Reducing the Complexity of Calculating Syndromes for Error-Correcting Codes

L. H. Harper and J. E. Savage[1]
Communications Systems Research Section

The calculation of the syndrome—the first step performed by all decoders of linear codes—can require a number of logical operations which grows faster than the square of block length. It is shown that the complexity of syndrome calculation can be reduced for many linear codes by a factor of log of the code block length and that Hamming codes can be decoded with combinational machines having a number of logic elements which is linear in block length.

I. Introduction

It has been suggested that error-correcting coding be used to improve the reliability of Ground Communications Facility (GCF) data transfer. However, if such coding is to be used, the problem of real-time decoding must first be dealt with; this problem will be especially acute in the 50-kbits/s wideband mode. This note shows how decoding complexity can be decreased for many important coding schemes.

Every linear (parity-check) code has a parity check matrix H associated with it. If the code words $\mathbf{x}_1, \ldots, \mathbf{x}_M$ are N-tuples over $GF(q)$, then H is an $N \times (N-K)$ matrix over $GF(q)$ where K is the number of information digits needed to represent the code word and $N-K$ is the number of dependent digits in a code word. Also, $M = q^K$ and every code word satisfies the equation

$$\mathbf{x}_i H = \mathbf{0}$$

Let $\mathbf{y}$ be the received sequence when $\mathbf{x}_i$ is transmitted and let $\mathbf{e} = \mathbf{y} - \mathbf{x}_i$ be the error sequence associated with $\mathbf{x}_i$. The *syndrome* $\mathbf{s}$ associated with $\mathbf{y}$ and $\mathbf{e}$ is

$$\mathbf{s} = \mathbf{y} H = \mathbf{e} H$$

and $\mathbf{s}$ is a compact reflection of the channel errors.

In this note we show that the calculation of $\mathbf{s}$ can be reduced for many codes by making use of the structure of H. We begin with an examination of Hamming codes.

II. Hamming Codes

The parity-check matrix H_m of the Hamming code (Ref. 1) is $N = 2^m - 1$ by m dimensional binary matrix which contains all binary m-tuples as rows except for the

[1]Division of Engineering, Brown University, and consultant, Communications Systems Research Section.

zero m-tuple. It is easily shown that each column of H_m contains 2^{m-1} *ones,* so that the straightforward calculation of each digit of **s** would require $2^{m-1} - 1$ modulo-2 sums of pairs for a total of $m(2^{m-1} - 1)$ additions. This number grows as $N \log_2 N$. We shall show that this number can be reduced to linear in N.

Theorem 1

The calculation of syndromes for a Hamming code of length $N = 2^m - 1$ can be accomplished with $2[2^m - (m + 1)]$ mod-2 additions, and at least $2^m - 2$ additions are required.

Proof

There is no loss of generality in assuming that the rows of H_m (which are m-tuples) are listed in order of increasing integers which they represent in dyadic form. Form H_m^* from H_m by adding the zero m-tuple as the first row. For example,

$$H_3^* = \begin{bmatrix} 0\ 0\ 0 \\ 0\ 0\ 1 \\ 0\ 1\ 0 \\ 0\ 1\ 1 \\ 1\ 0\ 0 \\ 1\ 0\ 1 \\ 1\ 1\ 0 \\ 1\ 1\ 1 \end{bmatrix} \quad H_3 = \begin{bmatrix} 0\ 0\ 1 \\ 0\ 1\ 0 \\ 0\ 1\ 1 \\ 1\ 0\ 0 \\ 1\ 0\ 1 \\ 1\ 1\ 0 \\ 1\ 1\ 1 \end{bmatrix}$$

Note that H_{m+1}^* can be formed from H_m^* as indicated below:

$$H_{m+1}^* = \begin{bmatrix} 0 & \\ 0 & H_m^* \\ \cdot & \\ \cdot & \\ \cdot & \\ 0 & \\ \cdots & \cdots \\ 1 & \\ \cdot & H_m^* \\ \cdot & \\ 1 & \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}0\\0\\\cdot\\\cdot\\\cdot\\0\end{matrix}}\right\} 2^m \\ \left.\vphantom{\begin{matrix}1\\\cdot\\\cdot\\1\end{matrix}}\right\} 2^m \end{matrix}$$

The number of modulo-2 additions to multiply H_{m+1} on the left, P_{m+1}, is the number to multiply with H_{m+1}^*. But, this is twice P_m plus the number to add each half of columns 2 through $m + 1$ of H_{m+1}^*, namely, m plus the number to multiply by the first column. The recursive construction of H_{m+1}^* shows that the bottom quarter of the second column of H_{m+1}^*, namely, the lower half of the first column of H_m^*, contains *ones,* so that the partial sum of the $2^m/2$ last components of **y** computed for the second column of H_{m+1}^* can be used to compute its first column. Using partial sums computed for the third and later columns, the first column can be computed with an additional m additions.

Then,

$$P_{m+1} = 2P_m + 2m$$

and it is easily shown that $P_2 = 2$. This is a linear difference equation with homogeneous solution $c2^m$ and particular solution $-2(m + 1)$. Therefore,

$$P_m = c2^m - 2(m + 1)$$

and $c = 2$ for $P_2 = 2$. Or

$$P_m = 2(2^m - (m + 1)).$$

To show that at least $2^m - 1$ additions are necessary to compute **s**, we observe that in computing $(y_1, y_2, \ldots, y_{2^m})H_m^*$ the sums $y_2, y_{2+1} + y_{2^2}, y_{2^2+1} + \ldots + y_{2^3}, \ldots, y_{2^{m-1}} + \ldots + y_{2^m}$ must be formed and that these are sums of overlapping variables. Also, each sum except the last is added to other partial sums. Therefore, the number of additions is at least

$$\sum_{j=1}^{m-1} 2^{j-1} + 2^{m-1} - 1 = 2^m - 2$$

This completes the proof of the theorem.

This reduction by a factor of $\log_2 (N + 1)$ in the complexity of syndrome calculations can be carried over to some BCH codes, as shown next.

III. Binary BCH Codes

A t-error-correcting BCH code (Ref. 1) over $GF(2)$ has a parity check matrix

$$H = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \alpha & \alpha^3 & & \alpha^{2t-1} \\ \alpha^2 & (\alpha^3)^2 & & (\alpha^{2t-1})^2 \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \alpha^{N-1} & (\alpha^3)^{N-1} & & (\alpha^{2t-1})^{N-1} \end{bmatrix}$$

where 1 and α are elements of $GF(2^m)$ and N is the multiplicative order of α.

Theorem 2

The syndrome $\mathbf{s}$ of a binary, t-error-correcting BCH code can be computed using $2t(2^m - (m+1))$ mod-2 additions when α is primitive and the multiplicative orders of $\alpha^3, \alpha^5, \ldots, \alpha^{2t-1}$ are all relatively prime to $N = 2^m - 1$, the block length of the code.

Proof

Each element of $GF(2^m)$ can be represented by a binary m-tuple. Under the conditions of the theorem, each element $\alpha, \alpha^5, \ldots, \alpha^{2t-1}$ is primitive in $GF(2^m)$ and each column contains all the nonzero m-tuples. Invoking Theorem 1, the result follows.

Under the conditions of the theorem the number of mod-2 additions to form H directly, without using partial sums for various columns, would be $mt(2^{m-1} - 1)$. Thus, a savings of a factor of about $m/4$ can be achieved.

When the conditions of Theorem 2 are not met, the bound of Theorem 2 may not apply. The interested reader can satisfy himself that 47 additions will be needed, using techniques of this note, for the (15,7) BCH code while the bound of Theorem 2 would predict 44. In this case, α^3 is not primitive in $GF(2^4)$.

IV. On Decoding Hamming Codes

Hamming codes can be decoded with a logic circuit containing a number of logic elements proportional to the block length N, as is now shown. The Hamming codes correct all single errors, and decoding is done by changing the ith received digit if $\mathbf{s}$ is equal to the ith row of H.

The circuit which generates correction signals from a syndrome vector computes all but one of the terms of the form $s_1^{c_1} \cdot s_2^{c_2} \cdot \ldots \cdot s_m^{c_m}$ where $\cdot$ denotes AND, $(c_1, c_2, \cdots, c_m)$ is a binary m-tuple, and $s_i^1 = s_i$, $s_i^0 = \bar{s}_i$, the INVERSE of s_i. The only term not computed is $\mathbf{s}_1 \cdot \ldots \cdot \mathbf{s}_m$. These terms are known as minterms, and it can be shown by induction that they can all be realized using $2(2^m - 1)$ logic elements of the type AND, OR, INVERSE (Ref. 2).

Thus, with a total number of logic elements proportional to N, syndromes can be computed and correction signals generated.

V. Conclusion

The reductions in decoder complexity demonstrated in this note might also be achieved for many other codes.

References

1. Peterson, W. W., *Error-Correcting Codes.* M.I.T. Press and Wiley and Sons, New York, 1961.
2. Savage, J. E., "The Effective Computing Power of Computer Memory," in *The Deep Space Network,* Space Programs Summary 37-64, Vol. II, pp. 30–32. Jet Propulsion Laboratory, Pasadena, Calif., Aug. 31, 1970.