A REFINED ESTIMATE FOR LOGICAL RECORD ACCESS IN DATA BASE DESIGN

T. C. WANG[*]

ABSTRACT

A statistical estimate is obtained for the total number of nodes in a three level tree. The number of nodes at each level is randomly distributed with certain pattern of distribution. Then the result is applied to the problem of estimating the LRA (logical record access) in data base design in a more refined way than the expected values used by Teory, etal.

GENERAL MODEL AND PROBLEM MOTIVATION

A three level tree is depicted as figure 1. The first level has m nodes, where m is random. Under each first level node, there are n second level nodes, where n is random. Under each second level node, there are X-1 third level nodes, where X is a random variable. The problem is to find an estimate for the total number of nodes in the tree in terms of the moments of m, X and n. The following three theorems will solve this problem.

Notation:

Random Variable (R.V.)X, distribution function $F_{X.}$. Moments of X in our context refer to the first three moments of X about the mean, $m_X^i = i\text{th}$ moment, i=1, 2, 3. iid means identically, independently distributed.

u_X = mean of X,

σ_X^2 = variance of X,

γ_X = 3rd moment of X about mean,

M_X = moment generating function of X.

Theorem 1:

Xi iid F_X i = 1 ... n,

n random variable, X and n independent,

$Y = X_1 + \ldots + X_n$,

Then $u_y = u_n u_x$

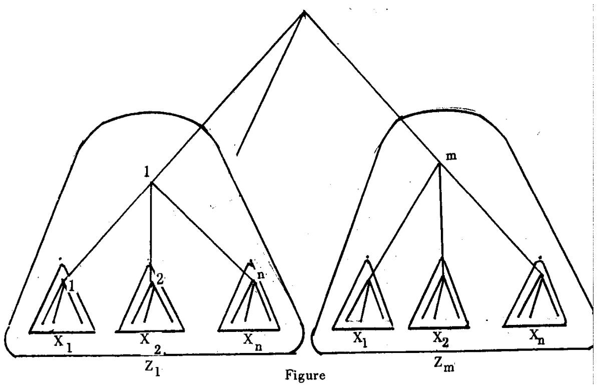
$$\sigma_y^2 = \sigma_n^2 u_x^2 + \sigma_x^2 u_n \quad ,$$

$$\gamma_y = \gamma_n u_x^3 + 3 \sigma_x^2 u_n \sigma_n^2 + u_n \gamma_x.$$

[*] National United University

Proof: $m_X^i = \frac{d(i)}{dt} \ln M_X (t) \Big|_{t=0}$

and $\ln (M_y (t)) = \ln (M_n (\ln M_X (t)))$.



Figure

Theorem 2:

Same assumption as Theorem 1, and Z = Y + 1, Z_j iid F_Zj =1...m, m random variable, indep. of Y, W= Z_1 + ... + Z_m ,

Then moments of W can be expressed as functions of moments of m, X, and n.

Proof: Apply **Theorem 1** twice.

Theorem 3:

For an arbitrary $\varepsilon > 0$

$\Pr (W > x_w^{\varepsilon}) \simeq \varepsilon$

Where $x_w^{\varepsilon} = \mu_w + \sigma_w^2 (y_\varepsilon + \frac{n_w}{6} (y_\varepsilon^2 - 1))$

$n_w = \frac{r_w}{\sigma_w^3}$

$y_\varepsilon = \Phi^{-1}(1 - \varepsilon)$ Φ = Normal Distrib.

Proof: Use Edgeworth series expansion for F_w. *

APPLICATION TO DATA BASE DESIGN

Teorey (1982)** used logical record accesses (LRA), transport volume (TRVOL), data storage space (DSTOR) and pointer storage space (PTRSTOR) as performance measures for logical data-base-structures. He further suggested: "Designers with more experience may wish to analyze record occurrence distributions instead of expected values in terms of child record occurrences per parent record occurrence". But no specific details of analyzing record occurrence distributions were given. The following intends to derive a statistically sound and easily calculable estimate for the worst case LRA based on the record occurrence and query frequency distributions' moments. The derivation is by no means obvious. The estimate thus derived is claimed to be more refined than using the sample maximum of the record occurrence distributions as illustrated in Teorey's example. The same principle can be used to estimate the worst TRVOL, DSTOR, PTRSTOR, and physical block accesses (PBA) in hierarchical record clustering (Teorey p.216).

Notation:

LRA = Total # of logical accesses, a random variable.
LRA_i =Total # of logical accesses by query i, a r. v.
i= 1 ...M
LRA_j indep. i = 1...M.

We have

(1) $$LRA = \sum_{i=1}^{M} LRA_i$$

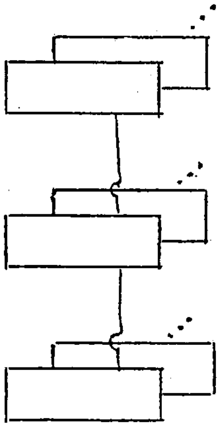
Let q_j= frequency of query i,
LRA_i^o = # of logical accesses by each occurrence of query i, K_i=number of hierarchies passed by query i,

(2) $$LRA_i^o = \sum_{k=1}^{K_i} LRA_i^k$$

Where LRA_i^k = # of logical record accesses along kth hierarchy by an occurrence of query i, indep. over k.

(3) $$LRA_i = LRA_{i1} + \ldots + LRA_{iq_i} \; ,$$
$$LRA_{ij} \text{ iid } F_{LRA_i^o}$$

A typical LRA_i^k can be calculated as follows:
Assume the kth hierarchy passed by one occurrence of query i looks like:



oc_{i1}^k = record occurrence at level 1, a r. v.

oc_{i2}^k = record occurrence at level 2 per level 1 record.

oc_{i3}^k = record occurrence at level 3 per level 2 record.

Then setting m in theorem 2 as oc_{i1}^k, n as oc_{i2}^k , X as $oc_{i3}^k + 1$, $LRA_i^k = W$. A more detailed explanation is given in the Appendix.

Hence moments of LRA_i^k can be expressed as functions of moments of record occurrence distributions in the kth hierarchy accessed by query i.

By (2), LRA_i^{Q}'s moments are functions of moments of record occurrence distributions accessed by query i.

By (3) and Theorem 1, the moments of LRA_i are functions of moments of record occurrences accessed by query i and the frequency of query i.

BY (1), the moments of LRA are functions of moments of record occurrences and frequency of queries.

Note if queries are assumed Poisson, the first three moments are the same.

Applying Theorem 3, we have the following result:

For an arbitrary $\varepsilon > 0$

$$\Pr (LRA > x_{LRA}^{\varepsilon}) \simeq \varepsilon$$

Where $x_{LRA}^{\varepsilon} = \mu_{LRA} + \sigma_{LRA}^{z} (y_\varepsilon + \frac{n_{LRA}}{6} (y_\varepsilon^{z} - 1))$

$$^{n}LRA = \frac{\tau_{LRA}}{\sigma_{LRA}^{3}}$$

$y_\varepsilon = \Phi^{-1} (1 - \varepsilon)$ $\Phi =$ Normal Distrib.

<u>Conclusion</u>:

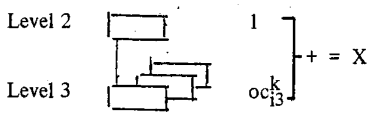
x_{LRA} seems to be a more refined worst LRA estimate with error rate ε.

It is more refined in the sense of using more information contained in the empirical distributions and being associated with a probability statement.

*Beard, R.E. (1969) Risk Theory, Methuen, London
**Teorey, T. and Fry, J. (1982) Design of Database Structures, Prentice Hall, pp. 140-141.

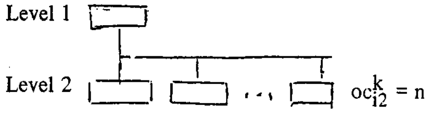
APPENDIX

For each level 2 record occurrence, there are oc_{i3}^k level 3 record occurrences, so there are $X = oc_{i3}^k + 1$ logical record accesses for each access through an occurrence of level 2 record. It is depicted as

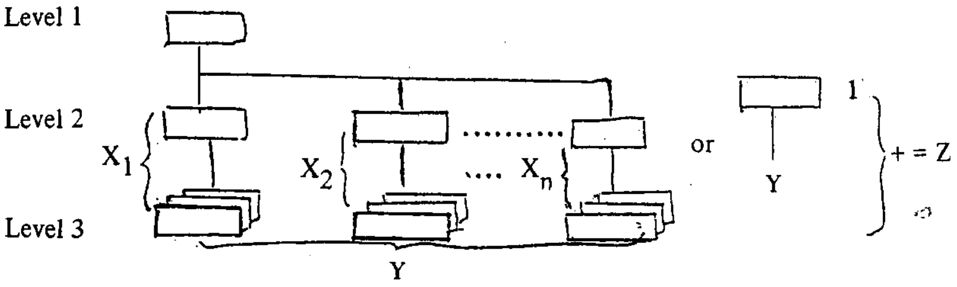


(Figure 1)

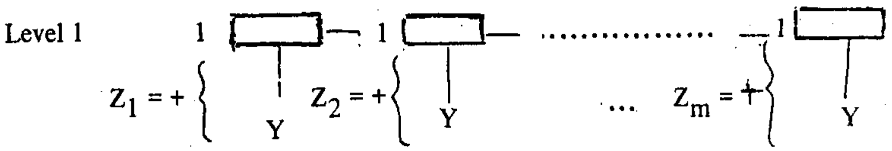
Under a particular occurrence of level 1 record, there are $n = oc_{i2}^k$ level 2 record occurrences, depicted as:



After accessing this particular level 1 record occurrence, the query i would access $X_1+X_2+X_3+...X_n=Y(X_j$ iid X, see Figure 1) level 2 and 3 record occurrences. It is depicted as



Because there are not only 1, but $m = oc^k_{i1}$ level 1 record occurrences, the total number of accesses is the sum of accesses for each particular level 1 record occurrence. If Z =Y+1, $Z_j$iid Z, j=1...m, $LRA^k_i = Z_1 + Z_2 + \ldots +Z_m$. It is depicted as



<u>Example 1</u>:

Teacher
|
Class
|
Student

(score report)
<u>Example 2</u>:

Ins Co Headquarter
|
Agency Director
|
Agency Manager
|
Broker
|
Customer