PARTIAL INFORMATION AND ITS LIMITED UTILITY
- THE CASE OF REORGANIZING LISTS

by

H. Shachnai and M. Hofri

Technical Report #656

November 1990
PARTIAL INFORMATION AND ITS LIMITED UTILITY  
— THE CASE OF REORGANIZING LISTS

Hadas Shachnai and Micha Hofri†

Computer Science Department  
Technion - Israel Institute of Technology  
Haifa 32000, Israel

November 1990

ABSTRACT

1. Introduction and Notation

The structure we consider is a linear list of \( n \) records, \( L = \{R_1, \ldots, R_n\} \). Each record \( R_i \) is uniquely identified by a key \( K_i \), \( 1 \leq i \leq n \).

Requests for the keys are drawn from a multinomial distribution driven by the reference probability vector \( (rpv) \): \( p = (p_1, \ldots, p_n) \). Thus, \( R_i \) may be accessed at any stage with a fixed probability \( p_i \). We assume the independent reference model (irm).

As each of the references requires a sequential search of the list, we let \( C \), the cost of a single access be defined as the number of key comparisons made till reaching the specified record. Under the irm, with a fixed rpv, the average access cost to the list is minimized when permuting the records to the optimal static order: \( R_i \) precedes \( R_j \) whenever \( p_i > p_j \). Doing that requires a complete knowledge of the rpv, or at least of the relative magnitude of the access probabilities. This knowledge is assumed unavailable.

Previous works considered the situation where there is no initial information at all concerning the correct order of the records. Thus, the initial arrangement is chosen randomly (with equal probability) out of all possible permutations, and thereafter the list is constantly reorganized, with the aim of approaching the optimal ordering as the reference sequence grows longer.

A comprehensive survey of many permutation rules suggested for the above model and their analyses appears in [Hester and Hirschberg, 1985].

Recent work [Hofri and Shachnai, 1989] proves the optimality of Counter Scheme (CS) among all deterministic policies. It hinges on the assumption that throughout the reference history, the only criteria for distinguishing between the list records are their relative position at every stage and the history of requests for each.

† Currently at the Department of Computer Science, University of Houston, Houston TX 77204-3475, USA.
Various performance measures were considered for that model. Under a given policy \( H \) and an unknown \( r_p v p \), the following will be used below:

- The average access cost after the \( m \)th reference, \( m \geq 0 \), denoted below as \( C_m(H | p) \).
- The expected access cost in the limiting state: \( C(H | p) \).
- The rate of convergence of \( C_m(H | p) \) to its limiting value, quantified by the Overwork measure [Binner, 1979], which is denoted by \( OW(H | p) \).

Our definition of an optimal policy is natural: it is an admissible policy \( H^* \) that satisfies

\[
C_m(H^* | p) \leq C_m(H | p), \quad \forall m, p, H \in H_{(n=1)}.
\]

provably.

(The requirement expressed here is necessary to exclude the following scenario: for any \( r_p v p \), the optimum is obtained by the static policy that places \( R_i \) in position \( x_i \) when \( p_{x_1} \geq p_{x_2} \geq \ldots \geq p_{x_n} \). This policy is admissible, but — unless we know \( p \) — is not provably optimal). This definition holds for any level of \( a \)-priori information available.

In the present work, we distinguish between degrees of partial \( a \)-priori knowledge and discuss its contribution to the efficiency of the list reorganization process. A list model characterized by no initial information will be of the Null Information (NI) type, while any other model will be of Partial Information (PI).

A useful knowledge may be either the correct relative order of every pair of records in a given sub-list, or the precise values of some of the access probabilities. We concentrate on information of the second kind.

As before, the 'unknown' records will be dynamically reorganized. The others will be kept at their best relative order. The main problem at each stage is merging the two ordered sub-lists into a complete structure so as to achieve a minimal average access cost for subsequent requests.

The issue of dealing with \( a \)-priori information — on any sub-set of the parameters — is common in problems of adaptive control and statistical estimation. The dynamic list model is strongly related to both:

- In the context of adaptive control, it may be viewed as a finite-state system (having \( n! \) states), with unknown transition probabilities. The reorganization process is intended to study the system behavior while trying to minimize gradually the cost of its usage.

Alternatively, a statistical point of view will classify the list reorganization as a ranking problem, where a set of multinomial parameters is to be ordered properly after a finite sequence of independent trials.

The basic approach in both cases is the use of a given parameter values while studying the test.

Some examples may be found in [Bather, 1980] and [Gibbons et al. 1977]. We show below that this approach does not work particularly well in the list model, for certain cost criteria.

In Section 2 we discuss the notion of an optimal policy and define the class of policies which may be applied for the PI model. We then show that none of these rules minimizes the access cost after each reference. (Obviously, more than one rule may minimize the limiting cost).

Section 3 refers to a sub-class of policies based on counters. We define the MLE Rule (MLR), which is analogous to Counter Scheme — which is the optimal strategy for the NI model. We show, that is spite of its usage of extra knowledge, the MLR might be slower than the CS in converging to the same asymptotic average cost. Finally, we point out, that partial information should be used to improve the list order accepted in the NI conditions.
We conclude in section 4 with discussion and a few related open problems.

2. Reordering Methods for the PI Model

Any a priori information on the rpv calls for a basic change in the symmetric approach adopted for the NI model.

We use \( \sigma_m = (\sigma_m(1), \ldots, \sigma_m(n)) \) to denote the list order after the \( m \)-th request, where \( \sigma_m(i) \) is the position of \( R_i \). In an earlier paper, [Hofri and Shachnai, 1989], dealing exclusively in the NI case, we defined the set of key ignoring policies \( H_{KI} \). Intuitively, this simply means that records are only managed by the reorganizing policy according to their role in the reference history. A precise definition of \( H_{KI} \) runs as follows: Let \( \sigma_m \) be a permutation of the records indices, denoting their order following the \( m \)-th reference. Then record \( R_i \) is in position \( \sigma_m(i) \). A policy \( H \) is said to be in \( H_{KI} \) when it satisfies the following constraint: Consider a pair of initial orderings \( \sigma_0^{(1)}, \sigma_0^{(2)} \) and the permutation \( g = \sigma_0^{(2)} \circ \sigma_0^{(1)} \). Let the canonical history vector \( \tilde{I}_m = (\tilde{i}_1, \ldots, \tilde{i}_m) \) denote the history of references to the list in terms of the initial positions of the records. E.g., \( \sigma_m(i) = \sigma_m(\sigma_0(i)^{-1}) \). Then for every history, expressed by the canonical vector \( \tilde{I}_m \), we find

\[
\text{Prob}_H(\sigma_m \circ \tilde{I}_m, \sigma_0^{(1)}) = \text{Prob}_H(\sigma_m \circ \tilde{I}_m, \sigma_0^{(2)}).
\] (2.1)

Where \( \text{Prob}_H(\sigma_m \circ \tilde{I}_m, \sigma_0^{(r)}) \) is the probability of obtaining the specified permutation (here \( \sigma_m \circ \tilde{I}_m \)) for the given initial order and canonical history vector, when policy \( H \) is used. When \( H \) is deterministic this merely says that the effect of \( I_m \) is invariant under changes of the names of the records. Similarly, we say a policy \( H \) is of type \( r-KI \), for \( 2 \leq r \leq n \), if there exists a sub-list \( L' \) of \( r \) records, such that \( H \) uses at each stage a uniform, key-ignoring criterion for determining the relative order of \( R_{i_1}, R_{i_2}, \ldots, R_{i_r} \). 

Let \( H_{r-KI} \) denote the set of \( r-KI \) policies. If there is no initial knowledge on \( r \) of the access probabilities, then any policy \( H \in H_{r-KI} \) may be applied on the list.

We concentrate on PI models in which a sub-group of the access probabilities is known in advance, and where no other information is revealed. Let \( p \) be the reference probability vector (rpv), where \( p_1, \ldots, p_r \) are known.

Without loss of generality, we may assume a remumbering such that

\[
1^{-l} \sum_{i=1}^{l} p_i \geq p_{l+1} \geq \cdots \geq p_n \geq 0.
\] (2.2)

As in the NI model, the initial state of the list is assumed random and uniform over the \( n! \) orderings of \( L \). Thus, the list reorganization process commences after the first request.

Our first claim refers to the way in which partial knowledge disrupts our ability to specify a realizable optimal strategy. It suggests a clear distinction between the state of complete ignorance (the NI model), for which an optimal strategy exists (and is known), and the asymmetric situation embodied by the PI models.

For simplicity, we formulate the following result for the case where \( l = 1 \). With minor changes it holds for any \( 1 \leq l \leq n-2 \).

**Theorem 2.1**: For any list of length \( n \geq 2 \), let \( 1/n < p_1 < 1/2 \), then within the class of \( H_{2-n-KI} \) no single policy minimizes the expected access cost at the \( m \)-th request, \( m > 1 \), for all values of \( p_2, \ldots, p_n \).
Shachnai, Hofri: List-reorganization with partial information...

Proof:
Let $1/n < p_1 < 1/2$ be the known access probability of $R_1$, and

$$S_p = \{ p \mid p_1 = p \}.$$  

Our proof presents a partition of $S_p$ with respect to the optimal policy. That is, distinct subsets in $S_p$ will be shown to exhibit minimal access cost under different realizable policies. In Lemma 2.3 we define a policy $H^*$, which minimizes the average access cost to the list for a particular sub-set of $S_p$. Then, Lemma 2.4 shows the existence of a distinct (realizable) policy $H^{**}$, which does better than $H^*$ for another sub-set of $S_p$. Both $H^*$ and $H^{**}$ are in $H_{s(n-1)}$.

We define $C^{(m)} = (c_1, $ \ldots $, c_n)$ as the frequency vector recorded during the first $m$ references; $c_i$ is the counter of $R_i$. Note, however, that while we shall use the counters vector $C^{(m)}$ in characterizing the various policies, the latter are not necessarily counter-based: they are not barred from using any additional information.

Proposition 2.2: The optimal relative arrangement of $R_2, \ldots, R_n$, with respect to the average access cost after each request, is in decreasing order of the counters $c_2, \ldots, c_n$.

Proof: The claim follows from the composition of the expected cost by summing the probabilities of relative positions over all pairs of records (see Hofri and Shachnai (1989)).

The above result focuses our effort on positioning properly $R_1$ among the other records, which we know how to order optimally at each stage.

Let $\pi_m = (R_{i_1}, \ldots, R_{i_n})$ denote the list order after the $m$th request.

Denote by $V_m$ the set of all $C^{(m)}$'s describing access sequences which consist of two indices only, 1 and $i$, for some $2 \leq i \leq n$ - or the only index $j$, $2 \leq j \leq n$:

$$V_m = \{ C^{(m)}: \exists i \neq 1, c_1 + c_i = m, c_j = 0 \}. \quad (2.3)$$

Lemma 2.3: Assume the $r_{pv}$ has the form

$$p_a = (p, 1-p, 0, \ldots, 0).$$

where $1/n < p_1 = p < 1/2$. For this $r_{pv}$, let $H^*$ be the optimal policy in $H_{s(n-1)}$. Then for all $C^{(m)} \in V_m, H^*$ chooses the ordering $\hat{\pi}_m$:

$$\hat{\pi}_m = (R_1, R_{i_1}, R_{i_2}, \ldots, R_{i_m}), \quad (2.4)$$

where $R_{i_1}, \ldots, R_{i_m}$ are the records not accessed at the first $m$ references.

Proof: Considering the $r_{pv} p_a$, nearly all feasible counter vectors $C^{(m)}$ are in $V_m$, as all references are to either $R_1$ or $R_2$. The only instance of $C^{(m)} \in V_m$ is where $c_1 = m$ and $c_i = 0$ for all $2 \leq i \leq n$.

By its definition, $H^*$ picks the best position for $R_1$, for all possible histories, including those that generate that special case for $C^{(m)}$. Since $p_2 > p_1$, the optimal ordering of the list is $(R_2, R_1, R_{i_1}, \ldots, R_{i_m})$.

For any $C^{(m)} \in V_m$, this order can be accepted using the following rule (remember that $p_1 = p$ is a known quantity):

For any $i, 2 \leq i \leq n$ and $c_i = m$, if

$$c_i (1-p) > p, \quad \frac{m - c_i}{m - c_1} > p, \quad (2.5)$$

then place $R_i$ ahead of $R_1$ in $\pi_m$. 

Technion - Computer Science Department - Technical Report CS0656 - 1990
Hence, for all \( C^{(m)} \in V_m \), \( H^* \) chooses after the \( m \)-th request an order of the form \( \hat{\pi}_m \).

Lemma 2.4: Now consider the rps \( p_b \) in which
\[
p_1 = p, \quad p_2 = \cdots = p_n = \frac{1-p}{n-1},
\]
where as above, \( 1/n < p < V_\infty \).

Let \( H^{**} \) be defined as the optimal policy in \( H_{(n-1)} \) for all \( m \). That is
\[
C_m(H^{**} | p_b) = \min_{H \in H_{(n-1)}} C_m(H | p_b).
\]

Then, if \( C^{(m)} \in V_m \), the policy \( H^{**} \) will not always choose an ordering of the form \( \hat{\pi}_m \) defined above in equation (2.4).

Proof: We know that \( \hat{\pi}_m \) is not the optimal order — but we need to show that a realizable policy is capable of finding this out as well. Assume, by way of contradiction, that whenever \( C^{(m)} \in V_m \), \( H^{**} \) does choose the ordering \( \hat{\pi}_m \). We’ll argue that in some cases it will find a better choice.

The following assumptions will be used:

(i) As no information has been accumulated on \( R_{t_1}, \ldots, R_{t_{n-2}} \), any of the \( (n-2)! \) relative orders of these records may appear in \( \hat{\pi}_m \) with equal probability.

(ii) \( H^{**} \) can use any criterion for positioning \( R_1 \) in the list after the \( m \)-th request, as long as it handles symmetrically the \( n-1 \) unknown records.

Randomizing over all possible counter vectors, and using what we hope is a self-explanatory notation, we write
\[
C_m(H^* | p_b) = \sum_{C^{(m)} \in V_m} C_m(H^{**} | p_b, C^{(m)}) \Pr(C^{(m)}) + \sum_{C^{(m)} \in V_m} \min_{H \in H_{(n-1)}} C_m(H^{**} | p_b, C^{(m)}) \Pr(C^{(m)})
\]
\[= C_1 + \sum_{k=1}^m \Pr(m-k, 0, \ldots, k, \ldots, 0) \left( \frac{1-p}{n-1} + 2p + \sum_{i=1}^k \frac{(1-p)}{n-1} \right) \]

Now, assume that \( H \) is some policy, which places \( R_1 \) in the list the same way as \( H^{**} \) does when \( C^{(m)} \in V_m \), but for \( C^{(m)} \in V_m \) it may choose different placement.

A possible criterion for \( H \) after the \( m \)-th request is to emulate \( CS \) in \( V_m \).

Thus, in that case, \( H \) will choose
\[
\hat{\pi}_m = \begin{cases} \hat{\pi}_m & c_i > c_1 \\ \hat{\pi}_m & c_i < c_1 \\ \hat{\pi}_m \text{ or } \hat{\pi}_m \text{ with probab. } 1/2 & c_i = c_1 \end{cases}
\]
where \( 2 \leq i \leq n \) and
\[
\tilde{\pi}_m = (R_{t_1}, R_{t_2}, \ldots, R_{t_{n-1}}).
\]

It is easy to verify, that for all \( m > 1 \)
\[
C_m(H | p) < C_m(H^* | p),
\]
in contradiction to the optimality of \( H^{**} \).
Hence, the former assumption fails, and $H^{**}$ does not choose an ordering of the form $\hat{n}_m$ for all $c^{(m)} \in V_m$. 

Observe, that Theorem 1 still holds, when we know a-priori that $p_1 = \rho$ and that $R_1$ is to be placed $i$-th in the optimal static arrangement, provided that there is no initial knowledge of the other components of the $r_{pv}$.

We comment, that there are some $PL$ situations, for which an optimal strategy does exist. A trivial case is where $p_1$ is in the range $[\frac{1}{2}, 1]$. Another instance is where the whole $r_{pv}$ is known, up to the permutation of the keys, i.e., all the access probabilities are given, but only $R_1$ can be identified by its access probability. In that situation, the optimal position of $R_1$ after the $m$-th reference, given any ordered counters vector $(c^{(m)}, \ldots, c^{(m)})$ may be determined by a direct computation when averaging over the $n!$ mappings of that vector to the indices $(1, \ldots, n)$. Hence we get an optimal realizable policy.

3. Counter Based Rules

Previous work [Hofri & Shachnai, 1989] discusses the optimality of $CS$ for the $NI$ scenario. It agrees with the statistical observation, that frequency counts are efficient estimates for the access probabilities.

Now we consider $CS$ in the $PI$ situation. As above, we assume that $l$ of the access probabilities are known in advance, for some $1 \leq l \leq n-2$. Likewise, we say that $H \in H_{H(s)}$ is counter based (CB), if it uses only frequency counts for determining the relative order of the $(n-l)$ 'unknown' records.

Following Proposition 2.2, we limit the discussion to those policies, which reorder $R_{i+1}, \ldots, R_n$ after each reference, in non-increasing order of the counters $c_{i+1}, \ldots, c_n$.

One approach to arguing for the optimality of $CS$ (in the $NI$ scenario) observes that it uses the counters to compute $p_l$, where these are the Maximum Likelihood Estimates (MLE) of the access probabilities. Under the $NI$ model,

$$\hat{p}_l = \frac{c_l}{m}. \quad (3.1)$$

Similarly, when considering the $PI$ model, with $l$ of the probabilities known, the $MLR$ maintains the records ordered by the new estimates $\hat{p}_l$'s, where

$$\hat{p}_l = \begin{cases} 
\frac{p_l}{c_l^{(m)} \sum_{j=1}^{l} p_j} & 1 \leq l \leq l \\
\frac{c_l^{(m)} (1 - \sum_{j=1}^{l-1} p_j)}{m - \sum_{j=1}^{l-1} c_j^{(m)}} & l+1 \leq l \leq m, \quad \sum_{j=1}^{l} c_j^{(m)} < m \\
0 & l+1 \leq l \leq n, \quad \sum_{j=1}^{l} c_j^{(m)} = m 
\end{cases} \quad (3.2)$$

Observe, that both of the estimates of $(3.1)$ and $(3.2)$ induce the same relative order of $R_i, R_j$ for $l+1 \leq i, j \leq n$.

A rather surprising distinction between the $MLR$ and the $CS$ may be shown: whenever $p_i > p_j, 1 \leq i, j \leq n$,

$$Pr_{CS} \left[ \sigma_m(i) < \sigma_m(j) \right] \rightarrow 1 \quad (3.3)$$

monotonically, while for any $1 \leq j \leq l$ and $l+1 \leq i \leq n$ the value of
\[ P_{\text{MLR}}(\sigma_m(i) < \sigma_m(j)) \]

is not necessarily monotone in \( m \). It will be seen below that this is equivalent to the statement that the cost under CS can be strictly smaller than under the MLR. We demonstrate this for \( l = 1 \) and a specific class of rpvs.

As both CS and MLR converge to the optimal ordering, a criterion for comparing their transient behaviour may be the overwork measure. We define this measure now.

Assume a temporary renumbering of the records so that their probabilities, now denoted by \( \overline{P}_i \), satisfy \( \overline{P}_1 \geq \overline{P}_2 \cdots \geq \overline{P}_n \). (This may mix the known and unknown components). With this notation, \( C_m(\text{OPT} \mid p) = C(\text{OPT} \mid p) = \sum_{i=1}^{n} \overline{P}_i \).

For an arbitrary policy \( H \) [Bitner, 1979],

\[
C_m(H \mid p) = 1 + \sum_{i=1}^{n} \sum_{j=i+1}^{n} P_{ij}(\sigma_m(j) < \sigma_m(i))
\]

\[
= 1 + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \{ \overline{P}_i P_{ij}(\sigma_m(j) < \sigma_m(i)) + \overline{P}_j (1 - P_{ij}(\sigma_m(j) < \sigma_m(i))) \} \tag{3.4}
\]

\[
= C(\text{OPT} \mid p) + OW_m(H \mid p) , \quad 1 \leq j < n .
\]

where \( OW_m(H \mid p) \) is the contribution of step \( m \) to the overwork measure, such that

\[
OW(H \mid p) = \sum_{m \geq 0} OW_m(H \mid p) . \tag{3.5}
\]

From equation (3.4) we see that \( OW_m(H \mid p) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} (\overline{P}_i - \overline{P}_j) P_{ij}(\sigma_m(j) < \sigma_m(i)) \). Let us introduce further notation: for a fixed \( p \) and any pair of indices \( 1 \leq i < j \leq n \) (such that \( \overline{P}_i > \overline{P}_j \)), define

\[
OW_{ij}(H \mid p) = (\overline{P}_i - \overline{P}_j) \sum_{m \geq 0} P_{ij}(\sigma_m(j) < \sigma_m(i)) . \tag{3.6}
\]

Then,

\[
OW(H \mid p) = \sum_{(1 \leq i, j \leq n : \overline{P}_i > \overline{P}_j)} OW_{ij}(H \mid p) .
\]

Observation: For some \( p \) satisfying \( p = \min \{ p_i \}_{1 \leq i \leq n} \),

\[
OW(\text{MLR} \mid p) > OW(\text{CS} \mid p) . \tag{3.7}
\]

Proof: For all \( 2 \leq i \leq n \), the MLR would assign \( \sigma_m(i) < \sigma_m(1) \), when \( c = c_1 \) satisfies \( [m - c]p / (1 - p) \) \leq c; for all \( 1 \leq m \), hence

\[
P_{\text{MLR}}(\sigma_m(i) < \sigma_m(1)) < 1 - (1 - p_i)^m .
\]

Thus,

\[
OW_{ij}(\text{MLR} \mid p) \geq (p_i - p_j) \sum_{m \geq 0} (1 - p_i)^m \cdot \frac{p_i - p_j}{p_i} . \tag{3.8}
\]

Now, it is shown in [Hofri and Shachnai, 1988], that

\[
OW_{ij}(\text{CS} \mid p) \leq \frac{p_i + p_j}{2(p_i - p_j)} , \quad \text{for all } 1 \leq i, j \leq n , \text{where } p_i > p_j .
\]

It only remains now to observe that there are values of \( p_i > p \), such that
For example, let $p_1 = r_p$, then we only need $r > (3 + \sqrt{17})/2 = 4.56$ for inequality (3.9) to hold. Hence, if $p_i > p$ for all $2 \leq i \leq n$, and they satisfy relation (3.9), then

$$O_{\bar{W}_i}(MLR | p) > O_{\bar{W}_i}(CS | p).$$

And since we have

$$O_{\bar{W}_i}(MLR | p) = O_{\bar{W}_i}(CS | p),$$

for all $2 \leq i, j \leq n$, we get the inequality (3.7).

The last result, together with equation (3.4) imply that, in some cases, although $p_1$ is known, we could do better using the estimate $\bar{p}_1$ for positioning $R_1$ in the list, rather than the actual value of $p_1$.

Now, we have shown, that the presence of partial information raises problems in choosing a reorganization rule which is globally optimal. Yet, a proper use of the knowledge of $p_1$, in conjunction with the CS, can improve the ordering of the records, as suggested by the following result. This is a simple generalization of the observation that if we knew $p_1$ to exceed one half, we would place $R_1$ first, since that value must be the largest.

Let $V = (v_1, \ldots, v_n)$ be a vector of $n$ components, where $v_i \in \Omega$, and define further

$$V_{-i} = (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n)$$

$$\bar{V} = (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_n),$$

such that $v_i$ is omitted.

Then $V_{-i}$ represents the sorted vector, when $v_i$ is omitted.

Assume a list of $n$ records, such that $1 \geq p_1 \geq \cdots \geq p_n \geq 0$.

**Lemma 3.1:** For any ordered frequency vector $\bar{C}(m)$, the function

$$f(k) = \sum_{i=k}^{n} p_i P_{c} (C_m(i) = k | \bar{C}(m)), \quad 1 \leq k \leq n$$

is monotone decreasing in $k$.

The proof is quite simple but requires surprisingly heavy notation, and is therefore omitted. We note however that it is a special case of a more general property: if the real sequence $(a_i)$ is monotonically non-increasing, and $B$ a matrix with elements that satisfy $i < j \Rightarrow b_{ij} > b_{ji}$, then $\sum_i a_i \sum_j (b_{ij} - b_{ji}) \geq 0$.

Assume as before that $p_1 = p$ is known. Let $c \{n\} = c_1$ denote the frequency count of $R_1$ after $m$ requests. We use $\bar{C}_m$ to describe the relative order of $R_2, \ldots, R_n$ in a list of $n-1$ locations, $1, \ldots, n-1$.

**Corollary 3.2:** If $p=p\{j\}$ for some $1 \leq j \leq n$, then for any ordered frequency vector $\bar{C}_m(1)$, the function

$$f_{-k}(i) = \sum_{j=2}^{n} p_j P_{c} (\bar{C}_m(i) = k | \bar{C}_{-k}(m)),$$

is monotonically decreasing in $k$, $\forall 1 \leq k \leq n-1$.

Let $\sigma_{H}^m(i)$ denote the position of $R_i$ under the policy $H$ after the $m$th reference to the list.

**Theorem 3.3:** Let $H \in H_{|n-1|}$ be a CB policy, which reorders $R_2, \ldots, R_n$ by CS. If $1 < \frac{1-p}{p} = l^* < n$ and $H_2$ satisfies:

(i) $\sigma_{H}^m(i) = \min(\sigma_{H}^m(1), l^*)$
(ii) \( \sigma_{m}^{H}(i) < \sigma_{m}^{H}(j) \Rightarrow \sigma_{m}^{H}(i) < \sigma_{m}^{H}(j) \), \( \forall 2 \leq i, j \leq n \)

then for all \( m \geq 1 \)

\[ C_m(H_2 | p) \leq C_m(H_1 | p) \]

Proof: It is sufficient to show, that for all \( C^{(m)} \)

\[ C_m(H_2 | p , C^{(m)}) \leq C_m(H_1 | p , C^{(m)}) \]

The first step is showing, that for any given \( C^{(m)} \) and \( k > l^* \), if \( H_b, H_e \in H_{(k-1)} \) satisfy

\[ \sigma_{m}^{H}(1) = k \quad \text{and} \quad \sigma_{m}^{H}(1) = k-1 \]

with the same relative order of \( R_1, \ldots, R_n \), then

\[ C_m(H | p , C^{(m)}) \leq C_m(H_1 | p , C^{(m)}) \]

By definition,

\[ C_m(H_1 | p , C^{(m)}) - C_m(H | p , C^{(m)}) = p - f_{-1}(k-1) \]

where \( f_{-1}(k) \) is defined in (3.10).

The proof is by way of contradiction. Assume, that

\[ f_{-1}(k-1) > p \quad (3.11) \]

Then, by corollary 3.2, for all \( 1 \leq l \leq k-2 \)

\[ f_{-1}(l) > p \quad (3.12) \]

Now,

\[ \sum_{l=1}^{k-1} f_{-1}(l) < 1-p \quad (3.13) \]

Using (3.12),

\[ \sum_{l=1}^{k-1} f_{-1}(l) > (k-1)p \]

But as \( 1-p < l^* p < (k-1)p \), we get

\[ \sum_{l=1}^{k-1} f_{-1}(l) > 1-p \]

which contradicts (3.13).

Hence (3.11) is false and for any \( C^{(m)} \),

\[ C_m(H | p , C^{(m)}) \leq C_m(H_1 | p , C^{(m)}) \]

and this holds for all \( l^* < k \leq n \), thus

\[ C_m(H_1 | p , C^{(m)}) \leq C_m(H_2 | p , C^{(m)}) \]

and

\[ C_m(H_1 | p ) \leq C_m(H_2 | p ) \]

\( \square \)
2. Concluding Remarks

We have studied a variation of the classical dynamic list model and have shown, that frequency counts may be of limited merit when coupled with partial information. However, they are still the best choice for handling the unknown components of the structure.

Though Theorem 1 considers a model in which the number of records is fixed, it may be easily generalized for a list of varying length:

A fortiori, there is no optimal rule for placing a new record \( R_{n+1} \), with the known access probability \( p_{n+1} \), in a model which preserves ratios between old access probabilities, i.e. if \( R_{n+1} \) is inserted in the \( m \)th reference, then

\[
\frac{p_i^{(m)}}{p_j^{(m)}} = \frac{p_i^{(m-1)}}{p_j^{(m-1)}} \quad \forall \quad 1 \leq i, j \leq n
\]

(We assume \( \sum_{i=1}^{n+1} p_i^{(m)} = 1 \quad \forall \quad m \geq 1 \)).

This holds when trying to minimize the expected access cost at each reference. Obviously, the difficulty only arises for the transient analysis: the placement chosen for \( R_{n+1} \) in the list, following its insertion, will not affect the asymptotic optimality of the rule applied, as long as the other access probabilities are eventually known (and used to construct the best ordering).

Some other models of asymmetry call for inspection:

- Consider a list implementation of a dictionary, with unknown \( r_{p,v} \), on which searches, insertions and deletions of records are permitted; no extra information is given on the access probabilities.

Is there an optimal strategy for dynamic reorganization of such a list?

Consider a merge of two lists, where each has already benefited from some reference history of its own. How should the merge be performed to use this information to best advantage?

We conclude with the comment, that a result such as presented in Theorem 2 may not be achieved for other models of self-organizing sequential search. We recall the dynamic path tables described in Topkis (1986):

Each path \( i \) has a fixed but unknown failure probability \( p_i \), independently of all other path failures. When trying to route a new message, the table is scanned from the top, till the first success (or exhaustion of all paths). Then a new permutation of the paths is chosen. Clearly, knowing any sub-set of these failure probabilities will not reduce in advance the range of locations among which those paths are to be positioned, due to the mutual independence between the \( p_i \)'s (in particular, we have no constraint on their sum).

REFERENCES


Hofri M. and Shachnai H., on the Optimality of Counter Scheme for Dynamic Linear Lists. The Department of Computer Science, Purdue University, CSD-TR-927, November 1989.
