allocated memory is much less full using external chaining because of the separate overflow area. From the practical point of view the two load factors approximately satisfy the equation $\alpha = \alpha' \tilde{p}_1$, where α and α' are, respectively, the load factors in open addressing and in external chaining [5].

Since, in practice, both overflow and prime areas must be fixed in advance, for external chaining, a maximum overflow area has to be reserved. Therefore, it is more realistic to compare the extra access times belonging to the load factors α' and $\alpha(1) = \alpha\tilde{p}_1(1)$, where $\tilde{p}_1(1)$ is the probability $\tilde{p}_1$ calculated for $\alpha' = 1$.

It can be seen from Table V that if $R \geqq 2$, then open addressing is always more efficient than external chaining. For $R = 1$, the two methods are roughly equal.

Figure 3 shows the effect of bucket size on the extra access time for corresponding load factors. It can be seen that, contrary to popular belief, small bucket sizes (1–2) are almost always superior to bigger ones.

Received 4/80; revised and accepted 4/81

References

1. Introduction to IBM DASD and Organization Methods GC-20-1649, IBM Tech.Man.
2. Knott, G.D. Hashing functions. *The Computer Journal 18*, 3 (1973) 265–278.
3. Knuth, D.E. *The Art of Computer Programming.* Addison-Wesley, Reading, Mass. 1973.
4. Lum, V. Y. General performance analysis of key-to-address transformation methods using an abstract file concept. *Comm. ACM 16*, 10 (Oct. 1973), 603.
5. Lum, V.Y., Yuen, P.S.T. and Dodd, M. Key-to-address transformation techniques. *Comm. ACM 14*, 4 (Apr. 1971) 228–239.
6. Martin, J. *Data Base Organization.* Prentice Hall, Englewood Cliffs, NJ, 1973.
7. Peterson, W.W. Addressing for random-access storage. *IBM J. Res. Develop. 1*, 2 (Apr. 1957), 130–146.
8. Quittner, P. and Kotsis, D. Comparison of different disk searching methods. *Software Practice and Experience 8*, 6 (Nov.–Dec. 1978) 673–679.
9. Quittner, P. *Problems, Programs, Processing Results.* Akadémiai Kiadó, Budapest, A. Hilger, Bristol, England 1977.
10. Quittner, P. and Kotsis, D. Vergleich von Datenzugriffsarten. *Rechnentechnik und Datenverarbeitung 14*, 31 (1977) 35.
11. Quittner, P. and Kotsis, D. Computing Science for Systems Analysis. Oour Institut za Organizaciju Poslavanja, Subotica, Pokrajinski Savot za Informatiku Novi Sad, 1976, and Akadémiai Kiadó, Budapest (to be published).
12. Rényi, A. *Probability Theory*, North Holland, Amsterdam, The Netherlands 1970.

Technical Note
Programming Techniques and Data Structures
M. Douglas McIlroy Editor

On Sharing Secrets and Reed–Solomon Codes

R.J. McEliece and D.V. Sarwate
University of Illinois at Urbana–Champaign

Shamir's scheme for sharing secrets is closely related to Reed-Solomon coding schemes. Decoding algorithms for Reed-Solomon codes provide extensions and generalizations of Shamir's method.

Key Words and Phrases: secret sharing systems, Reed-Solomon codes, cryptography, key management, privacy, interpolation.

CR Categories: 3.79, 5.39, 5.6

We wish to remark that Shamir's scheme [5] for sharing a secret among many persons is very closely related to Reed–Solomon coding schemes [1–3], and that there are several advantages to discussing the former in the context of the latter. Let $(\alpha_1, \alpha_2, \ldots, \alpha_{r-1})$ be a fixed list of the nonzero elements in a finite field F with r elements. In one form of Reed–Solomon coding, an information word $\mathbf{a} = (a_0, a_1, \ldots, a_{k-1})$, $a_i \in F$, is encoded into the codeword $\mathbf{D} = (D_1, D_2, \ldots, D_{r-1})$, where $D_i = \sum_{j=0}^{k-1} a_j \alpha_i^j$. The "secret" is $a_0 = -\sum_{i=1}^{r-1} D_i$, while the "pieces" of the secret are the D_i's. Suppose that s of the pieces are given *but t of these are in error.* Then by applying an errors-and-erasures decoding algorithm [1, 2, 6], it is possible to recover $\mathbf{D}$ and $\mathbf{a}$ (and hence a_0) provided that $s - 2t \geq k$. Shamir's scheme corresponds to a special case of this result where r is a prime, $\alpha_i = i$, and $t = 0$. However, the more general case is also of interest. Consider a situation where an opponent's purpose is served if legitimate users are *denied access* to the secret. To this end, the opponent may tamper (either covertly or by suborning executives) with the pieces D_i. It is easily seen that if t pieces have been tampered with, the secret can be accessed by legitimate users provided

This research was supported by the Joint Services Electronics Program under Contract N00014-79-C-0424.

Authors' present address: Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 1101 W. Springfield Avenue, Urbana, IL 61801

that at least $k + t$ valid pieces are available. In particular, for a (k, n) threshold scheme, the opponent must tamper with more than $\lfloor (n - k)/2 \rfloor$ pieces to ensure that the secret is inaccessible. The errors-and-erasures algorithm is also useful in the more mundane situation where there are imperfections in the storage medium used to record the piece D_i. In this case, what is read out may differ from what was stored, and the piece D_i becomes invalid without the assistance of any sinister external agency. In either situation, the errors-and-erasures algorithm will point out what invalid D_i pieces were submitted to it. Since efficient $O(n \log^2 n)$ [4] as well as standard $O(n^2)$ [1, 6] errors-and-erasures algorithms are known, all this is available without any significant increase in the complexity over that of Shamir's scheme. Indeed, for $r = 2^m$, Reed-Solomon coders have been implemented in hardware and software, and are available commercially.

When the secret to be shared is rather long, the fact that each piece D_i is as long as the secret may be a nuisance. In such a case, an alternative implementation of a Reed-Solomon coding scheme enables us to achieve a certain amount of data compression at the cost of some degradation in security. Let $\mathbf{b} = (b_1, b_2, \ldots, b_k)$ be the secret. Then there exists a unique codeword $\mathbf{D}$ in the Reed-Solomon code with $D_1 = b_1$, $D_2 = b_2$, $\ldots$, $D_k = b_k$; indeed, $\mathbf{D}$ can be found by Lagrange interpolation as suggested by Shamir [5] or by standard Reed-Solomon encoding algorithms [1, 2]. Only the $r - 1 - k$ pieces $D_{k+1}, \ldots, D_{r-1}$ are available for distribution to those sharing the secret. As before, at least k (ungarbled) pieces are required to recover the secret. Furthermore, if only $k - 1$ pieces are available and a wrong guess is made for the missing kth piece, then it follows from known properties of Reed-Solomon codes that the resulting "secret" $\mathbf{b}'$ differs from $\mathbf{b}$ in all k positions! However, an opponent who has discovered $k - 1$ pieces knows that the secret must be one of only r different vectors $\mathbf{b}'$ whereas a priori the secret could have been any one of r^k different vectors. On the other hand, the pieces D_i are shorter than the secret by a factor of k. We note that the errors-and-erasures algorithm can be applied to this Reed-Solomon coding scheme also, and provides protection against opponents who wish to deny access as well as against defects in memories. As a final remark, we mention that the version of Reed-Solomon coding which maps $\mathbf{a}$ into $\mathbf{D}$ also can provide data compression but is not as secure. As before, an opponent who had discovered $k - 1$ pieces knows that the secret $\mathbf{a}$ must be one of only r different vectors. Unfortunately, however, it is possible that all r vectors are identical in several positions, and thus the corresponding portion of the secret is revealed, even though the entire secret is still inaccessible.

Received 3/80; accepted 4/81

References

1. Berlekamp, E.R. *Algebraic Coding Theory.* McGraw-Hill, New York, 1968.
2. McEliece, R.J. *The Theory of Information and Coding.* Addison-Wesley, Reading MA, 1977.
3. Reed, I.S. and Solomon, G. Polynomial codes over certain finite fields. *J. SIAM 8*, 2 (June 1960), 300–304.
4. Sarwate, D.V. On the complexity of decoding Goppa codes. *IEEE Trans. Inform. Theory 23*, 4 (July 1977), 515–516.
5. Shamir, A. How to share a secret. *Comm. ACM, 22*, 11 (Nov. 1979), 612–613.
6. Sugiyama, Y., Kasahara, M., Hirasawa, S., and Namekawa, T. An erasures-and-errors decoding algorithm for Goppa codes. *IEEE Trans. Inform. Theory 22*, 2 (Mar. 1976), 238–241.