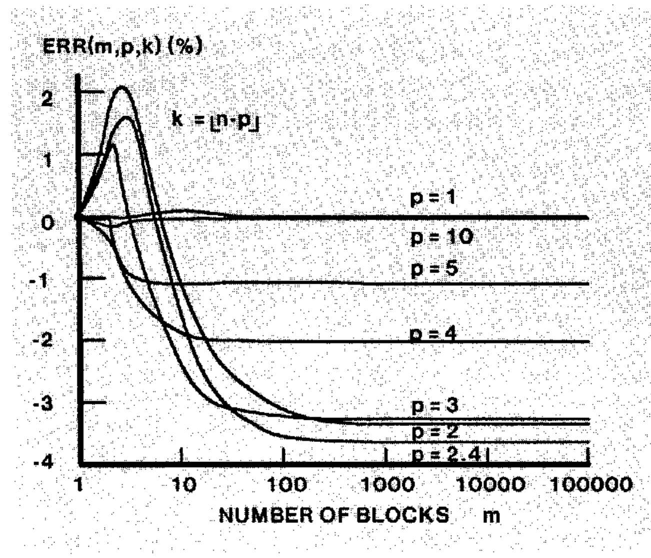

FIGURE 2. Error of Eq. (7) when k = [n − p].

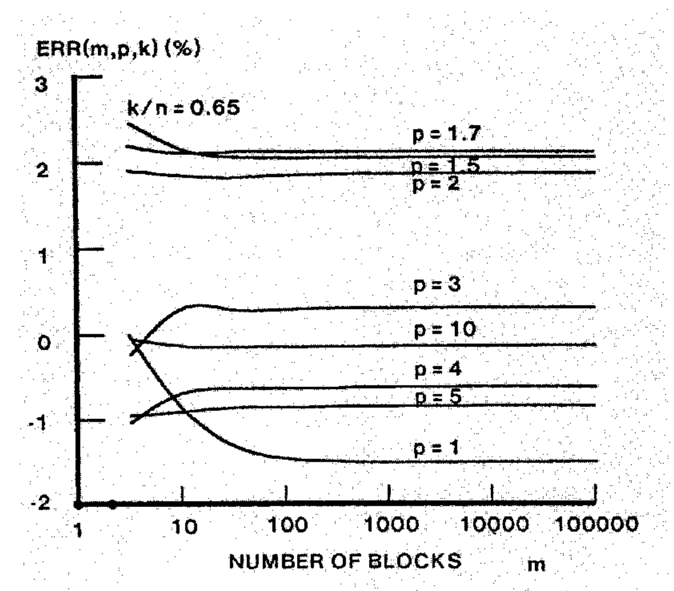

FIGURE 3. Error of Eq. (7) when k = 0.65n.

Figure 3, which shows the values when $\kappa = 0.65$ with corresponding k values rounded to the nearest integers. In Figure 3, the values at $m = 1$ and $m = 2$ are 0 from Eq. (4) and Eq. (8), since at these points $k = 0.65n > n - p$.

The values of variables we used in the exhaustive computer calculation with the constraint that $mp \leq 10^7$ (10^6 for noninteger blocking factors) are as follows:

- m: 1; 2; 3; 10; 32; 100; 316; 1,000; 3,162; 10,000; 31,623; 100,000; 316,228; 1,000,000
- p: 1; 2; 3; 4; 5; 10; 32; 100; 316; 1,000; 3,162; 1.1, 1.2, . . . , 1.9; 2.1, 2.2, . . . , 2.9 $\kappa k/n$: 0.0, 0.02, 0.05, 0.1, 0.15, 0.2, . . . , 1.0
- k: $\lfloor n - p \rfloor$, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 32, 100

4. COMPUTATIONAL ERROR DUE TO LIMITED PRECISION

The major computational error is due to the evaluation of $(1 - 1/m)$ in Eq. (7). For example, if $m = 10^6$, we need better resolution than 10^{-6}. However, it is shown in [15] that the number of valid digits required by Eq. (7) is roughly proportional to $\log_{10}(m)$, while that required by Eq. (6) using the Gamma function is proportional to $\log_{10}(mn \ln(n))$ for the same precision in the result. In the exhaustive calculation using a DEC System 20 with single-precision variables, we obtained a maximum error of 0.2% when $m = 10^6$ over the range of variables shown in Section 3.

5. COMMENTS ON RELATED WORK

Formulas essentially identical to Cardenas's and Yao's formulas were derived independently by Waters and Karayiannis [11–13]. Waters summarized three related formulas [13], which are

$$b_{WAT1}(m, p, k) = m[1 - (1 - k/n)^p] \tag{13}$$

$$b_{WAT2}(m, p, k) = m[1 - (1 - p/n)^k] \tag{14}$$

$$b_{WAT3}(m, p, k) = m\left[1 - \prod_{i=1}^{k} (1 - p)/(n - i + 1)\right] \tag{15}$$

Eq. (14) and Eq. (15) are identical to Eq. (5) and Eq. (3), respectively. Eq. (13) was derived in [11, 12], as follows:

RHR = number of distinct records hit/total number of records in file
= probability that any particular record is hit
= k/n.

$\therefore$ $1 - RHR$ = probability that any particular record is not hit.

$\therefore$ $(1 - RHR)^p$ = probability that any particular block is not hit.

$\therefore$ $1 - (1 - RHR)^p$ = probability that any particular block is hit.

Subsequently, during one of Waters' lectures, Karayiannis (then a student) suggested that Eq. (13) was incorrect, pointing out that Eq. (13) gives an incorrect result where $m = 1$ (correct results is $b(m, p, k) = 1$ if $k > 0$). He further suggested Eq. (14) as an alternative formula. Later, Waters [3] announced that Eq. (13) and the above derivation were incorrect and instead suggested Eq. (15) as an alternative formula.

We note, however, that the derivation of Eq. (13) is correct if we make the *independence* assumption in calculating the probability that any particular block will not be hit. More rigorous derivation should use *conditional probability*, since the events of each record's being hit are not mutually probabilistically independent.

Note that if we interchange p and k, Eq. (12) bears the same relationship with Eq. (13) as Eq. (3) does with Eq. (14). In this sense, Eq. (12) and Eq. (13) are a *dual* of Eq. (3) and Eq. (14).

It was observed that Eq. (14) yields a good approximation when $k \ll n(\kappa \ll 1)$ or $p \gg 1$ [17]. Hence, Eq. (13) will give a good approximation when $p \ll n(m \gg 1)$ or $k \gg 1$ by duality. This means that one formula will result in a good approximation when its counterpart yields a poor one, and vice versa. Therefore, an obvious alternative approach to the one presented in this paper is to combine these two formulas in such a way as to get a good approximation over the entire range. As an example, we suggest here the following formula:

$$\begin{aligned} b_{W2}(m, p, k) &= \max\{b_{WAT1}(m, p, k), b_{WAT2}(m, p, k)\} \\ &= \max\{m[1 - (1 - k/n)^p], \\ &\qquad m[1 - (1 - p/n)^k]\} \end{aligned} \tag{16}$$

where 'max' represents the maximum of the two arguments. This equation will be a good approximation, since either formula always produces a value smaller than the exact formula. (This can be easily understood by examining the underlying assumptions.)

6. APPLICATION

An implicit assumption made throughout the development of all the formulas is that a block is accessed no more than once. We encounter this situation in practice when the records selected are accessed in TID (tuple identifier or database key) order.

Two typical applications of these formulas are in query optimization [18] and physical-database design [5, 14]. The formulas are used to estimate the number of block accesses, which is an important measure of cost. They are also used to estimate the number of logical groups of records selected [14]. A *logical group* is a set of records grouped according to certain criteria—for example, common possession of the same value on a certain field. Close estimation of the number of logical groups selected is necessary in analyzing the interactions among relations in the design of a physical database. In this application, we are very likely to have low grouping factors (number of records in a group) that correspond to the blocking factors of a block (physical group). For example, we have a grouping factor of 1 when the records are grouped according to the values of a key field.

Although Cardenas's formula, currently used in System R [1], gives a reasonable approximation in many cases, it is especially prone to failure at low blocking factors (particularly when $p < 10$). Eq. (7) proves to be very useful in these situations.

7. CONCLUSION

A closed, noniterative formula for estimating the number of block accesses was introduced. It improves Yao's exact formula in the sense that it significantly reduces the computation time by eliminating the iterative loop, while providing a practically negligible deviation (maximum error = 3.7%) from the exact formula over the entire range of variables involved. The computational error due to the machine's limited precision has been greatly reduced as compared with a method using the gamma function based on Yao's formula. It significantly improves Cardenas's earlier formula, which has a maximum error of $e^{-1} = 36.8\%$ (at $p = 1$).

REFERENCES

1. Astrahan, M.M. et al. Performance of the System R access path selection mechanism. Information Processing, IFIP, North-Holland Publishing Co., 1980, pp. 487–491.
2. Berstein, P.A., et al. Query processing in a system for distributed databases (SDD-1). *ACM Trans. Database Syst.* 6, 4 (December 1981), 602–625.
3. Cardenas, A.F. Analysis and performance of inverted database structures. *Comm. ACM 18*, 5 (May 1975), 253–263.
4. DEC system-10/DECSYSTEM-20 Hardware Reference Manual-Central Processor. Digital Equipment Corporation, 1978.
5. Hammer, M. and Chan, A. Index selection in a self-adaptive database management system. Proc. Intl. Conf. on Management of Data, Washington, DC, ACM SIGMOD, June, 1976, pp. 1–8.
6. IBM, *System 360 Scientific Subroutine Package*. 1970.
7. Luk, W.S. On Estimating Block Accesses in Database Organizations. Comm. ACM, this issue.
8. Rothnie, J.B. and Lozano, T. Attribute-based file organization in a paged memory environment. *Comm. ACM 17*, 2 (February 1974), 63–69.
9. Severance, D.G. Some generalized modeling structures for use in design of file organizations. Ph.D. Thesis, University of Michigan, Ann Arbor, Michigan, 1972.
10. Siler, K.F. A stochastic evaluation model for database organizations in data retrival systems. *Comm. ACM 19*, 2 (February 1976), 84–95.
11. Waters, S.J. File design fallacies. *Comput. J. 15*,[1] (1972), 1–4.
12. Waters, S.J. Estimating magnetic disc seeks. *Comput. J. 18*, 1 (1975), 12–17.
13. Waters, S.J. Hit ratios. *Comput. J. 19*,[1] (1976), 21–24.
14. Whang, K., Wiederhold, G., and Sagalowicz, D. Separability: an approach to physical database design. Proc. Intl. Conf. on Very Large Databases, Cannes, France, IEEE, September, 1981, pp. 320–332.
15. Whang, K., Wiederhold, G., Sagalowicz, D. Separability as an approach to physical database design. Tech. Rept. STAN-CS-81-898, Stanford University, October 1981 (also numbered as CSL TR-222).
16. Wiederhold, G. Database Design. McGraw-Hill, New York 1977.
17. Yao, S.B. Approximating block accesses in database organizations. *Comm. ACM 20*, 4 (April 1977), 260–261.
18. Yao, S.B. Optimization of query evaluation algorithm. *ACM Trans. Database Syst.* 4, 2 (June 1979), 133–155.
19. Yue, P.C. and Wong, C.K. Storage cost considerations in secondary index selection. *Int. J. Comput. Inf. Sci.* 4, 4 (1975), 307–327.

CR Categories and Subject Descriptors: H.2.2 [**Database Management**]: Physical Design—*access methods*; H.3.2 [**Information Storage and Retrieval**]: Information Storage—*file organization*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*retrieval models, selection process*

General Terms: Performance

Additional Key Words and Phrases: database, number of block accesses, blocking factor, page, query-processing, database performance, information processing, database access cost, records, conditional probability

Received 6/81; revised 10/82; accepted 11/82