Short Communications
Numerical Mathematics

Quadratic Search for Hash Tables of Size p^n

A. Frank Ackerman
Bell Telephone Laboratories

Key Words and Phrases: hashing, quadratic search
CR Categories: 4.10

It has previously been claimed [1 and 2] that the quadratic hash table search method of Maurer cannot usefully be applied to tables of size 2^n. This is not so; the method can in fact be applied to tables of size p^n for any prime p. It is shown below that rather simple conditions on the coefficients suffice to guarantee that all table locations will be examined once and only once. Specifically, if the equation is

$$k + bi^2 + ai \bmod p^n (*)$$

where k is the initial hash address and $0 \leq i < p^n$, then, if p divides b but not a, the range of values is all the least positive residues of p^n. To prove that all values are covered, we consider some fixed value, say $k + bi_0^2 + ai_0 \bmod p^n$ and ask, what conditions must be true if the congruence equation

$$k + bi^2 + ai \equiv k + bi_0^2 + ai_0 \bmod p^n$$

is to have solutions i, $0 \leq i < p^n$, other than i_0?

Now this is entirely equivalent to asking for solutions j, $0 \leq j < p^n$, to

$$k + b(i_0 + j)^2 + a(i_0 + j) \equiv k + bi_0^2 + ai_0 \bmod p^n.$$

When this is simplified, we see that j is required to satisfy

$$j(2bi_0 + bj + a) \equiv 0 \bmod p^n.$$

But under the above conditions on a and b, $2bi_0 + bj \equiv 0 \bmod p$, and hence $2bi_0 + bj + a$ will not contain any factors of p for any values of j, $0 \leq j < p^n$. Since j can contain at most $n - 1$ factors of p, the values of (*) will all be distinct for $0 \leq i < p^n$.

These conditions degenerate to a linear search for $n = 1$. In this case one can follow Mauer's suggestion [1] and examine at most half the table positions, or use Day's technique [3] to cover the whole table. To prevent secondary clustering [4] for the $n > 1$ case, b can, of course, be varied by taking it to be pk', where k' is a function of the key and independent of k.

Author's address: Bell Telephone Laboratories, P.O. Box 2020, New Brunswick, NJ 08903.

Received February 1973; revised October 1973

References
1. Maurer, W.D. An improved hash code for scatter storage. *Comm. ACM 11*, 1 (Jan. 1968), 35–38.
2. Gries, D. *Compiler Construction for Digital Computers.* Wiley, New York, 1971.
3. Day, A.C. Full table quadratic searching for scatter storage. *Comm. ACM 13*, 8 (Aug. 1970), 481–482.
4. Bell, J.R. The quadratic quotient method: a hash code eliminating secondary clustering. *Comm. ACM 13*, 2 (Feb. 1970), 107–109.

Scientific Applications

Emotional Content Considered Dangerous

Stephen W. Smoliar
University of Pennsylvania

Key Words and Phrases: artificial intelligence, heuristic programming, models of cognitive processes, computer music, computer composition, music theory
CR Categories: 3.44, 3.65

I had hoped that Moorer's rebuttal to my short communication in the November 1972 Communications would close the debate on a topic which, like the computer itself, has provoked an inordinately large quantity of unqualified argument. Unfortunately, the short communications by McMorrow and Wexelblat in the May 1973 Communications lead me to believe that my position is still grossly misunderstood. Therefore, allow me to clarify these matters.

First and foremost, I did not "completely discount the work of Moorer merely for the sake of a concept not completely understood and almost completely disregarded by present-day computer composers" (by which I assume McMorrow means "the introduction of 'emotional content' "). My criticism of Moorer's original paper was that through lack of proper modeling techniques, Moorer presented musical composition as being far more "unreachable" than it really is. Nor does this imply, as Wexelblat seems to think, that I propose the software composer as a viable competitor to "its meatware rival." Speaking as a "meatware composer," I prefer to regard my software as a partner rather than as a rival (see the descriptions of some of my pieces in [1]). Rather than pursuing "the practical

Author's address: The Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia, PA 19174.

implementation of machine-oriented music generators," I have sought to design and implement a device which I and other composers might use as conveniently as a piano, but to obtain results of a higher order of complexity.

As to "the composer's judicious use of the emotional content" by which we, the audience, "subjectively rate him," I must take an opposing stand, and I don't think I am alone. Speaking as a composer who has used conventional instruments and electronic synthesizers as well as computers, I find that "emotional content" is very much an ex post facto phenomenon. I never fabricate it, rarely "use" it (at least consciously), and always discover it. "Variations on a Theme of Steve Reich" is far more emotional than the description in [1] would lead one to believe, not because I *made* it that way but because it *turned out* that way. For me, the art of composition consisted in properly formulating the framework and then adjusting the details within this framework. When I grasped the emotional impact of the piece (which is to say, when the piece discovered what it wanted to be), there was very little "adjustment" left to be done.

Received October 1973

References

1. Smoliar, Stephen W. Music theory—a programming linguistic approach. Proc. 25th Ann. Conf. ACM, New York, 1972, pp. 1001–1014.

Computer Systems

A Note on a Combinatorial Problem of Burnett and Coffman

Harold S. Stone
Stanford University

Key Words and Phrases: memories, interleaving, derangements, rencontres, combinatorial analysis
CR Categories: 5.39, 6.34

1. Introduction

Burnett and Coffman [1] analyze interleaved memory systems to determine mean memory bandwidth. Their analysis depends on the numbers $C_{n,k}$ where $C_{n,k}$ is the number of sequences of length k drawn from the set of integers $\{1, 2, \ldots, n\}$ such that: (i) each sequence has k distinct integers; (ii) the initial integer of each sequence is 1; and (iii) no sequence contains the subsequence $(\ldots, i, i+1, \ldots)$ for any i.

The third property states that each sequence counted by $C_{n,k}$ has no successor transitions, i.e. subsequences of the form $(\ldots, i, i+1, \ldots)$. In the Burnett-Coffman problem each sequence represents a collection of k distinct memories that are the targets of k distinct address references. The reason for the restriction on successor transitions is due to the Markov process that they assume to generate the address references. If a reference to memory module i occurs, then with probability α, the next request goes to module $(i+1) \bmod n$; the next request goes to any of n other modules with probability $(1-\alpha)/(n-1)$. They show that the entire analysis depends only on the sequences counted by $C_{n,k}$. Note that the successor of memory module n is memory module 1, so that the transition $(\ldots, n, 1, \ldots)$ is a successor transition. However, by restricting our attention to sequences that begin with a 1, we need never treat transitions of the form $(\ldots, n, 1, \ldots)$, and we enumerate precisely $(1/n)$-th of the sequences of interest.

The central point of this note is that the problem of determining $C_{n,k}$ is isomorphic to the well-known combinatorial problem of derangements (see Liu [2]). A derangement of n letters is a permutation on n letters in which no letter is mapped onto itself. We show that $C_{n,n}$ is equal to the number of derangements on $n-1$ letters. More generally we define a k-derangement on n letters to be a mapping from the set $\{1, 2, \ldots, k\}$ into the set $\{1, 2, \ldots, n\}$ such that the k images are distinct, and no element is mapped back onto itself. Then there is one-to-one correspondence between the $(k-1)$-derangements on $n-1$ letters and the sequences counted by $C_{n,k}$.

There are various ways of establishing the one-to-one correspondence. We might proceed by finding a one-to-one correspondence between the $C_{n,k}$ sequences and $(k-1)$-derangements on $n-1$ letters, but this is rather tedious, even though many such maps exist. Since the computation of the number of derangements on n letters is very simple, we proceed by determining $C_{n,k}$ by direct enumeration. The correspondence between the $C_{n,k}$ sequences and $(k-1)$-derangements then follows because the enumeration techniques and the numbers obtained from the enumerations are identical for the two problems.

2. The Derivation of $C_{n,k}$

The calculation of $C_{n,k}$ uses an inclusion-exclusion argument. In this discussion we use the notation $(n)_k$ to denote the falling factorial $n(n-1)(n-2)\ldots(n-k+1)$, with $(n)_0$ defined to be 1. Also, in a sequence of length k, a transition of the form $(\ldots, i, i+1, \ldots)$ is called a *successor* transition. We compute

This work was supported by the National Science Foundation under Grant GJ-1180. Author's address: Digital Systems Laboratory, Departments of Computer Science and Electrical Engineering, Stanford University, Stanford, CA 94305.