

Sonderdruck
Reprint

AEÜ

Archiv für Elektronik und Übertragungstechnik
International Journal of Electronics and Communications

Contribution to the Special Issue
on Teletraffic Theory and Engineering
in Memory of Félix Pollaczek
(1892–1981)



S. Hirzel Verlag · Stuttgart

Scope

The AEÜ is an international scientific journal published bimonthly. The field of interest covered includes information theory and communications, modulation, coding, networks; microwave theory and techniques, antennas, wave propagation, optical communication, radar, sonar; switching theory and techniques; system and signal theory, speech and image processing; solid-state electronics; network theory and circuit design, computer-aided design methods and testing.

Editorial Board

Prof. Dr.-Ing. H. Döring, RWTH-Aachen, Templergraben 55, D-5100 Aachen, Phone (02 41) 80 79 31-32.

Prof. Dr. rer. nat. W. Harth, Technische Universität München, Arcisstrasse 21, D-8000 München 2, Phone (0 89) 21 05-29 48.

Prof. Dr.-Ing. Dr.-Ing. E.h. H. Marko, Technische Universität München, Arcisstrasse 21, D-8000 München 2, Phone (0 89) 21 05-85 83.

Prof. Dr. R. Mittra, Department of Electrical and Computer Engineering, University of Illinois, Urbana, Ill. 61801-2991, USA.

Prof. Dr. techn. J. A. Nossek, Technische Universität München, Arcisstrasse 21, D-8000 München 2, Phone (0 89) 21 05-85 01, Fax (0 89) 21 05-85 04.

Prof. Dr.-Ing. K. Tröndle, Universität der Bundeswehr, Werner-Heisenberg-Weg 38, D-8014 Neubiberg, Phone (0 89) 60 04 36 29.

Prof. Dr.-Ing. Dr.-Ing. E.h. H.-G. Unger, Technische Universität Braunschweig, Mühlenpfordstrasse 23, D-3300 Braunschweig, Phone (05 31) 3 91 24 22.

Managing Editor

Dr.-Ing. R. Pauli, Fischmühle 3, D-8255 Schwindegg, Germany. Phone (0 89) 21 05-85 21, Fax (0 80 82) 81 13, e-mail pauli@nws.e-technik.tu-muenchen.de, (responsible).

Publisher.

S. Hirzel Verlag GmbH & Co., Birkenwaldstrasse 44, Postfach 10 22 37, D-7000 Stuttgart 10, Phone (07 11) 25 82-0, Telex 72 36 36 daz d, Fax (07 11) 25 82-290.

Printer

Konrad Triltsch, Graphischer Betrieb, Haugerring 5, D-8700 Würzburg.

Subscription

The annual subscription rate is DM 598.- plus postage, the price of a single copy is DM 118.- plus postage. Orders can be placed with any bookseller or with the publisher. A subscription, if not placed for a limited period, is valid until cancellation. Notice of cancellation can be given at the end of the year and must be received by the publisher by 15th November of the current year.

Advertising

For advertising please contact the Publisher and ask for Media Information.

Copyright

© 1992 S. Hirzel Verlag GmbH & Co., Stuttgart. The journal and all contributions and illustrations contained herein are under copyright. Any reproduction outside copyright law is inadmissible and a punishable offence. This concerns in particular translation, reprinting, microfilming or reproduction in any other form, or storage in information retrieval systems.

Printed in Germany.

Instructions for Authors.

1. The paper must contain original research results not yet published or submitted for publication elsewhere.
2. Please send 2 copies of the complete manuscript to the Managing Editor.
3. The manuscript must be typewritten using double space on one side only of DIN A4 sized paper and should not exceed 20 pages for full papers and 6 pages for Letters.
4. The manuscript must comprise the following parts:
 - (a) Title of paper.
 - (b) Full names (first name, initials, last name) of all authors.
 - (c) Academic titles of all authors.
 - (d) Affiliation and address of all authors.
 - (e) Correspondence address with phone, Fax, e-mail.
 - (f) Abstract, not exceeding 200 words.
 - (g) German translation of title and abstract (if available).
 - (h) Body of the paper. Sections, equations, figures, tables, and references must be numbered consecutively by arabic numbers as 1, 2, ... Refer to these items in the text as follows: Section(s), Fig(s), Table(s) and for equations and references to the bibliography as (), ()-() and [], []-[], respectively.
 - (i) Footnotes, numbered consecutively within the text. However, footnotes should be avoided as far as possible.
 - (j) Acknowledgement (optional).
 - (k) Appendix (optional).
 - (l) List of references.
 - (m) Photography and a short technical Biography not exceeding 200 words of each author in case of a full Paper. Not required for Letters.
 - (n) Figure captions on separate sheet.
 - (o) Tables with table caption on top of the table on separate sheets.
 - (p) Copies of figures on separate sheets. After acceptance of paper camera-ready drawings will be required.
5. Style of References:
 - (a) Articles in journals:
Youla, D. C.; Winter, F.: New theorems in the synthesis of passive multiplexers. *Int. J. Circuit Theory Appl.* 19 (1991), 341-364.
 - (b) Books:
Roberts, L. A.; Mullis, C. T.: *Digital signal processing*. Reading MA.: Addison Wesley, 1987.
 - (c) Conference papers:
Øien, G. E.; Lepsoy, S.: An inner product space approach to image coding by contraction transformation. *Proc. IEEE Int. Conf. Acoust., Speech, Sig. Process.*, Toronto, 1991, 2773-2776.
6. Figures should be drawn with black ink or laser printed on separate sheets, width of drawings not exceeding 18.5 cm (in rare cases 38 cm), size of lettering 5 mm. (The reduction factor is 0.36). Photographs glossy black/white, 9 cm x 12 cm or greater.
7. **Electronic Manuscripts.** When a paper is accepted for publication an electronic version of the manuscript will be requested formatted with LATEX. A special style file and further information is available from the Managing Editor. Please make use of electronic publishing as far as possible. It speeds up publication of your paper considerably and avoids time consuming proof reading.
8. The authors receive 75 reprints free of charge.

Teletraffic Theory Applied to the Analysis of Hash-Structured Files

Robert B. Cooper and Martin K. Solomon

Teletraffic Theory Applied to the Analysis of Hash-Structured Files

Certain hash-structured files consist of sequences (chains) of computer memory locations (slots) into which records are inserted, and from which they are later retrieved or deleted. If we assume that the records arrive to the file according to a Poisson process for insertion into a chain (randomly selected by the hash function), and reside in memory for a random length of time before being deleted, then we can associate this with a teletraffic model in which the records are "calls" and the slots in the chain are "trunks." In particular, we consider two models: (1) *hardpack*, where a deleted record is immediately physically removed from the file and is replaced by the last record on its chain, and (2) *softpack*, where a deleted record is marked, and a subsequent arriving record is written over the first marked record on its chain. The slots can be accessed individually or in groups (called "buckets"). In each case we are interested in the length of a *successful search* (the number of buckets inspected until an arbitrary record is located) and the length of an *unsuccessful search* (the number of buckets inspected until it is determined that an arbitrary record is not in the file). If we identify these file structures as analogous to trunk groups with ordered hunt, then these models can be analyzed using results obtained many years ago (in particular, Kosten (1937) and Burke (1971)) for teletraffic applications, results which are not well known or easily accessible to the database-performance-analysis community.

Verkehrstheoretische Analyse von hash-strukturierten Speichern

Gewisse hash-strukturierte Speicher bestehen aus Sequenzen (Ketten) von Speicherplätzen (Slots) im Rechner, in die Records eingefügt und in denen sie später wieder aufgesucht oder gelöscht werden können. Wir nehmen an, daß die Records entsprechend einem Poisson Prozeß zur Einfügung in die Kette eintreffen (wobei sie durch die Hash-Operation zufällig ausgewählt werden), und daß sie bis zur Löschung zufällig lang im Speicher verweilen. Dieses Geschehen kann man einem Verkehrsmodell zuordnen, in dem die Records als "Rufe" und die Slots als "Kanäle" interpretiert werden. Insbesondere werden zwei Modelle betrachtet: (1) die *Hardpackung*, bei der ein gelöschter Record physisch entfernt und durch den letzten Record der Kette ersetzt wird, und (2) die *Softpackung*, bei der ein gelöschter Record markiert und ein anschließend ankommender Record in den ersten markierten Record seiner Kette eingeschrieben wird. Die Slots können individuell oder in Gruppen (genannt "Buckets") angesprochen werden. In jedem dieser Fälle interessieren die Länge einer *erfolgreichen Suche* (Anzahl der inspizierten Buckets bis ein freier Record gefunden wurde) und die Länge einer *vergeblichen Suche* (Anzahl der inspizierten Buckets bis zur Feststellung, daß im Speicher kein freier Record verfügbar ist). Solche Speicherstrukturen dürfen in Analogie zu Kanalgruppen mit geordnetem Suchvorgang gesehen werden und können daher unter Verwendung von Ergebnissen analysiert werden, die vor vielen Jahren für verkehrstheoretische Anwendungen (insbesondere von Kosten (1937) und Burke (1971)) erarbeitet wurden. Diese Ergebnisse sind den Experten für die Leistungsanalyse von Datenbanken weniger gut bekannt bzw. nur schwer zugänglich.

1. Introduction

Chapter 6 of Knuth's 1973 classic 3-volume tome, *The Art of Computer Programming* [1], deals with methods of searching for specified entities (records) in main-memory and disk files. Each method has its own advantages and disadvantages in comparison with competing methods, and therefore computer scientists would like to describe and quantify performance measures that would facilitate making the appropriate choice of data structure for any given type of application. In particular, we would like to provide answers for questions such as: (1) How long does it take to locate a particular record that was previously stored in the file (*successful search*)? (2) How long does it take to ascertain that a particular record is not stored in the file (*unsuccessful*

search)?

The natural "first-pass" performance model of such a data structure is static and combinatorial, as exemplified by Knuth's discussion. Typically, one imagines that the file is constructed by selecting (the keys of) n records at random, and inserting the records one-by-one into the file according to the particular algorithm (or method) under study; then, one constructs a combinatorial probabilistic argument to calculate (1) the time required by the algorithm to locate a record chosen at random from among the n records known to be in the file (successful search), or (2) the time required to ascertain that a record with a randomly-generated key is not in the file (unsuccessful search). These models are static in the sense that they do not explicitly account for changes in the size or structure of the file as it evolves in response to deletions or insertions of new records after the initial construction of the n -record file.

If we take the view that the records arrive (for insertion into the file) according to some stochastic process, and remain in the file for some random length of time before being deleted, then we can associate this with a

Received November 27, 1992.

Prof. Dr. R. B. Cooper, Prof. Dr. M. K. Solomon, Department of Computer Science and Engineering, Florida Atlantic University, Boca Raton, Florida 33431, USA.

queueing (or teletraffic) model in which the records are customers (or calls) and the memory locations ("slots") where the records are stored are servers (or trunks). The slots can be accessed individually, or in groups (called "buckets"). Although it may seem obvious that the queueing-theory paradigm might be especially useful because it naturally captures the dynamic behavior of the system, nevertheless relatively few papers have been written that adopt this viewpoint. A recent (1992) example of a paper that uses queueing theory is Aldous, Hofri, and Szpankowski [2], which addresses the management technique known as "hashing with lazy deletion" (in which *all* records in a bucket whose residence times have expired are physically deleted upon the arrival of the first subsequent record for insertion into that bucket). These authors use queueing theory to study the excess amount of storage space required to handle lazy, as opposed to immediate, deletion. (Significantly, the authors felt it necessary to supply footnotes that give basic facts about queueing theory and its standard notation.) Earlier (1987) Morrison, Shepp, and Van Wyk [3] also had applied queueing theory to the analysis of space requirements under lazy deletion. Mendelson and Yechiali (1979) [4] studied a model similar to ours, but without deletions. An earlier (1984) paper by us, Cooper and Solomon [5], which addresses the question of how much time elapses until a bucket overflows, is similar in spirit to the present paper; previously (1981), Larson [6] used queueing theory to study a similar model. But, in view of the power of the approach, it seems that relatively little use has been made of queueing theory in this context.

In this paper, we apply queueing theory to address questions (1) and (2) for two models of a hash-structured file. More precisely, we consider certain hash-structured files that consist of sequences (chains) of computer memory locations (slots) into which records are inserted and from which they are later retrieved (read, nondestructively) or deleted. In the first model, which we call *hardpack*, a record is deleted by immediately physically removing it from the file and replacing it by the last record on its chain. In the second model, which we call *softpack*, a record is deleted by marking it, and a subsequent arriving record is inserted by writing it over the first marked record on its chain. The slots can be accessed singly or in buckets. In each case, we are interested in the average time elapsed during a successful search, and the average time elapsed during an unsuccessful search.

In Section 2 we formulate hardpack and softpack as queueing models. Our results can be used to quantify the degree of (obvious) superiority of hardpack over softpack (there are considerations other than length of search that might make softpack an attractive alternative); but, more importantly, they illustrate the power of teletraffic and queueing models in an area in which they have not been widely applied. In Section 3 we show that our dynamic hardpack model is essentially equivalent to Knuth's static model for hash files with separate chaining. Finally, in Section 4 we discuss some interesting qualitative properties of hardpack and softpack, and

we suggest some questions for future research. In particular, we observe that the length of successful search is insensitive to the distribution of residence times in both hardpack and softpack, and the length of unsuccessful search remains insensitive under hardpack, but not under softpack; in the latter case, constant residence times perform worse than exponential residence times. Also, unless the file is tightly packed the average length of an unsuccessful search (which goes to the end of a chain) is less than the average length of a successful search (which goes only to the "middle" of the chain).

2. The Models: Hardpack and Softpack

We assume that records arrive randomly in time for insertion into a file, where they reside for a random length of time before being deleted. Each record is identified by a key, which is mapped ("hashed") to one of the buckets in a primary file. When two different keys hash to the same bucket, a "collision" is said to occur. If there is not room for a colliding record in its primary bucket, the record is stored in a separate overflow area (which can also consist of buckets). Each primary bucket and its overflow buckets are linked together on a chain; and each bucket on the chain contains only those records that hashed to the primary bucket on that chain.

More precisely, we assume that (a) the sequence of times at which records arrive for insertion into the file can be described by a Poisson process, (b) the addresses generated by the hash function constitute a sequence of statistically independent numbers, (c) the residence times of the records are statistically independent of the arrival process and each other, and (d) each chain can grow without bound. Then, the structure of each chain is that of the $M/G/\infty$ queue; and the length of each chain (i.e., the number of records on the chain) is independent of the length of each other chain, and is described by a Poisson distribution. Therefore, we can restrict our attention to a single generic chain.

Let N be the number of records on the chain; then

$$P\{N = j\} = \frac{a^j}{j!} e^{-a}, \quad j = 0, 1, \dots \quad (1)$$

with $a = \lambda/\mu$, where λ is the rate at which the records arrive for insertion onto this chain, and μ^{-1} is the average residence time of the records (and a equals the mean number of records on the chain).

We consider two variations of this model: In *hardpack* a deleted record is immediately physically removed from the file and replaced by the last record on its chain. In *softpack* a deleted record is marked, and a subsequent arriving record is written over the first marked record on its chain. In both variations, the distribution of the length of the chain is given by eq. (1), and this result is well known to be *insensitive* to the particular distribution of the residence times.

Our objective is to calculate, for each model, the mean number of accesses (that is, the mean number of buckets probed) during an unsuccessful search and

during a successful search. In the case of an unsuccessful search, the chain is searched bucket-by-bucket until it is ascertained that the record in question is not on the chain. Thus, if the chain is not empty, the length of search (the number of accesses) equals the index of the bucket that contains the last record on the chain; and if the chain is empty, we take the length of search to be 1. For the hardpack model, this calculation is trivial, because every bucket on the chain except for the last bucket must be full. For the softpack model, this calculation is more complicated, because any bucket that precedes the last nonempty bucket on this chain can be partially full or empty; here we will appeal to some classical results from teletraffic theory that are not widely known to the computer-performance-analysis community. (We remark that Knuth defines the length of unsuccessful search to be the number of accesses required to search to the end of the chain; he does not "condition" on the knowledge that the record being inserted is not already on the chain.)

In the case of successful search, the definition of length of search is more problematical. If a record never changes its location as long as it remains on the chain, then the number of accesses required to locate the record equals the number of accesses required to insert that record. Thus, in the case of softpack, the analysis of length of successful search reduces to calculation of the index of the first idle trunk in an infinite-capacity group with ordered hunt (or, ordered entry). In the hardpack model, each new record is inserted into the slot that follows the last occupied slot on the chain; and then the record "migrates" down toward the beginning of the chain. We will take the view that if the record is known to be on the chain, then it is equally likely to be located in any of the $N + 1$ occupied slots, where N is given by eq. (1). (That is, at a retrieval epoch the additional number of records on the chain, not counting the record being retrieved, has the same distribution as the number of records on the chain just prior to the arrival epoch of that record.)

For the hardpack model, let X_H and Y_H be the number of accesses required for a successful search, and an unsuccessful search, respectively; similarly, let X_S and Y_S be the corresponding random variables for the softpack model. Then, clearly,

$$E(Y_H) = 1 \cdot P\{N = 0\} + 1 \cdot \sum_{j=1}^m P\{N = j\} + \sum_{k=2}^{\infty} k \sum_{j=m+(k-2)m_1+1}^{m+(k-1)m_1} P\{N = j\}, \quad (2)$$

where $P\{N = j\}$ is given by eq. (1); and where m is the capacity of the primary bucket, and m_1 is the capacity of each overflow bucket. (Note that the first term on the right-hand side of eq. (2) reflects the assumption that a single access is counted when an empty chain is "searched." Note also, as a check, that when $m = m_1 = 1$, then $E(Y_H) = e^{-a} + a$, as it should.)

In the case of successful search in hardpack, if N^* is the total number of records on the chain (resulting in

$k + 1$ nonempty buckets) at the retrieval epoch, then the expected value of the number of buckets that must be probed until the particular record is located is

$$E(X_H) = \sum_{j=0}^{m-1} E(X_H | N^* = j + 1) P\{N^* = j + 1\} + \sum_{k=1}^{\infty} \sum_{j=m+(k-1)m_1}^{m+km_1-1} E(X_H | N^* = j + 1) \times P\{N^* = j + 1\}.$$

Clearly, when $0 \leq N^* = j + 1 \leq m - 1$, then $E(X_H | N^* = j + 1) = 1$; and when $m + (k - 1)m_1 \leq N^* = j + 1 \leq m + km_1 - 1$, then

$$E(X_H | N^* = j + 1) = 1 \left(\frac{m}{j + 1} \right) + (2 + \dots + k) \left(\frac{m}{j + 1} \right) + (k + 1) \left(\frac{j + 1 - m - (k - 1)m_1}{j + 1} \right).$$

Now, using the identity $2 + \dots + k = \frac{1}{2}k(k + 1) - 1$ and the fact that $P\{N^* = j + 1\} = P\{N = j\}$ (given by eq. (1)), we have

$$E(X_H) = \sum_{j=0}^{m-1} P\{N = j\} + \sum_{k=1}^{\infty} \sum_{j=m+(k-1)m_1}^{m+km_1-1} \left\{ 1 \left(\frac{m}{j + 1} \right) + \left(\frac{k(k + 1)}{2} - 1 \right) \left(\frac{m_1}{j + 1} \right) + (k + 1) \left(\frac{j + 1 - m - (k - 1)m_1}{j + 1} \right) \right\} P\{N = j\}. \quad (3)$$

(These results are not pretty, but their numerical calculation is straightforward.)

For the softpack model, as indicated above, the index of the slot on the chain that is occupied by the record being retrieved corresponds to the index of the first idle trunk in an infinite-trunk group with ordered hunt (because the record is retrieved from the same slot into which it was originally inserted). Therefore, the number X_S of buckets that must be probed in order to locate the first empty slot (that is, the number of accesses required to implement a successful search) is given by

$$P\{X_S = k\} = \begin{cases} 1 - B(m, a), & k = 1 \\ B(m + (k - 2)m_1, a) - B(m + (k - 1)m_1, a), & k = 2, 3, \dots \end{cases}$$

where $B(n, a)$ is the Erlang loss formula for a group of n trunks that receives a erlangs of Poisson traffic. (Like X_H and Y_H , X_S is well known to be insensitive

to the distribution of residence times.) Therefore, the mean number of accesses $E(X_S) = \sum kP\{X_S = k\}$ is given by

$$E(X_S) = 1 + \sum_{j=0}^{\infty} B(m + jm_1, a), \quad (4)$$

an interesting-looking formula, which is easy to calculate numerically using the well-known recurrence (for $a > 0$),

$$B(n, a) = \frac{aB(n-1, a)}{n + aB(n-1, a)}, \quad n = 1, 2, \dots,$$

$$B(0, a) = 1.$$

As we did with hardpack, we assume in our softpack model that the system "knows" the address of the last occupied slot in the chain; hence, Y_S is the index of the last nonempty bucket (unless the chain is empty, in which case $Y_S = 1$). Then we can write (with $P\{Y_S > 0\} = 1$)

$$E(Y_S) = \sum_{k=0}^{\infty} P\{Y_S > k\} =$$

$$= 1 + \sum_{k=1}^{\infty} (1 - P\{Y_S \leq k\}). \quad (5)$$

Let $P_j(n, i)$ be the joint probability that in an $M/G/\infty$ system with ordered hunt, exactly n of the first j trunks are busy and an additional i trunks are busy. Let

$$Q_j = \sum_{n=0}^j P_j(n, 0), \quad (6)$$

where Q_j is the probability that there are no busy trunks beyond trunk j . Then

$$P\{Y_S \leq k\} = Q_{m+(k-1)m_1},$$

and eq. (5) can be written

$$E(Y_S) = 1 + \sum_{k=1}^{\infty} (1 - Q_{m+(k-1)m_1}). \quad (7)$$

Unlike the first three performance measures considered in this paper, the length of an unsuccessful search in softpack is *not insensitive* to the distribution of service times. Two cases are available: exponential service times (reported in 1937 by Kosten [7]) and constant service times (reported in 1971 by Burke [8]).

When the service times are exponentially distributed, Kosten gives (see, e. g., [9, pp. 139–147])

$$P_j(n, 0) = \phi_0(j) \sum_{\nu=0}^{\infty} (-1)^{\nu} \frac{a^{\nu}}{\nu!} \frac{\phi_{\nu}(n)}{\phi_{\nu+1}(j)\phi_{\nu}(j)},$$

$$j = 0, 1, \dots, \quad (8)$$

where

$$\phi_{\nu}(j) = \begin{cases} \frac{a^j}{j!} e^{-a}, & \nu = 0 \\ e^{-a} \sum_{i=0}^j \binom{\nu+i-1}{i} \frac{a^{j-i}}{(j-i)!}, & \nu = 1, 2, \dots \end{cases} \quad (9)$$

(These equations are those in [9, eqs. (3.33), (3.13)].) The functions $\{\phi_{\nu}(j)\}$ satisfy

$$\phi_{\nu+1}(j) = \phi_{\nu}(j) + \phi_{\nu+1}(j-1) \quad (10)$$

and

$$\sum_{n=0}^j \phi_{\nu}(n) = \phi_{\nu+1}(j). \quad (11)$$

Then, in view of eqs. (8) and (11), eq. (6) can be written

$$Q_j = \phi_0(j) \sum_{\nu=0}^{\infty} (-1)^{\nu} \frac{a^{\nu}}{\nu!} \frac{1}{\phi_{\nu}(j)}, \quad (12)$$

and $E(Y_S)$ can be calculated from eqs. (7), (12), and (10).

When the service times are constant, Burke gives (his eq. (4))

$$Q_j = \frac{e^{-a}}{\sum_{i=0}^j \frac{a^i}{i!}} \left\{ \sum_{n=0}^j \frac{(2a)^n}{n!} + \sum_{n=j+1}^{2j} \frac{a^n}{n!} \times \right. \quad (13)$$

$$\left. \times \left[\sum_{\nu=n-j}^j \binom{n}{\nu} - \binom{n}{j+1} (2j-n+1) \right] \right\}.$$

3. Knuth's Model

Knuth's 1973 model (considered by computer scientists to be classic) assumes that there are a fixed number of records, whose keys are distributed randomly among the possible hash-function addresses (that is, among the chains). Then the distribution of the number of records on a chain is binomial; for appropriate values of the parameters, this distribution would be approximated by the Poisson distribution, given by eq. (1), where the parameter a is the ratio of the number of records in the file to the number of possible hash-function addresses (i.e., the average number of records per chain). Thus Knuth's model, which is static, yields the same (Poisson) distribution for the number of records on a typical chain as do our dynamic models (both hardpack and softpack), which assume that the records "arrive" for insertion into the file according to a Poisson process and are deleted from the file after a "service time" whose distribution is arbitrary. This broadens the applicability of Knuth's conclusions.

To calculate the expected length C_n of a successful search when the number of records in the file is n ,

Knuth argues that

$$C_n = \frac{1}{n} \sum_{k=0}^{n-1} C'_k \quad (14)$$

(eq. (14) is that of Knuth [1, eq. (36), p. 528]), where C'_k denotes the expected length of an unsuccessful search when the number of records in the file is $k + 1$: $1/n$ is the probability that the record being retrieved was the i^{th} record of the n records (all present initially) to be inserted, and C'_{i-1} is the expected number of probes required to reach the end of the chain onto which the i^{th} record was inserted (if there are already $i - 1$ records inserted into the file). Knuth gives formulas (based on "some very interesting mathematics") that embody a combinatorial analysis of C'_k , from which he then calculates C_n according to eq. (14) (see Knuth [1, eqs. (55)-(58), pp. 536-537]). He provides numerical values (in [1, Tables 2 and 3, p. 535]) for C'_k and C_n .

As one would expect (because the Poisson distribution can be obtained as a limit of the binomial), our values for unsuccessful search, calculated from eq. (2), are in agreement (in all decimal places shown) with Knuth's Table 2; interestingly, we obtain the same agreement for successful search, calculated from eq. (3), with Knuth's Table 3. (We say "interestingly" because Knuth's static model does not account for the evolution of the file as a consequence of a stochastic process of insertions and, especially, deletions). Again, this illustrates the robustness of Knuth's static model by showing its equivalence to our dynamic hardpack model. (Of course, he cannot address the softpack model, because he does not deal explicitly with deletion operations.)

4. Discussion

It is interesting and useful to know that models and formulas whose development was motivated by teletraffic engineering can find direct application in the performance analysis of database systems. The powerful queueing-theory machinery assembled by Pollaczek and his colleagues over the past 80 years thus provides a ready-made theoretical framework for performance modeling, one that would otherwise require time-consuming and expensive independent development.

In the case of the particular models considered in this paper, some qualitative insights can be gleaned: Knuth's static model of hash files with separate chaining can be broadened to include the case where the insertions and deletions occur according to a stochastic process. The ability to include deletions in the analysis permits two variations of the basic model, hardpack and softpack; and the degree of (obvious) superior performance (in terms of length of search) of hardpack over softpack can be quantified. Interestingly, the queueing-theory analysis shows that for hardpack (with Poisson arrivals) the performance is insensitive to the form of the particular distribution of residence times. But not

so for softpack; somewhat surprisingly (as foreseen in 1971 by Burke), the average length of an unsuccessful search is longer when the residence times have low variability rather than high variability. But (again surprising, in view of this observation) the average length of a successful search in softpack remains insensitive to the form of the residence-time distribution. In both models, numerical calculations show that, if the chain is not too long, the average length of an unsuccessful search, which is defined to be the number of accesses required to reach the end of the chain, is less than the average length of a successful search, which goes only to the "middle" of the chain. (This phenomenon is reflected in Knuth's tables, but he did not remark on it.) Apparently, this is an example of "batch biasing," a discrete version of the famous waiting-time (or inspection) paradox, so dear to the hearts of queueing theorists.

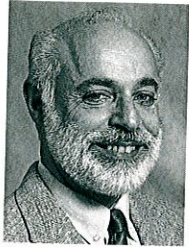
We intend to investigate the numerical properties of these models (with parameter values, bucket sizes, etc., that are of interest to database performance analysts) in a future paper. Also, we intend to apply the methods of queueing theory to the analysis of other file structures, such as open addressing (Knuth [1, p. 536]) and linear hashing (Litwin [10]).

Acknowledgement

This research was supported in part by the National Science Foundation under grant DDM-9001751.

References

- [1] Knuth, D. E.: The Art of computer programming, Vol. III: Sorting and searching. New York: Addison-Wesley, 1973.
- [2] Aldous, D.; Hofri, M.; Szpankowski, W.: Maximum size of a dynamic data structure: Hashing with lazy deletion revisited. *SIAM J. Computing* **21** (1992), 713-732.
- [3] Morrison, J. A.; Shepp, L. A.; VanWyk, C. J.: A queueing analysis of hashing with lazy deletion. *SIAM J. Computing* **21** (1992), 1155-1164.
- [4] Mendelson, H.; Yechiali, U.: Performance measures for ordered lists in random-access files. *J. ACM* **26** (1979), 654-667.
- [5] Cooper, R. B.; Solomon, M. K.: The average time until bucket overflow. *ACM Trans. Database Systems* **9** (1984), 392-408.
- [6] Larson, P. A.: Analysis of index-sequential files with overflow chaining. *ACM Trans. Database Systems* **6** (1981), 671-680.
- [7] Kosten, L.: Über Sperrungswahrscheinlichkeiten bei Staffelschaltungen. *Elektr. Nachrichten-Technik* **14** (1937), 5-12.
- [8] Burke, P. J.: The overflow distribution for constant holding time. *Bell Syst. Tech. J.* **50** (1971), 3195-3210.
- [9] Cooper, R. B.: Introduction to queueing theory. 2nd ed. Amsterdam: North-Holland (Elsevier), 1981, Republished 1990, CEEPress, The George Washington University.
- [10] Litwin, W.: Linear hashing: A new tool for file and table addressing. *Proc. 6th Int. Conf. Very Large Data Bases*, Montreal, Oct. 1980.



Robert B. Cooper received the BS degree in 1961 from Stevens Institute of Technology, and the MS (Systems Engineering and Operations Research, 1962) and PhD (Electrical Engineering, 1968) from the University of Pennsylvania. From 1961 to 1969 he was a Member of Technical Staff at AT&T Bell Laboratories, and since then has held faculty positions at Georgia Institute of Technology, University of

Michigan, and New Mexico Institute of Mining and Technology. He is currently Professor of Computer Science and Engineering at Florida Atlantic University, where he has been since 1978. His primary research interest is queueing theory and its applications in performance analysis of telecommunications and computer systems.



Martin K. Solomon received the BA in Mathematics from Rutgers University, the MS in Computer Science from New York University, and the PhD in Mathematics from Stevens Institute of Technology. He has been an Assistant Professor at the Graduate School of Management, Rutgers University, and a Member of Technical Staff at AT&T Bell Laboratories. He is currently an Associate Professor of Computer Science and Engineering at Florida Atlantic University. His present research interests include structural complexity theory, the philosophical aspects of computability theory, and the theory, implementation, and performance analysis of database systems.

ence and Engineering at Florida Atlantic University. His present research interests include structural complexity theory, the philosophical aspects of computability theory, and the theory, implementation, and performance analysis of database systems.