Regular Paper

A New Analysis of Hashing Algorithm for External Searching

RYOZO NAKAMURA,† NINGPING SUN†† and TAKUO NAKASHIMA†

Two mathematical analyses are proposed to evaluate exactly the number of accesses of the separate chaining method for external searching on secondary storage devices in consideration of the frequency of access on each key. The first one is compared to the traditional one based on the Knuth's model, and the second one is a new analysis based on the AHU (Aho, Hopcroft and Ullman) model. The evaluation formulae derived from the proposed analyses can exactly evaluate the average and variance of the search cost in conformity with any probability distribution of the frequency of access on a key, and then under the assumption that the frequency of access is uniform these formulae can be represented concisely and approximately by a function of the load factor and the bucket size.

1. Introduction

Hashing is an important technique widely used to provide fast access to information stored on external storage devices as well as on main memory. Hashing for external searching allows the records to be stored in a potentially unlimited space of storage, and therefore it is possible to place the vast quantities of records without the limit on the number of records. The separate chaining technique lends itself well to external searching on direct-access storage devices such as disks.

Time required to solve a problem is one of the most important measures in evaluating an algorithm. Hashing requires the time, in the worst case, to be proportional to the number of records for operations (INSERT, DELETE and MEMBER). In spite of this, on average it takes constant time for each operation. The search cost is usually defined as the product of the number of probes and the frequency of access on each key. If the frequency of access on a key is uniform, the average search cost of the separate chaining technique is independent of the order of the insertion. However, if the frequency of access is not uniform, the inserting order then plays a crucial role.

The search cost considering the frequency of access on an individual key had never been analyzed[2),3)]. Considering the frequency of access on a key for internal searching, we have been mathematically analyzing the search cost of the hashing method [4)], the binary search tree technique[5)], etc. For the separate chaining algorithm of external searching, the traditional analysis has been derived by Knuth in conformity to the assumption that all keys are uniformly accessed[2)].

In this paper, the average and variance of the search cost of the separate chaining technique for external searching is analyzed in consideration of the frequency of access on keys. In our analysis, it is important to clarify the relationship between the inserting order and the locating position of keys. The evaluation formulae of the search cost are derived from the concrete probability distribution of the number of accesses, which can exactly evaluate the search cost with the arbitrary probability distribution of the frequency of access on a key. Under the assumption that the frequency of access is uniform, these formulae can be represented concisely and approximately by a function of the load factor and the bucket size, and compared to the traditional ones. The proposed analysis is shown to be both accurate and appropriate in evaluating the average search cost of the separate chaining method for external searching.

2. Basic Concept of Hashing for External Searching

In the basic data structure of separate chaining technique, the hash table is indexed by the numbers $0, 1, 2, \ldots, M-1$ according to its positions on the table of size M, and it contains the headers of M linked lists. The elements of the i-th list are a set of records whose value of hash function $h(x)$ is equal to i, namely, where the elements have the same hash address i.

We assume that N keys (records) are uniformly mapped into the hash table of size M by

† Department of Computer Science, Faculty of Engineering, Kumamoto University
†† Graduate School of Science and Technology, Kumamoto University

a hash function, and each of the M^N possible hash sequences $a_1, a_2, \ldots, a_N$ $(0 \leq a_j < M)$ is equal likely, where a_j denotes the hash address of the j-th key to be inserted into the table.

Let P_{Nk} be the probability that the number of keys on any list is equal to k $(k = 0, 1, 2, \ldots, N)$. Therefore, P_{Nk} is the binomial probability as follows,

$$P_{Nk} = \binom{N}{k} \left(\frac{1}{M}\right)^k \left(1 - \frac{1}{M}\right)^{N-k}, \quad (1)$$

and its generating function $P(Z)$ is given by

$$P(Z) \doteq \left(1 + \frac{Z-1}{M}\right)^N. \quad (2)$$

In the hashing method for external searching, the records with the same hash address usually are grouped into buckets containing b records each and linked together. There are two models for external searching using the separate chaining technique. One is Knuth's model, in which the average number of accesses was analyzed[2]. In this model, there is only one bucket whose size is b in each list. If more than b keys have the same hash address, a link of "overflow" records, linked together in a record unit, will be inserted at the end of the bucket. The other model is the abbreviated AHU model which was described by Aho, Hopcroft and Ullman[3]. The major difference between Knuth's model and AHU model is that in AHU model an appropriate number of buckets is chained together in a linked list.

We derive the evaluation formulae of the search cost for these two models. To compromise the discrepancies of the terms used in these two models, the ones in Ref. 2) are used in the rest of this paper, and α and b denote the load factor $(\alpha = N/(Mb))$ and the bucket size respectively.

3. Traditional Analysis

The traditional analysis of the separate chaining technique for external searching was derived by Knuth under the assumption that all keys are uniformly accessed[2].

In the Knuth's model, the keys with the same hash address are inserted in order from the head of a bucket. If more than b keys fall into the same bucket, the "overflow" keys, which have been linked together in a record unit, are inserted into a special overflow area. During searching, the keys in a bucket will be accessed by a bucket unit, while the keys in the special overflow area will be accessed by a record unit.

The evaluation formula of the average number of accesses in a successful search was derived as (3),

$$\begin{aligned} S_N &= 1 + \frac{M}{N} \sum_{k>b}^{N} \binom{k-b+1}{2} P_{Nk} \\ &\doteq 1 + \frac{1}{2} e^{-\alpha b} (\alpha b)^b b!^{-1} (\alpha b - b + 2 \\ &\quad + (\alpha^2 b - 2\alpha(b-1) + b - 1) R(\alpha, b)) \\ &= 1 + (1 - b(1-\alpha)/2) t_b(\alpha) \\ &\quad + e^{-\alpha b} (\alpha b)^b R(\alpha, b)/2b!, \end{aligned} \quad (3)$$

where the Poisson approximation is used instead of the exact probability P_{Nk} as mentioned below,

$$\begin{aligned} P_{Nk} &= \binom{N}{k} \left(\frac{1}{M}\right)^k \left(1 - \frac{1}{M}\right)^{N-k} \\ &= \frac{N}{M} \cdot \frac{N-1}{M} \cdots \frac{N-k+1}{M} \left(1 - \frac{1}{M}\right)^N \\ &\quad \times \left(1 - \frac{1}{M}\right)^{-k} \frac{1}{k!} \\ &\doteq \frac{e^{-\alpha b} (\alpha b)^k}{k!}. \end{aligned}$$

The above approximation is valid as N and M become large enough[6]. And, functions $t_b(\alpha)$ and $R(\alpha, b)$ are defined by formulae

$$t_b(\alpha) = e^{-\alpha b} \left(\frac{(\alpha b)^b}{(b+1)!} + \frac{2(\alpha b)^{b+1}}{(b+2)!} + \frac{3(\alpha b)^{b+2}}{(b+3)!} + \cdots \right) \quad (4)$$

and

$$R(\alpha, b) = \frac{b}{b+1} + \frac{\alpha b^2}{(b+1)(b+2)} + \frac{\alpha^2 b^3}{(b+1)(b+2)(b+3)} + \cdots. \quad (5)$$

Here, the relationship between $t_b(\alpha)$ and $R(\alpha, b)$ is described as follows,

$$t_b(\alpha) = e^{-\alpha b} (\alpha b)^b (1 - (1-\alpha) R(\alpha, b))/b!. \quad (6)$$

In addition, the following approximation also is tenable,

$$\begin{aligned} R_\alpha(b) &= 1 + \left(\frac{b}{b+1}\right)\alpha \\ &\quad + \left(\frac{b}{b+1}\right)\left(\frac{b}{b+2}\right)\alpha^2 + \cdots \\ &= \frac{1}{1-\alpha} + \frac{\alpha}{(1-\alpha)^3 b} + O(b^{-2}), \\ &\qquad (\alpha < 1). \end{aligned} \quad (7)$$

Therefore, the relationship between (5) and (7) can be expressed as follows,

$$R(\alpha, b) = (R_\alpha(b) - 1)/\alpha \quad (8)$$

and

$$R(\alpha, b) = \frac{1}{1-\alpha} - \frac{1}{(1-\alpha)^3 b} + O(b^{-2}). \quad (9)$$

The following variance evaluation formula of

the number of accesses has been derived as well,

$$
\begin{aligned}
V_N &= \frac{\sum \binom{N}{k_1,\cdots,k_M}}{M^N} \\
&\quad \times \left\{ \frac{\binom{k_1-b+1}{2}+\cdots+\binom{k_M-b+1}{2}}{N} \right\}^2 \\
&\quad -(S_N-1)^2 \\
&= \frac{1}{M^N N^2}\Bigg\{ \{M(M-1)\sum \binom{N}{k_1,\cdots,k_M} \\
&\quad \times \binom{k_1-b+1}{2}\binom{k_2-b+1}{2} \\
&\quad \times (M-2)^{N-k_1-k_2} \\
&\quad + M\sum \binom{N}{k_1,\cdots,k_M}\binom{k_1-b+1}{2}^2 \\
&\quad \times (M-1)^{N-k_1}\Bigg\} -(S_N-1)^2 \\
&= \frac{1}{M^N N^2}\Bigg\{ M(M-1) \\
&\quad \times \sum_{k_1>b}\binom{N}{k_1}\binom{k_1-b+1}{2} \\
&\quad \times \sum_{k_2>b}\binom{N-k_1}{k_2}\binom{k_2-b+1}{2} \\
&\quad \times (M-2)^{N-k_1-k_2} \\
&\quad + M\sum_{k_1>b}\binom{N}{k_1}\binom{k_1-b+1}{2}^2 \\
&\quad \times (M-1)^{N-k_1}\Bigg\} -(S_N-1)^2. \qquad (10)
\end{aligned}
$$

The traditional analysis mentioned above is a straightforward modification of the analysis of the separate chaining technique for internal searching[2], however, Knuth's analysis for internal hashing is criticized to be incorrect[4]. In the traditional analysis, provided that all of the N keys are equally scattered over the M lists, the total number of probes to find all keys on M lists was counted and divided by N, namely, the random variable was represented by the number of probes per key. Here its probability distribution is the same as the probability of Maxwell-Boltzmann statistics[6]. Besides this, the matter that for a successful search only lists with more than one key in a list are probed was not regarded. Therefore, it is obvious that the traditional analysis of the hashing technique for external searching can not evaluate correctly the actual behavior.

4. Proposed Analysis

It is necessary to clarify the relationship between the inserting order of a key and its locating position in a list for constructing a model in consideration of the frequency of access on keys. In the case that keys are inserted at the tail of a list, the probability that the i-th key inserted will be located in the j-th position from the head of a list with k keys is

$$\binom{i-1}{j-1}\binom{N-i}{k-j}\Big/\binom{N-1}{k-1}.$$

In the analysis of the search cost, let ρ_i be the probability that the i-th key inserted will be retrieved, i.e., ρ_i is the probability of the frequency of access on the i-th key, and let γ_{kj} be the probability that the j-th key from the head of a list with k keys will be probed.

The probability γ_{kj} can be expressed as follows,

$$\gamma_{kj} = \sum_{i=1}^{N}\binom{i-1}{j-1}\binom{N-i}{k-j}\rho_i\Big/\binom{N-1}{k-1} \qquad (11)$$

where $i \geq j \geq 1$.

In this section, we present two analyses, one for the Knuth's model, and the other for the AHU model.

4.1 Proposed Analysis for the Knuth's Model

We suppose the ordinal number of bucket linked into any list to be 1, and ordinal numbers of keys in the special overflow area to be 2, 3, ..., sequentially. The probability of access to the key whose ordinal number is h, namely, q_{Nh} $(h = 0, 1, 2, \ldots, N-b+1)$ then becomes (12),

$$
q_{Nh} = \begin{cases} P_{N0} & (h=0) \\ \sum_{i=1}^{b}\sum_{j=i}^{N}\gamma_{ji}P_{Nj} & (h=1) \\ \sum_{j=h+b-1}^{N}\gamma_{j\ h+b-1}P_{Nj} & (h>1). \end{cases} \qquad (12)
$$

Here, we have

$$\sum_{h=0}^{N-b+1} q_{Nh} = 1.$$

Now we introduce the following condition into our analysis: for a successful searching only lists with more than one key per list are probed. If we take the number of accesses h as the random variable of probability, then the average and variance of the search cost can be expressed as (13) and (14),

$$S_N = \sum_{h=1}^{N-b+1} hq_{Nh} \Big/ \sum_{k=1}^{N} P_{Nk}$$
$$= \left\{ \sum_{i=1}^{b} \sum_{j=i}^{N} \gamma_{ji} P_{Nj} + \sum_{h=2}^{N-b+1} h \sum_{j=h+b-1}^{N} \gamma_{j\ h+b-1} P_{Nj} \right\} \Big/ \sum_{k=1}^{N} P_{Nk} \quad (13)$$

and

$$V_N = \left\{ \sum_{i=1}^{b} \sum_{j=i}^{N} \gamma_{ji} P_{Nj} + \sum_{h=2}^{N-b+1} h^2 \sum_{j=h+b-1}^{N} \gamma_{j\ h+b-1} P_{Nj} \right\} \Big/ \sum_{k=1}^{N} P_{Nk} - S_N^2. \quad (14)$$

The evaluation formulae (13) and (14) can exactly evaluate the average and variance of the search cost with any probability distribution of the frequency of access based on the Knuth's model.

The proposed analysis is compared to the traditional one under the assumption that the probability of the frequency of access on each key is equal likely. From this assumption, ρ_i becomes $1/N$ $(i = 1, 2, \ldots, N)$ independently, and then the probability γ_{kj} becomes $1/k$. Thus, we can derive the following evaluation formulae of S_N and V_N,

$$S_N = \left\{ \sum_{i=1}^{b} \sum_{j=i}^{N} \frac{1}{j} P_{Nj} + \sum_{h=2}^{N-b+1} h \sum_{j=h+b-1}^{N} \frac{1}{j} P_{Nj} \right\} \Big/ \sum_{k=1}^{N} P_{Nk}$$
$$= \left\{ P_{N1} + P_{N2} + \cdots + P_{NN} + \frac{P_{N\ b+1}}{b+1} + \cdots + \frac{(N-b)(N-b+1)P_{NN}}{2N} \right\} \Big/ \sum_{k=1}^{N} P_{Nk}$$
$$= 1 + \sum_{k=b+1}^{N} \frac{(k-b)(k-b+1)P_{Nk}}{2k} \Big/ \sum_{k=1}^{N} P_{Nk}$$

and

$$V_N = 1 + \sum_{k=b+1}^{N} P_{Nk} \times \frac{(k-b+1)(k-b+2)(2(k-b)+3) - 6(k-b) - 6}{6k} \Big/ \sum_{k=1}^{N} P_{Nk} - S_N^2,$$

where the binomial probability P_{Nk} can be approximated by the Poisson distribution in the same way as Knuth's analysis, i.e.,

$$P_{Nk} \doteq e^{-\alpha b} (\alpha b)^k / k!.$$

We are now in a position to approximate the average number of accesses by Poisson distribution.

$$S_N = 1 + \left\{ \sum_{k=b+1}^{N} (kP_{Nk} + (1-2b)P_{Nk} + b(b-1)P_{Nk}/k) \right\} \Big/ 2 \sum_{k=1}^{N} P_{Nk} \quad (15)$$

In order to realize the approximation, the S_N represented in (15) is transformed as following steps. Moreover, for convenience, we denote the load factor, the bucket size and the function with α, b and $R(\alpha, b)$ respectively. As step 1, the first term of (15) can be approximated as follows,

$$\sum_{k=b+1}^{N} kP_{Nk} \doteq \sum_{k=b+1}^{N} e^{-\alpha b} (\alpha b)^k / (k-1)! = e^{-\alpha b} (\alpha b)^{b+1} \{1 + \alpha R(\alpha, b)\} / b!. \quad (16)$$

Similar to above, the following approximations of the second and third term of (15) can also be derived respectively,

$$\sum_{k=b+1}^{N} P_{Nk} \doteq \sum_{k=b+1}^{N} e^{-\alpha b} (\alpha b)^k / k! = e^{-\alpha b} (\alpha b)^{b+1} R(\alpha, b) / (bb!) \quad (17)$$

and

$$\sum_{k=b+1}^{N} P_{Nk}/k \doteq \sum_{k=b+1}^{N} P_{Nk}/(k+1) \doteq \sum_{k=b+1}^{N} e^{-\alpha b} (\alpha b)^k / (k+1)! = e^{-\alpha b} (\alpha b)^{b+1} \{(b+1)R(\alpha, b) - b\} / \alpha b^2 (b+1) b!. \quad (18)$$

In addition, from

$$\sum_{k=1}^{N} P_{Nk} = 1 - e^{-\alpha b},$$

the S_N can be represented concisely by the load

factor α and the bucket size b as (19) finally,

$$S_N = 1 + \left\{e^{-\alpha b}(\alpha b)^{b+1}/2(1-e^{-\alpha b})b!\right\} \times \left\{1-(b-1)/\alpha(b+1) + R(\alpha,b)(\alpha+\alpha^{-1}+b^{-1}-(\alpha b)^{-1}-2)\right\}. \quad (19)$$

The evaluation formula of variance (20) can also be approximated by Poisson distribution.

$$V_N = 1 + \left\{\sum_{k=b+1}^{N}(2k(k-1)P_{Nk} + (11-6b)kP_{Nk} + (6b^2-18b+7)P_{Nk} - b(b-1)(2b-7)P_{Nk}/k\right\} \Big/ 6\sum_{k=1}^{N}P_{Nk} - S_N^2 \quad (20)$$

As done above, the following approximation of the first term of (20) can be obtained as well,

$$\sum_{k=b+1}^{N}k(k-1)P_{Nk} \doteqdot \sum_{k=b+1}^{N}e^{-\alpha b}(\alpha b)^k/(k-2)! = e^{-\alpha b}(\alpha b)^{b+1}\left\{\alpha+\alpha b+\alpha^2 bR(\alpha,b)\right\}/b!. \quad (21)$$

Therefore, similar to the average, the variance can be expressed concisely as follows,

$$V_N = 1 + \frac{1}{6}e^{-\alpha b}(\alpha b)^{b+1}(1-e^{-\alpha b})^{-1}(b!)^{-1} \times \left\{2\alpha b-4b+11+(b-1)(2b-7)\alpha^{-1}(b+1)^{-1} + R(\alpha,b)(2\alpha^2 b+(11-6b)\alpha+6b-18+7b^{-1} - (b-1)(2b-7)(\alpha b)^{-1})\right\} - S_N^2. \quad (22)$$

From the above arguments, the difference between Knuth's analysis and the proposed one is caused by the way in which a random variable and its probability distribution are defined. Besides this, only lists with more than one key per list are accessed for a successful searching in the proposed analysis, whereas Knuth failed to do so. As a result, the evaluation formula of the variance (10) derived by the traditional analysis can not be expressed in terms of the load factor α and bucket size b, but the formulae (19) and (22) derived by the proposed analysis are represented concisely and approximately by the load factor α, bucket size b and the function $R(\alpha,b)$.

4.2 Proposed Analysis for the AHU Model

In the AHU model[3], the keys with the same hash address are inserted in order from the head of a bucket, if the number of these keys are more than the bucket size b, a new bucket is linked at the end of the just previous bucket. Therefore, the keys located between the $(h-1)b+1$-th and the hb-th position from the head of a linked list are stored into the h-th bucket. During searching, the keys in a bucket will be accessed by a bucket unit.

Let $\hat{q}_{Nh}$ be the probability that a key is probed with the number of accesses h. The probability $\hat{q}_{Nh}$ will be given by

$$\hat{q}_{Nh} = \sum_{j=(h-1)b+1}^{hb}\sum_{k=j}^{N}\gamma_{kj}P_{Nk}, \quad (23)$$

where $h = 1,2,\ldots,\lambda$ and $\lambda = \lceil N/b \rceil$. Here, for a successful searching, we can derive the evaluation formulae of the average and the variance of the search cost, namely, $\hat{S}_N$ and $\hat{V}_N$ as (24) and (25),

$$\hat{S}_N = \sum_{h=1}^{\lambda}h\hat{q}_{Nh}\Big/\sum_{k=1}^{N}P_{Nk} = \sum_{h=1}^{\lambda}h\sum_{j=(h-1)b+1}^{hb}\sum_{k=j}^{N}\gamma_{kj}P_{Nk}\Big/\sum_{k=1}^{N}P_{Nk} \quad (24)$$

and

$$\hat{V}_N = \sum_{h=1}^{\lambda}h^2\sum_{j=(h-1)b+1}^{hb}\sum_{k=j}^{N}\gamma_{kj}P_{Nk} \Big/ \sum_{k=1}^{N}P_{Nk} - \hat{S}_N^2. \quad (25)$$

The formulae (24) and (25) can exactly evaluate the average and variance of the search cost with any probability distribution of the frequency of access based on the AHU model.

Here, we also assume that the frequency of access on a key is uniform as already described. Thus, it is possible that $\hat{S}_N$ is represented concisely by Poisson approximation. It has been obtained before that $\gamma_{kj} = 1/k$ $(j = 1,2,\ldots,k)$. Thus, the formula (24) can be rewritten as (26),

$$\hat{S}_N = \sum_{h=1}^{\lambda}h\sum_{j=(h-1)b+1}^{hb}\sum_{k=j}^{N}\frac{1}{k}P_{Nk}\Big/\sum_{k=1}^{N}P_{Nk} = 1 + \sum_{h=1}^{\lambda-1}\sum_{k=hb+1}^{N}((k-hb)/k)P_{Nk} \Big/ \sum_{k=1}^{N}P_{Nk}$$

$$= 1 + \sum_{h=1}^{\lambda-1} \left\{ \sum_{k=hb+1}^{N} P_{Nk} - hb \sum_{k=hb+1}^{N} P_{Nk}/k \right\} \Bigg/ \sum_{k=1}^{N} P_{Nk}, \quad (26)$$

where the Poisson approximations of $\sum_{k=hb+1}^{N} P_{Nk}$ and $\sum_{k=hb+1}^{N} P_{Nk}/k$ can be expressed as follows,

$$\sum_{k=hb+1}^{N} P_{Nk} \doteq e^{-\alpha b}(\alpha b)^{hb+1} R(\alpha/h, hb) / (hb(hb)!) \quad (27)$$

and

$$\sum_{k=hb+1}^{N} P_{Nk}/k \doteq \sum_{k=hb+1}^{N} P_{Nk}/(k+1) \doteq \frac{e^{-\alpha b}(\alpha b)^{hb+1}\{(hb+1)R(\alpha/h, hb) - hb\}}{h\alpha b^2 (hb)!(hb+1)}. \quad (28)$$

As the result of above approximation, $\hat{S}_N$ can be represented concisely by the load factor α, bucket size b and the function $R(\alpha, b)$ as follows,

$$\hat{S}_N \doteq 1 + \sum_{h=1}^{\lambda-1} \left(\left[e^{-\alpha b}(\alpha b)^{hb+1} \times \{R(\alpha/h, hb)(\alpha - h)(hb+1)/(h\alpha b) + h/\alpha\} \right] / \left[(hb+1)!(1 - e^{-\alpha b}) \right] \right). \quad (29)$$

Approximated by Poisson distribution, the variance $\hat{V}_N$ of (25) can also be represented with the following concise evaluation formula

$$\hat{V}_N = 1 + \sum_{h=1}^{\lambda-1} \sum_{k=hb+1}^{N} \{(2h+1)(k-hb)/k\} P_{Nk} \Bigg/ \sum_{k=1}^{N} P_{Nk} - \hat{S}_N^2$$
$$\doteq \sum_{h=1}^{\lambda-1} \left(\left[(2h-1) e^{-\alpha b}(\alpha b)^{hb+1} \times \{R(\alpha/h, hb)(\alpha - h)(hb+1)/(h\alpha b) + h/\alpha\} \right] / \left[(hb+1)!(1 - e^{-\alpha b}) \right] \right)$$
$$+ \left\{ \sum_{h=1}^{\lambda-1} \left(\left[e^{-\alpha b}(\alpha b)^{hb+1} \times \{R(\alpha/h, hb)(\alpha - h)(hb+1)/(h\alpha b) + h/\alpha\} \right] / \left[(hb+1)!(1 - e^{-\alpha b}) \right] \right) \right\}^2. \quad (30)$$

5. Numerical Tests

The proposed evaluation formulae can evaluate the external search cost in accordance with any probability distribution of the frequency of access for a successful search. In this numerical tests, we assume the following three probability distributions of the frequency of access on a key.

(1) The probability of the frequency of access on each key is equal likely, called "uniform", the probability ρ_i holds the relation $\rho_i = 1/N$ $(i = 1, 2, \ldots, N)$.

(2) The probability of the frequency of access on a key is reduced in half according to the order of inserting a key, called "binary", the probability ρ_i takes the relation $\rho_i = c/2^{i-1}$ $(i = 1, 2, \ldots, N)$, where $c = 1/(2 - 2^{1-N})$.

(3) The probability of the frequency of access on a key is reduced harmonically with the inserting order of a key, called "Zipf's law", the probability is $\rho = c/i$ $(i = 1, 2, \ldots, N)$, where $c = 1/H_N$ and H_N is the harmonic number, $H_N = \sum_{k=1}^{N} 1/k$.

Tables 1 and **2** express the average search cost in a successful search. It is a function of the load factor α and the bucket size b for the fixed hash table size M $(M = 50)$.

Table 1 shows the average search cost based on the Knuth's model, where the proposed analysis is compared to the Knuth's one with the probability distribution "uniform". The upper level of each row of Table 1 shows Knuth's results obtained by (3), and the lower level shows the results obtained by proposed evaluation formula (19). The numerical results in Table 1 show Knuth's overestimates more or less. These results suggest that when the number of overflow is statistically small enough the average search cost is very good.

Table 2 shows the numerical results obtained by the proposed evaluation formula (24) with the probability distribution such as "uniform", "binary" and "Zipf's law" based on the AHU model. The numerical behavior expressed in Table 2 shows that it is possible to evaluate the average search cost appropriately in accordance with the frequency of access on a key by the proposed analysis.

From these two tables, we can see the following rule: fixed the bucket size b the asymptotic average search cost will increase with increasing

Table 1 Comparisons of the average search cost in a successful search with uniform probability distribution based on the Knuth's model (table size $M = 50$, load factor $\alpha = N/(Mb)$).

Bucket size b	Load factor α									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	95%
1	1.0500	1.1000	1.1500	1.2000	1.2500	1.3000	1.350	1.400	1.450	1.50
	1.0352	1.0712	1.1080	1.1456	1.1840	1.2232	1.263	1.304	1.345	1.37
2	1.0063	1.0242	1.0520	1.0883	1.1321	1.1823	1.238	1.299	1.364	1.40
	1.0033	1.0127	1.0275	1.0471	1.0709	1.0986	1.130	1.164	1.200	1.22
3	1.0010	1.0071	1.0216	1.0458	1.0806	1.1259	1.181	1.246	1.319	1.40
	1.0004	1.0031	1.0097	1.0211	1.0378	1.0596	1.086	1.118	1.153	1.17
4	1.0002	1.0023	1.0097	1.0257	1.0527	1.0922	1.145	1.211	1.290	1.30
	1.0001	1.0009	1.0040	1.0109	1.0230	1.0409	1.065	1.094	1.129	1.15
5	1.0000	1.0008	1.0046	1.0151	1.0358	1.0699	1.119	1.186	1.286	1.30
	1.0000	1.0003	1.0018	1.0061	1.0151	1.0300	1.052	1.080	1.114	1.13
10	1.0000	1.0000	1.0002	1.0015	1.0070	1.0226	1.056	1.115	1.206	1.30
	1.0000	1.0000	1.0001	1.0006	1.0028	1.0092	1.023	1.045	1.076	1.09
20	1.0000	1.0000	1.0000	1.0000	1.0005	1.0038	1.018	1.059	1.150	1.20
	1.0000	1.0000	1.0000	1.0000	1.0002	1.0015	1.007	1.021	1.047	1.06
50	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.001	1.015	1.083	1.20
	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.001	1.005	1.021	1.04

Table 2 The average search cost in a successful search based on the AHU model considering the probability distribution such as "Uniform", "Binary" and "Zipf's law" (table size $M = 50$, load factor $\alpha = N/(Mb)$).

Bucket size b	Probability distribution	load factor α					
		0.5	1	2	3	4	5
1	Uniform	1.1305	1.2864	1.6529	2.0761	2.5358	3.017
	Binary	1.0002	1.0117	1.0132	1.0144	1.0154	1.016
	Zipf's law	1.0604	1.1182	1.2411	1.3731	1.5097	1.647
2	Uniform	1.0475	1.1741	1.5634	2.0301	2.4897	2.990
	Binary	1.0000	1.0002	1.0002	1.0003	1.0003	1.000
	Zipf's law	1.0158	1.0528	1.1597	1.2828	1.4081	1.532
3	Uniform	1.0245	1.1399	1.5463	1.9527	2.4263	2.935
	Binary	1.0000	1.0000	1.0000	1.0000	1.0000	1.000
	Zipf's law	1.0068	1.0362	1.1371	1.2532	1.3704	1.486
4	Uniform	1.0147	1.1235	1.5034	1.9266	2.3751	2.895
	Binary	1.0000	1.0000	1.0000	1.0000	1.0000	1.000
	Zipf's law	1.0037	1.0289	1.1260	1.2370	1.3490	1.460
5	Uniform	1.0095	1.1134	1.4836	1.9072	2.3412	2.864
	Binary	1.0000	1.0000	1.0000	1.0000	1.0000	1.000
	Zipf's law	1.0022	1.0247	1.1191	1.2263	1.3347	1.423
10	Uniform	1.0017	1.0877	1.4494	1.8629	2.2819	2.701
	Binary	1.0000	1.0000	1.0000	1.0000	1.0000	1.000
	Zipf's law	1.0003	1.0188	1.1044	1.2040	1.3285	1.370

the load factor α; while, fixed the load factor α the average search cost will decrease slightly with increasing the bucket size b. And, when the frequency of access on a key reduces quickly according to the inserting order, the average search cost is very good, since in our analysis we assume that the keys with the same hash address are inserted in order from the head of a bucket, and a new bucket is linked at the end of the previous bucket.

6. Conclusions

As mentioned above in this paper, taking account of the frequency of access on an individual key we have analyzed mathematically the average and variance of the search cost of hashing method for external searching on secondary storage devices. The proposed analyses have clarified the relationship between the inserting order and the locating position of keys. The

proposed evaluation formulae have been derived from the concrete probability distribution of the number of accesses.

Finally, it has been shown that the proposed analyses make it possible to evaluate exactly the performance of the hashing method for external searching.

Acknowledgments The authors would like to thank the anonymous referees for their helpful comments in improving the clarity of this paper.

References

1) Knuth, D.E.: Fundamental Algorithms, *The Art of Computer Programming*, Vol.1, pp.51–78, Addison-Wesley, Reading, MA (1973).
2) Knuth, D.E.: Sorting and Searching, *The Art of Computer Programming*, Vol.3, pp.506–549, Addison-Wesley, Reading MA (1973).
3) Aho, A.V., Hopcroft, J.E. and Ullman, J.D.: *Data Structure and Algorithms*, pp.122–134, Addison-Wesley, Reading, MA (1987).
4) Nakamura, R.: An Alternative Analysis of the Algorithm for Separate Chaining Technique of the Hashing Method, *Trans. IPS. Japan*, Vol.34, No.1, pp.10–15 (1993).
5) Nakamura, R.: An Analysis of Insertion and Search Algorithms of a Binary Search Tree, *Trans. IPS. Japan*, Vol.26, No.6, pp.1106–1112 (1985).
6) Feller, W.: *An Introduction to Probability Theory and Its Applications*, Vol.1, pp.39–40, John Wiley & Sons, New York (1957).

(Received October 9, 1995)
(Accepted September 12, 1996)

Ryozo Nakamura received the M.E. degree from Kumamoto University in 1968 and the D.E. degree in computer science from Kyushu University in 1985. From 1968 to 1974, he joined Chubu Electric Power Company. Since 1975 he has joined in Faculty of Engineering of Kumamoto University, and is presently a professor in Department of Computer Science. His current research interests include the design and analysis of algorithms and data structures.

Ningping Sun received the B.E. degree from Beijing Polytechnic University in 1982 and the M.E. degree from Kumamoto University in 1995. She joined the Department of Computer Science and Technology, Beijing Information Technology Institute as an instructor from 1983 to 1992, and is presently working for the D.E. degree in the Graduate School of Science and Technology, Kumamoto University. Her research interests include the design and analysis of algorithms and data structures, operating systems.

Takuo Nakashima received the M.E. degree from Kumamoto University in 1986. From 1986 to 1988 he joined Fujitsu Company. Since 1991 he has joined in the Faculty of Engineering, Kumamoto University, and is presently a research associate in Department of Computer Science. His research interests include the design and analysis of algorithms and data structures, and computer network.