January 1985

Report No. STAN-CS-85- 1038

Uniform Hashing is Optimal

by

Andrew C. Yao

Department of Computer Science

Stanford University
Stanford, CA 94305

Uniform Hashing Is Optimal

Andrew C. Yao
Computer Science Department
Stanford University
Stanford, California 94305

Abstract. It was conjectured by J. Ullman that uniform hashing is optimal in its expected retrieval cost among all open-address hashing schemes (*JACM* 19 (1972), 569-575). In this paper WC show that, for any open-address hashing scheme, the expected cost of retrieving a rccord from a large table which is a-fraction full is at least $\frac{1}{\alpha} \log \frac{1}{1-\alpha} + o(1)$. This proves Ullman's conjecture to be true in the asymptotic sense.

Keywords: Hashing, hashing function, optimality, uniform hashing.

This work was supported in part by thc National Scicncc Foundation under Grant Number MCS-8308109.

1. Introduction.

Hashing is a frequently used technique for storing and retrieving records maintained in the form of a table. In open-address hashing, the "key" of each record is mapped by a hashing function to a sequence of table locations, and records are inserted and retrieved by following this sequence. In particular, *uniform hashing* employs a hashing function that maps keys into random permutations. For uniform hashing, it is known [2] that the expected cost of inserting a new key into a table α-fraction full is essentially equal to $\frac{1}{1-\alpha}$ for a large table, while the expected cost of retrieving a record in the table is† essentially $\frac{1}{\alpha}\log\frac{1}{1-\alpha}$.

In 1972, Ullman [4] raised the optimality question of hashing function, and defined a mathematical model for discussing it. He showed that, in terms of the expected insertion cost of a new key, no hashing function can have a lower cost than the uniform hashing function *all the* time; he also exhibited a hashing function that performs better than uniform hashing some of the time. Ullman conjectured that, in terms of the expected retrieval cost, uniform hashing is optimal all the time. The main theorem of the present paper establishes Ullman's conjecture in the asymptotic sense, namely, the retrieval time using any hashing function is at least $\frac{1}{\alpha}\log\frac{1}{1-\alpha}+o(1)$.

Knuth [3] raised a weaker conjecture that, among *single-hashing functions,* which form a restricted family of hashing functions, none can perform substantially better than the performance bounds of a random single-hashing function. That conjecture was proved by Ajtai, Komlós, and Szemerédi [1]. The proof of our main theorem is based on an adaptation of the approach developed in [1]. interested readers may refer to [1] for more discussions on the intuition behind this approach.

2. Terminology.

A model for studying the optimality of hashing functions was first formulated by Ullman [4]. We summarized the essential definitions below, with some slight changes in terminology.

Consider a table of M locations, where M is any positive integer. Let Q_M be the set of all permutations of $\{0,1,2,\ldots,M-1)$. *A hashing function h* assigns each *key* K a permutation $h(K)=i_1 i_2 \cdots i_M \in Q_M$. In inserting a key K into the table, we try locations $i_1, i_2, \ldots$ in turn until an empty slot is found, where K is then inserted; the *insertion cost* is measured by the number of locations tried until an empty slot is found. Now, suppose a sequence of N keys have been inserted, where $1 \le N \le M$; let T be the resulting hash table. To retrieve a key K in T, the rule is again to try in sequence the locations $i_1, i_2, \ldots$ as given by $h(K)$, until K is found; the *retrieval cost* is the number of locations tried.

An (N, *M)-scenario p* is a sequence $(\sigma_1,\sigma_2,\ldots,\sigma_N)$ where each $\sigma_i \in Q_M$. Let T_ρ be the table obtained when a sequence of N keys $K_1, K_2, \ldots, K_N$ have been inserted, with $h(K_i) =$

---

†In this paper all the logarithms are in the natural base e.

σ_i. Denote by $A(K_i, T_\rho)$ the retrieval cost if K_i is to be retrieved from T_ρ, and let $A_h(T_\rho) = \frac{1}{N}\sum_{1\le i\le N} A(K_i, T_\rho)$.

To analyze the performance of a hashing function, we simply identify each hashing function h with a probability distribution p_h over Q_M. Consider a random (N, M)-scenario $\rho = (\sigma_1, \sigma_2, \ldots, \sigma_N)$, where each $\sigma_i \in Q_M$ is independently distributed according to p_h. Let $C_h(N-1, M)$ be the expected value of $A(K_N, T_\rho)$, and $C'_h(N, M)$ the expected value† of $A_h(T_\rho)$.

Uniform hashing corresponds to the distribution $p_h(\pi) = 1/M!$ for all $\pi \in Q_M$. For this hashing function, it was known (see, e.g. Knuth [2]) that $C_h(N, M) = \frac{M+1}{M-N+1}$ and $C'_h(N, M) = \frac{M+1}{N}(H_{M+1} - H_{M-N+1})$, where H_S is the Harmonic number $1 + \frac{1}{2} + \cdots + \frac{1}{S}$; for fixed $0 < \alpha = N/M < 1$ and $N \to \infty$, this gives $C_h(N, M) = \frac{1}{1-\alpha} + o(1)$ and $C'_h(N, M) = \frac{1}{\alpha}\log\frac{1}{1-\alpha} + o(1)$.

In the remainder of this paper, we will use α to denote *the loading factor* N/M, and G_M to denote the set of all hashing functions for tables of size M (i.e., the set of all probability distributions over Q_M). Our main result is the following theorem.

Main Theorem. For any $\epsilon > 0$, there exists a constant a_1 such that the following is true: For all integers $N, M > 1$ satisfying $\epsilon < \alpha < 1 - \epsilon$, and for any hashing function $h \in G_M$,

$$C'_h(N, M) \ge \frac{1}{\alpha}\log\frac{1}{1-\alpha} - \frac{a_1 \log M}{M}. \tag{1}$$

Also, there exists an absolute constant b such that for all integer $M > 1$ and any hashing function $h \in G_M$,

$$C'_h(M, M) \ge \log M - \log\log M - b. \tag{2}$$

A *single-hashing function* h, in our notation, is a hashing function with $p_h(\pi) = 1/M$ for π in a certain set $\{\pi_0, \pi_1, \ldots, \pi_{M-1}\}$, where $\pi_j \in Q_M$ starts with j, and $p_h(\pi) = 0$ otherwise. Ajtai et. al. [1] proved that (1) and (2) are true when h is a single-hashing function.

3. Proof of the Main Theorem.

3.1. Reductions.

Let us call a hashing function $h \in G_M$ *regular* if $p_h(\pi) \neq 0$ for every $\pi \in Q_M$. To prove the Main Theorem, we need only to demonstrate that inequalities (1) and (2) hold for all regular hashing functions h, for some constants a_1 and b, because the quantity $C'_h(N, M)$ is a continuous function (in fact a polynomial) in the $M!$ variables $\{p_h(\pi) \mid \pi \in Q_M\}$.

Suppose that M, N $(1 \le N \le M)$ are given, and that d is any integer with $1 \le d < N$. Let $h \in G_M$ be any regular hashing function. For any integer L, a random (L, M)-scenario $\rho =$

† In Knuth [2], the notation C' is used to denote the *insertion* cost instead of the *retrieval cost.* We follow here the usage in Ullman [4].

$(\sigma_1, \sigma_2, \ldots, \sigma_N)$ will be called an *h-random (L, M)-scenario*, if σ_i are independently distributed according to p_h. Consider the insertion of N keys according to an h-random (N, M)-scenario. For each $k \in \{0, 1, 2, \ldots, M-1\}$, let v_k be the probability that table location k is occupied after $N -$ d keys have been inserted, and let δ_k be the expected number of times location k has been probed during the insertion process of the N keys. Clearly,

$$C'_h(N, M) = \frac{1}{N} \sum_{0 \leq k < M} \delta_k, \tag{3}$$

and

$$N - d = \sum_{0 \leq k < M} v_k. \tag{4}$$

Let $f(x) = \lambda - e^{-\lambda} \sum_{i > d} (i - d) \frac{\lambda^i}{i!}$, where $\lambda = -\log(1 - x)$. Our main effort will be in proving the following proposition. Roughly speaking, it states that if location k is probed at least once, then it is likely to have been probed a fair number of times.

Proposition 1. $\delta_k \geq f(v_k)$ for each k.

The validity of inequalities (1) and (2) (for some constants a, and b) is an analytic consequence of (3), (4) and Proposition 1, as demonstrated in [1]. We review it below. Without loss of generality, we can assume that N > $\lceil 10 \log M \rceil$ > 1. First, observe that f(x) is a convex function (which can be verified by showing that $f'' > 0$); this implies that $\sum_i r_i f(x_i) \geq f(\sum_i r_i x_i)$ for $r_i > 0$ and $\sum_i r_i = 1$. Then, from (3), (4) and Proposition 1, we obtain

$$\begin{aligned} C'_h(N, M) &\geq \frac{1}{N} \sum_{0 \leq k < M} f(v_k) \\ &\geq \frac{M}{N} f\left(\frac{\sum_k v_k}{M}\right) \\ &= \frac{M}{N} f\left(\frac{N-d}{M}\right). \end{aligned}$$

By choosing $d = \lceil 10 \log M \rceil$, one can show that $f\left(\frac{N-d}{M}\right)$ is well approximated by $-\log(1 - \frac{N-d}{M})$; that is, $f(x) \approx \lambda$ in this case. The error bounds involved in this approximation are dependent on N, M and d, but are clearly independent of h. A close examination of the error bounds leads to inequalities (1) and (2).

It remains only to prove Proposition 1. For the rest of the proof, let $k \in \{0, 1, \ldots, M-1\}$ be fixed. We shall divide the proof into three parts. In part 1 we give a procedure for generating an h-random (N, M)-scenario p. This procedure first generates randomly a special type of scenarios ω, called *skeletons*, and then generates a random p with a distribution determined by ω. In part 2, we derive a lower bound to δ_k for a random p generated by the above procedure when the skeleton is ω. The procedure in part 1 is designed in such a way that the derivation of a nontrivial lower bound is possible for a given skeleton. In part 3, the lower bound obtained in the previous part is averaged over ω to obtain a lower bound to δ_k to give Proposition 1. These three parts are presented in order in the ensuing three subsections.

3.2. Generating a Random Scenario.

We first dcfinc some notations. Let $0 \leq L < M$ bc any integer. For any $(L,$ M)-scenario ω, partition Q_M into two disjoint parts $Q[\omega]$ and $Q'[\omega]$ as defined below. Considcr the table T_ω obtaincd by inserting keys according to ω, and let $B_\omega \subseteq \{0, 1, \ldots, M-1\}$ be the set of occupied positions in T_ω. We put $\pi \in Q_M$ into $Q[\omega]$ if a new key K with $h(K) = \pi$ will occupy position k when inserted into T_ω; otherwise, let $\pi \in Q'[\omega]$. In other words, if $k \in B_\omega$, then $Q[\omega] = \emptyset$; otherwise, $Q[\omega]$ contains all those π of the form $i_1, i_2, \ldots, i_{\ell-1}, k, i_{\ell+1}, \ldots, i_M$ with $i_t \in B_\omega$ for $1 \leq t < \ell$. For example, when ω is the empty string, $Q[\omega]$ is the set of permutations π that start with k.

For any $(L,$ M)-scenario $\omega = (\pi_1, \pi_2, \ldots, \pi_L)$, wc will use $\omega^{(j)}$ to denotc its prefix, the (j, M)-scenario $\omega = (\pi_1, \pi_2, \ldots, \pi_j)$. An (N — d, M)-scenario $\omega = (\pi_1, \pi_2, \ldots, \pi_{N-d})$ will be called a *skeleton scenario,* or simply, a *skeleton,* if k is not occupied in the table T_ω. Note that we can alternatively define a skclcton as an (N — d, M)-scenario for which $\pi_j \in Q'[\omega^{(j-1)}]$ for all $1 \leq j \leq N-d$.

For any noncmpty subset $V \subseteq Q_M$, let p_V denote the probability distribution obtained when p_h is restricted to V. Lct $p_h(V)$ dcnote $\sum_{\pi \in V}$ p&r), then $p_V(\pi) = p_h(\pi)/p_h(V)$ for $\pi \in V$. Note that $p_h(V) \neq 0$ for all noncmpty V, since h is a regular hashing function.

We now dcscribe a procedure that generates a random (N, M)-scenario $p = (\sigma_1, \sigma_2, \ldots, \sigma_N)$. It will be seen that p is an h-random (N, M)-scenario, that is, σ_i are independently distributed according to p_h. It proceeds in three steps.

Procedure RANDSCEN;

Stcp 1: Generate a random skcleton $\omega = (\pi_1, \pi_2, \ldots, \pi_{N-d})$ by successively generating $\pi_1, \pi_2, \ldots,$ each new π_j is randomly chosen from $Q'[\omega^{(j-1)}]$ according to the probability distribution p_{V_j}, where $V_j = Q'[\omega^{(j-1)}]$.

Step 2: For each $1 \leq j \leq N-d$, generate first an integer $r_j \geq 0$ distributed geometrically with probability $u_{\omega,j} = p_h(Q[\omega^{(j-1)}])$, that is, $\Pr\{r_j = i\} = (1 - u_{\omega,j})(u_{\omega,j})^i$; generate a random (r_j, M)-sccnario $\omega_j = (\pi_{j1}, \pi_{j2}, \ldots, \pi_{j,r_j})$, where each $\pi_{j,t}$ is randomly and indepcndcntly chosen from $W_j = Q[\omega^{(j-1)}]$ distributcd according to p_{W_j}.

Step 3: Let $r = \sum_{1 \leq j \leq N-d} r_j$ and $\chi = (\omega_1, \pi_1, \omega_2, \pi_2, \ldots, \omega_{N-d}, \pi_{N-d})$. If $r > d$, then let p be the (N, M)-scenario $\chi^{(N)}$; otherwise, generate $d - r$ additional random $\sigma_{N-(d-r)+1}, \sigma_{N-(d-r)+2}, \ldots, \sigma_N$, each chosen independently from Q_M according to distribution p_h, and let ρ bc $(\chi, \sigma_{N-(d-r)+1}, \sigma_{N-(d-r)+2}, \ldots, \sigma_N)$.

End RANDSCEN.

Note that as h is rcgular, $p_h(\pi) \neq 0$ for cvery $\pi \in Q_M$, which implics $u_{\omega,j} = p_h(Q[\omega^{(j-1)}]) < 1$ in stcp 2 of thc abovc procedure. Thus, thc distribution for r_j, $\Pr\{r_j = i\} = (1 - u_{\omega,j})(u_{\omega,j})^i$ is wcll defined.

Lemma 1. The p generated by RANDSCEN is an h-random (N, M)-scenario.

Proof. Let $\eta = (\eta_1, \eta_2, \ldots, \eta_N)$ be any (N, M)-scenario . We will prove that for a random p generated by RANDSCEN, Pr{p = η} is equal to $\prod_{1 \le i \le N} p_h(\eta_i)$. This immediately implies the lemma.

Write η as $(\omega'_1, \pi'_1, \omega'_2, \pi'_2, \ldots, \omega'_t, \pi'_t, \omega'_{t+1})$, such that $\pi'_j \in Q'[\pi'_1\pi'_2 \ldots \pi'_{j-1}]$ for $1 \le j \le t$ and† $\omega'_j \in Q[\pi'_1\pi'_2 \ldots \pi'_{j-1}]^*$ for $1 \le j \le t+1$. It is easy to see that this representation is unique. Let us write $\omega'_j = (\pi'_{j1}, \pi'_{j2}, \ldots, \pi'_{j,r'_j})$ for $1 \le j \le t+1$, where each $\pi'_{j\ell} \in Q[\pi'_1\pi'_2 \ldots \pi'_{j-1}]$; r'_j may be 0. Define $z_j = p_h(Z_j)$ for $1 \le j \le t+1$, where $Z_j = Q[\pi'_1\pi'_2 \ldots \pi'_{j-1}]$.

Case 1) $0 \le t < N - d$.

Let X_1 be the event that in step 1, $\pi_j = \pi'_j$ for $1 \le j \le t$, X_2 be the event that in step 2, $\omega_j = \omega'_j$ for $1 \le j \le t$, and X_3 be the event that in step 2 $r_{t+1} \ge r'_{t+1}$ and $\omega_{t+1}^{(r'_{t+1})} = \omega'_{t+1}$. It is easy to see that RANDSCEN will generate $p = \eta$ if and only if events X_1, X_2, X_3 all occur. Due to the the independence of X_2 and X_3, we have

$$\begin{aligned}\Pr\{p = \eta\} &= \Pr\{X_1\} \cdot \Pr\{X_2, X_3 \mid X_1\} \\ &= \Pr\{X_1\} \cdot \Pr\{X_2 \mid X_1\} \cdot \Pr\{X_3 \mid X_1\}.\end{aligned}$$

An elementary probabilistic calculation shows that

$$\Pr\{X_1\} = \prod_{1 \le j \le t} \frac{p_h(\pi'_j)}{1 - z_j},$$

$$\begin{aligned}\Pr\{X_2 \mid X_1\} &= \prod_{1 \le j \le t} \left((1 - z_j)(z_j)^{r'_j} \prod_{1 \le i \le r'_j} \frac{p_h(\pi'_{ji})}{z_j} \right) \\ &= \prod_{1 \le j \le t} \left((1 - z_j) \prod_{1 \le i \le r'_j} p_h(\pi'_{ji}) \right),\end{aligned}$$

and

$$\begin{aligned}\Pr\{X_3 \mid X_1\} &= \Pr\{r_{t+1} \ge r'_{t+1} \mid X_1\} \prod_{1 \le i \le r'_{t+1}} \frac{p_h(\pi'_{t+1,i})}{z_{t+1}} \\ &= (1 - z_{t+1}) \sum_{\ell \ge r'_{t+1}} (z_{t+1})^{\ell} \prod_{1 \le i \le r'_{t+1}} \frac{p_h(\pi'_{t+1,i})}{z_{t+1}} \\ &= \prod_{1 \le i \le r'_{t+1}} p_h(\pi'_{t+1,i}).\end{aligned}$$

The above formulas lead to

$$\begin{aligned}\Pr\{p = \eta\} &= \prod_{1 \le j \le t} \left(p_h(\pi'_j) \prod_{1 \le i \le r'_j} p_h(\pi'_{ji}) \right) \prod_{1 \le i \le r'_{t+1}} p_h(\pi'_{t+1,i}) \\ &= \prod_{1 \le i \le N} p_h(\eta_i).\end{aligned}$$

---

†For any set D, the notation D^* will stand for the set of all finite sequences of elements in D (including the empty sequence).

Case 2) $t \geq N - d$.

Let X_1 be the event that in step 1, $\pi_j = \pi'_j$ for $1 \leq j \leq N - d$, X_2 be the event that in step 2, $\omega_j = \omega'_j$ for $1 \leq j \leq N - d$, and X3 be the event that in step 3 $\sigma_{N-(d-r')+1} = \eta_{N-(d-r')+1}, \sigma_{N-(d-r')+2} = \eta_{N-(d-r')+2}, \ldots, \sigma_N = \eta_N$, where $r' = \sum_{1 \leq j \leq N-d} r'_j$. As in case 1, RANDSCEN will generate $p = \eta$ if and only if events X_1, X_2, X_3 all occur. A calculation similar to that in case 1 gives $\Pr\{\rho = \eta\} = \prod_{1 \leq i \leq N} p_h(\eta_i)$. This completes the proof of Lemma 1. ▮

3.3. Lower Hound on δ_k for Skeleton w.

Suppose p is an (N, M)-scenario generated by RANDSCEN, with r being the parameter generated in step 2 during the process.

Lemma **2.** Let $s(p)$ be the number of times that table position k is probed during the insertion of N keys according to p. Then $s(p) \geq \min\{r, d\}$.

Proof. Write $p = (\omega_1, \pi_1, \omega_2, \pi_2, \ldots)$ with $\omega_j = \pi_{j1}\pi_{j2} \cdots \pi_{jr_j}$ in the notation of procedure RANDSCEN. It is easy to see that each insertion that corresponds to a $\pi_{j\ell}$ in p will probe location k in the insertion process, since even if we omit all the insertions $\pi_{j'\ell'}$ that precede it in p, this insertion will still probe location k. As the total number of $\pi_{j\ell}$ in p is equal to $\min\{r, d\}$, the lemma follows.1

Imagine that we follow the steps in RANDSCEN to generate an h-random (N, M)-scenario ρ. We wish to analyze this process of generating p to estimate the expected value of $\min\{r, d\}$; then Lemma 2 will provide a needed lower bound since δ_k is the expected value of $s(p)$.

Consider the execution of RANDSCEN as a stochastic process. Let Ω denote the random variable corresponding to w in Step 1, R_j denote the random variable for r_j in step 2, and $R = \sum_{1 \leq j \leq N-d} R_j$. Let S denote the random variable corresponding to $s(p)$ defined in Lemma 2. Clearly,

$$\delta_k = E(S). \tag{5}$$

We also introduce some scalars. Let $\xi_\omega = \Pr\{\Omega = w\}$ for skeleton w; let $\mu_{\omega,j}$ denote $p_h(Q[\omega^{(j-1)}])$ as defined in Step 2 of procedure RANDSCEN.

Our approach is to analyze the expected value of $\min\{r, d\}$ for fixed w, and then average over w. From Lemma 2, we obtain

$$\begin{aligned} E(S \mid \Omega = w) &\geq E(\min\{d, R\} \mid \Omega = w) \\ &= \sum_{1 \leq i \leq d} \Pr\{R \geq i \mid \Omega = w\} \\ &= \sum_{1 \leq i \leq d} \Pr\{R^{(\omega)} \geq i\}, \end{aligned} \tag{6}$$

where $R^{(\omega)} = \sum_{1 \leq j \leq N-d} R_j^{(\omega)}$ with $R_j^{(\omega)}$ being the random variable R_j restricted to the probability space specified by $\Omega = w$. As $R_j^{(\omega)}$, $1 \leq j \leq N - d$, are independent variables with distribution $\Pr\{R^{(\omega)} = i\} = (1 - \mu_{\omega,j})(\mu_{\omega,j})^i$, the following analytic result from Ajtai et.al. [1] applies.

Lemma 3. [1] Suppose $Y = \sum_{1\le j\le a} Y_j$, where $Y_1, Y_2, \ldots, Y_a$ are independent random variables with $\Pr\{Y_j = i\} = (1 - y_j)(y_j)^i$. Then $\Pr\{Y \ge i\} \ge e^{-\lambda} \sum_{\ell\ge i} \frac{\lambda^\ell}{\ell!}$ where $\lambda = -\log\left(\prod_{1\le j\le a}(1-y_j)\right)$.

Proof. See [1]. ∎

From Lemma 3 and (6) we have

$$E(S \mid \Omega=\omega) \ge \sum_{1\le i\le d} e^{-\lambda_\omega} \sum_{\ell\ge i} \frac{(\lambda_\omega)^\ell}{\ell!}$$
$$= \lambda_\omega - e^{-\lambda_\omega}\sum_{\ell>d}(\ell-d)\frac{(\lambda_\omega)^\ell}{\ell!}. \tag{7}$$

where $\lambda_\omega = -\log\left(\prod_{1\le j\le N-d}(1-\mu_{\omega,j})\right)$.

3.4. Completing the Proof.

Consider again a random ρ generated by RANDSCEN. Let A be the random variable that is equal to 1 if location k is occupied in $T_{\rho(N-d)}$ and 0 otherewise; let A_ω denote A restricted to the situation $\Omega = \omega$.

For any skeleton ω, it is easy to check from the definitions that

$$1 - \Pr\{\Delta_\omega = 1\} = \Pr\{R^{(\omega)} = 0\}$$
$$= \prod_{1\le j\le N-d}(1 - \mu_{\omega,j}). \tag{8}$$

It follows from (7), (8) that

$$E(S \mid \Omega=\omega) \ge f(\Pr\{\Delta_\omega = 1\}). \tag{9}$$

Using (5), (9) and the convexity of f, we obtain

$$\delta_k = \sum_\omega \xi_\omega E(S \mid \Omega=\omega)$$
$$\ge \sum_\omega f(\sum_\omega \xi_\omega \,.\Pr\{\Delta_\omega = 1\})$$
$$= f(\Pr\{\Delta = 1))$$
$$= f(v_k).$$

This proves Proposition 1 and hence the Main Theorem.

4. Concluding Remarks.

In this paper we have shown that uniform hashing is asymptotically optimal in retrieval cost. Can one prove that uniform hashing is also asymptotically optimal in the insertion cost all the time?

More precisely, can onc prove that for any fixed $0 < \alpha < 1$, $C_h(N, M) \geqq \$\text{-----}_{-\,\alpha} + o(1)$ for all h?

Rcfcrcnces.

[1] M. Ajtai, J. Komlós, and E. Szcmerédi, "There is no fast single hashing function," *Information ProcessingLetters* 7 (1978), 270-273.

[2] D. E. Knuth. *The Art of Cotnputer Programtning, Vol.* 3, Addison-Welslcy, Reading, Massachusctts, 1975, second printing.

[3] D. E. Knuth, "Computer science and its relation to mathematics," *Am. Math. Monthly* 8 (1974), 323-343.

[4] J. D. Ullman, "A note on the efficiency of hashing function," *JACM* 19 (1972), 569-575.