

Innovative Improvement of Data Storage Using Error Correction Codes

Aru Okereke Eze, Emole Chukwumeka Enoch, Enyenihi Henry Johnson

Department of Computer Engineering Michael Okpara University of Agriculture, Umudike, Umuahia, Abia State, Nigeria

Department of Electrical/Electronics Engineering Akwa Ibom State University, Uyo, Akwa Ibom State, Nigeria

ABSTRACT: *The need for codes capable of detecting and correcting byte errors are extremely important since many memory systems use b-bit-per-chip organization. Error correcting codes have been used to enhance the reliability and data integrity of computer memory systems (information-bearing objects); data-storage systems are natural consumers of information theoretic ideas. For many issues in data-storage systems, the best trade-off between cost, performance and reliability, passes through the application of error-correcting codes. Error correcting codes that are specialized for data-storage systems is the subject studied by this research. This work describes in some depth the applications of the theory of error-correcting codes, in particular Reed-Solomon codes, for optical media storage, namely, CDs, DVDs etc.*

Keywords: *Error correction code, storage, interpolation, etc.*

I. INTRODUCTION

The theory of error detecting and correcting codes is that branch of engineering and mathematics which deals with the reliable transmission and storage of data. Information media are not 100% reliable in practice, in the sense that noise (any form of interference) frequently causes data to be distorted. To deal with this undesirable but inevitable situation, some form of redundancy is incorporated in the original data. With this redundancy, even if errors are introduced (up to some tolerance level), the original information can be recovered or at least the presence of errors can be detected. We saw in class how adding to the original message the parity bit or the arithmetic sum allows the detection of a (certain type of) error. However, that kind of redundancy doesn't allow for the correction of the error. Error-correcting codes do exactly this: they add redundancy to the original message in such a way that it is possible for the receiver to detect the error and correct it, recovering the original message. This is crucial for certain applications where the re-sending of the message is not possible (for example, for interplanetary communications and storage of data). The crucial problem to be resolved then is how to add this redundancy in order to detect and correct as many errors as possible in the most efficient way.

1.1 KEY CONCEPTS

The error detecting and correcting capabilities of a particular coding scheme is correlated with its code rate and complexity. The code rate is the ratio of data bits to total bits transmitted in the code words. A high code rate means information content is high and coding overhead is low. However, the fewer bits used for coding redundancy, the less error protection is provided. A tradeoff must be made between bandwidth availability and the amount of error protection required for the communication. [1]

1.2 TRADE OFFS

When choosing a coding scheme for error protection, the types of errors that tend to occur on the communication channel must be considered. There are two types of errors that can occur on a communication channel: random bit errors and burst errors. A channel that usually has random bit errors will tend to have isolated bit flips during data transmissions and the bit errors are independent of each other. A channel with burst errors will tend to have clumps of bit errors that occur during one transmission. Error codes have been developed to specifically protect against both random bit errors and burst errors. [2]

Real-time systems must consider tradeoffs between coding delay and error protection. A coding scheme with high error-correcting capabilities will take longer to decode and will be non-deterministic. This could cause a missed deadline and failure if a piece of needed information is stuck being decoded.

In embedded systems, error coding is especially important because the system may be used in critical applications and cannot tolerate errors. Coding schemes are becoming increasingly complex and probabilistic, making implementation of encoders and decoders in software attractive. Since processing power is relatively fast and cheap, software coding is more feasible. However, software is much more prone to design defects and

errors, making the coding algorithm less reliable. Pushing complexity into software introduces more errors in design and implementation. [3]

II. STORAGE SYSTEM CHALLENGES

Delivering cost-effective reliable data storage to users is a paramount mission that involves a variety of efforts. As in other competitive technological markets, the numerous engineering challenges of large-scale storage systems are divided and encapsulated in standardized layers, allowing vendors to offer highly specialized solutions for small parts of the general problem. At the device level, the main challenge is to tame a chosen physical media (e.g. Magnetic, Semiconductor, and Micro-mechanical) into a dense and reliable storage unit [4]. At the enterprise level, multiple devices of different kinds and characteristics are combined into a storage array that protects the data from failures of individual units. Error-correcting codes are a major ingredient in driving performance and reliability of both storage devices and storage arrays. Higher layers of storage systems handle a variety of non-trivial services such as virtualization, backups and security. The results of this thesis address immediate concerns of storage systems, both at the device level (codes for Multi-level Flash memories, improved decoding of Reed-Solomon codes) and at the enterprise level (efficient array codes for Clustered failures, highly regular array codes with optimal redundancy and updates). Therefore, it is hoped and believed that the fast-evolving and innovation-demanding data-storage technology will benefit from the proposed methods and ideas of Shannon's theorem. [5]

2.1 ERROR CORRECTING CODES TECHNIQUES

We have seen that when redundancy is included in code words, such as when a parity check bit is added to a bit string, we can detect transmission errors. We can do even better if we include more redundancy. We will not only be able to detect errors, but we will also be able to correct errors. More precisely, if sufficiently few errors have been made in the transmission of a codeword, we can determine which codeword was sent. This is illustrated by the following example [6].

To encode a message we can use the triple repetition code. We repeat a message three times. That is, if the message is $x_1x_2x_3$, we encode it as $x_1x_2x_3x_4x_5x_6x_7x_8x_9$ where $x_1 = x_4 = x_7$, $x_2 = x_5 = x_8$, and $x_3 = x_6 = x_9$. The valid code words are 000000000, 001001001, 010010010, 011011011, 100100100, 101101101, 110110110, and 111111111.

We decode a bit string, which may contain errors, using the simple majority rule. For example, to determine x_1 , we look at x_1 , x_4 , and x_7 . If two or three of these bits are 1, we conclude that $x_1 = 1$. Otherwise, if two or three of these bits are 0, we conclude that $x_1 = 0$. In general, we look at the three bits corresponding to each bit in the original message. We decide that a bit in the message is 1 if a majority of bits in the string received in positions corresponding to this bit are 1s and we decide this bit is a 0 otherwise. Using this procedure, we correctly decode the message as long as at most one error has been made in the bits corresponding to each bit of the original message.

For example, when a triple repetition code is used, if we receive 01111010, we conclude that the message sent was 011. (For instance, we decided that the first bit of the message was 0 since the first bit is 0, the fourth bit is 1, and the seventh bit is 0, leading us to conclude that the fourth bit is wrong.) [7]

To make the ideas introduced by the triple repetition code more precise we need to introduce some ideas about the distance between code words and the probability of errors. We will develop several important concepts before returning to error correcting codes.

III. ERROR CORRECTING CODE APPLICATION FOR AUDIO COMPACT DISC (CD)

In this section we will see the application of Reed-Solomon codes in an audio CD and how a CD can further improve on its error correcting capabilities through the use of a technique called interpolation [8].

3.1 CROSS-INTERLEVED REED-SOLOMON CODE

There are many error correcting codes used in practice today but the most commonly used method is cross-interleaved Reed-Solomon code or CIRC for short.

Recording in stereo requires 2 amplitude measurements taken 44; 100 times per second. One sample is taken from the left and the other from the right [9]. In the encoding process, each binary word of length 16 corresponds to an amplitude measurement which is represented by 2 field elements in GF (256). Each element is referred to as a byte. When recording in stereo, 4 bytes are produced at each "tick". A tick is 1 44; 100 of a second. The amplitude measurements from 6 consecutive ticks are formed together to make a message of length 24. After 4 parity bytes are added, this message is encoded to a codeword using C1, a (28, 24, 5) Reed-Solomon

code over GF (256). The codewords in C1 are then 4-framed delay interleaved. An additional 4 parity bytes are added and the codewords in C1 are then encoded using C2, a (32, 28, 5) RS code over GF (256). Afterwards, an additional byte is added to each codeword in C2 for control and display purposes and the length of each codeword is now. [10]

3.2 INTERPOLATION

There is another operation in the encoding process than can be used to further the chances of correcting errors within the C1 code. The bytes within code words in C1 are reorganized. Every codeword contains information from the left and the right, L1 to L6 and R1 to R6 respectively. There are two parity symbols added on as well in the encoding process, P1 and P2. These bytes are arranged in this order: L1L3L5R1R3R5P1P2L2L4L6R2R4R6. The bytes are arranged as such so that if several consecutive bytes are still flagged after decoding, they can be treated as unreliable information. If this is the case, any unreliable value, L_i , can be replaced by averaging the amplitudes of the adjacent values L_{i-1} and L_{i+1} if those samples are reliable. This technique is called interpolating values [11]. If this interpolating technique is used, then correcting up to 4 erasures is not used. This strategy allows up to two-erasure correction, and up to single error correction [12]. The maximum interpolatable burst length is around 12; 300 bits which corresponds to a track length of about 7.7mm. A deliberately scratched CD or even a CD with holes drilled into it can be played without loss of music. An audible click will be heard for an undetected erroneous sample or if even interpolation fails to correct an error than the clicking sound will be muted so the listener cannot hear it.

3.3 COMPARING THE ERROR CORRECTING CODES

The maximum correctable burst length is approximately 500 bytes which is about 2.4mm for CIRC, while it is 2200 bytes, approximately 4:6mm for RSPC. RSPC is capable of reducing a random input error rate of 2×10^{-2} to a data error rate of 10^{-15} , which is a factor of 10 better than in a CD [13]. When comparing the error rates after correction of the RSPC and the picket code, consider this experiment from. A dust sprayed disc was exposed to an office environment. That disc had a byte error rate of 4×10^{-3} . The errors included many long and short burst errors. After error correction, the BIS code error rate was found to be below 10^{-25} . The LDC was able to reduce the error rate to 1.5×10^{-18} . Compared to the RSPC of a DVD, the error rate was only reduced to 5.7×10^{-8} . Conclusively, the picket code is much more powerful than the RSPC.

3.3 ERROR-CORRECTING CODES FOR HIGH-SPEED COMPUTER MEMORIES

Error-correcting codes are widely used to enhance the reliability of computer memories. In these storage applications, the codes have unique features not seen in most communication applications. This is due to the following conditions required for error control in computer memories [14].

- 1) Encoding and decoding must be very fast to maintain high throughput rates. Therefore, parallel encoding and decoding must be implemented with combinatorial logic instead of the linear feedback shift-register logic usually used in other applications.
- 2) Unique types of errors, such as byte errors, are caused by the organization of semiconductor memory systems. As memory systems become large, the -bit-per-chip organization shown in Fig. 1 tends to be adopted. Then a chip failure causes the word read out to have errors in a -bit block, called a -bit byte or simply a byte.
- 3) Redundancy must be small, because the cost per bit of high-speed semiconductor memories is high compared with magnetic memories and other low-speed memories. The best-known class of codes for semiconductor memories is the SEC/DED (single-error-correcting/double-error detecting) codes. It can be viewed as a class of shortened versions of extended Hamming codes. These codes have minimum distance and are capable of correcting single errors and detecting double errors. Usually, however, they can detect more than just double errors. Since the additional error-detection capability depends on the structure of the parity-check matrix, many proposals for enhanced error detection properties have been presented. Among the many proposed schemes, the SEC/DED/SbED (single-error-correcting/double-error-detecting/single b-bit byte error-detecting) codes have been put into practical use.

These codes can correct a single bit error and detect any errors in single b-bit bytes well as any two bit errors. Such detection capability is useful for the memory organization of computer.

As an example let us consider the case of $b = 4$. An SEC/DED/S4ED code is constructed as follows. Let $p = 2^{m-1}$ and let $\{h_i \mid i = 1, \dots, p\}$ be the set of all binary m -dimensional column vectors of even weight if m is odd and of odd weight if m is even. For $i, j, = 1, 2, \dots, p$, the following $2m \times 4$ matrices H_{ij} are defined as

$$H_{ij} = [1 + h_i^{h_i} + h_j \ 1 + h_i^{h_i} + h_j \ 1 + h_{ih_i} + h_j \ 1 + h_{ih_j} + h_j] \dots \dots (1)$$

Where 1 denotes the m -dimensional column vector whose components are all ones. Letting $n = 2p(p - 1)$, we define a $2m \times n$ matrix H as

$$H = [H_{12} \ H_{13} \ \dots \ H_{1p} \ H_{23} \ \dots \ H_{2p} \ H_{34} \ \dots \ H_{p-2p} \ H_{p-1,p}] \dots \dots \dots (2)$$

Then it is easily seen that the code having parity-check matrix is an SEC/DED/S4ED code. [15] For example, in the case of $m = 4$, an SEC/DED/S4ED code of length 112 with eight check bits is constructed. Shortening this code by 40 bits, we have a SEC/DED/S4ED code. In order to have 64 information bits, SEC/DED codes require at least eight check bits. The construction method described above gives a code having the same redundancy as ordinary SEC/DED codes and, in addition, the capability of detecting any errors in a single 4-bit byte. As high-speed semiconductor memories become larger, more powerful error-correcting codes are needed. This has made SbEC/DbED (single -bit byte error correcting/double -bit byte error-detecting) codes practical. Such codes are constructed from RS codes or Bose-Chaudhuri Hocquengham (BCH) codes over $GF(2^b)$. If each symbol of $GF(2^b)$ is expressed as a b -bit byte, then RS codes or 2^b - ary BCH codes having minimum distance $d_{\min} = 4$ are SbEC/DbED codes. In applying these codes to semiconductor memories, they are usually substantially shortened to fit the number of information bits to the respective application. In such cases, we can sometimes obtain more efficient codes by slightly modifying RS codes. [16]

Table I shows the minimum number of parity-check bits for the best known SEC/DED codes, SEC/DED/ SbED codes, and SbEC/DbED codes when the number of information bits is 32, 64, and 128 for $b = 4$ and 8. [17]

TABLE 1
MINIMUM NUMBER OF PARITY-CHECK BITS FOR THE BEST
KNOWN ERROR-CONTROL CODES FOR HIGH-SPEED MEMORIES

Code	b	Number of information bits		
		32	64	128
SEC/DED	-	7	8	9
SEC/DED/SbED	4	7	8	9
	8	10	10	11
SbED/DbED	4	12	14	16
	8	24	24	24

IV. CONCLUSION

In this paper, we have presented a number of examples illustrating the way in which error-control coding, an outgrowth of Shannon's Information theory, has contributed to the design of reliable digital storage systems over the past 50 years. Indeed, coding has become an integral part of almost any system involving the transmission or storage of digital information. And as we look to the future and the "digital revolution," with such innovations as digital cellular telephony, digital television, and high-density digital storage, it seems assured that the use of error correcting coding will become even more pervasive. Finally, error correcting coding will continue to find use on increasingly difficult channels as demands for more reliable communications and data storage accelerate.

REFERENCES

- [1] R.J. McEliece. Comment on "a class of codes for asymmetric channels and a problem from the additive theory of numbers". IEEE Transactions on Information Theory, pp 19(1):137, 1973.
- [2] Imai, H and Nagasaka, Y. "On decoding methods for double encoding systems," Electron. Commun. Japan, vol. 65-A, pp. 36-44, Dec. 1982.
- [3] Shannon, C.E. "A mathematical theory of communication," Bell Syst. Tech. J., vol. 27, pp. 379-423, 1948.
- [4] R.K. Otter and A. Vardy. "Algebraic soft-decision decoding of Reed-Solomon Codes". IEEE Transactions on Information Theory, pp. 49(11):2809-2825, November 2003.
- [5] Chen, C.L. "Symbol error-correcting codes for computer memory systems," IEEE, Trans. Comp., vol. 41, pp. 252-256, 1992.
- [6] Jacobs, I.M. "Practical applications of coding," IEEE Trans. Inform. Theory, Vol. IT-20, pp. 305-310, 1974
- [7] Fire, P. "A class of multiple-error-correcting binary codes for no independent errors," Proc. 1997 IEICE General Conf., Mar. 1997, vol. TA-3-8, pp. 572-573.
- [8] E. Fujiwara, "Application of coding theory to computer systems," in Proc. 1997 IEICE General Conf., Mar. 1997, vol. TA-3-8, pp. 572-573.
- [9] R. Shahbender, "Laminated-ferrite memories-Review and evaluation," RCA Rev. Vol. 29, June, 1968 pp. 180-198. IEEE Transactions on Information Theory, 52(7):3186-3197, 2006.

- [10] Imai, H and Nagasaka, Y. (1990). "On decoding methods for double encoding Systems," Electron. Commun. Japan, vol. 65-A, pp. 36–44, Dec. 1982. Marcel Dekker, Inc., New York, 2000
- [11] Rao, T.R.N and E. Fujiwara, E. (1989) Error-Control Coding for Computer Systems.
- [12] K. IMMINK, The digital versatile disc (DVD): System requirements and channel coding, SMPTE J., 105 Prentice-Hall, pp. 483–489, (1996),
- [13] Hattori, H; Ohya, N. Sasano, M.. Sato, K and Shiota, N. (1993) "Improved general purpose Reed–Solomon erasure encoder/decoder chip," in Proc. 1993 Picture Coding Symp..
- [14] F.J MacWilliams and N.J.A Sloane. "The Theory of Error-Correcting Codes". North Holland, Amsterdam, The Netherlands, 19
- [15] T. R. N. Rao and E. Fujiwara, Error-Control Coding for Computer Systems. Englewood Cliffs, NJ: Prentice-Hall, pp.77, 1989.
- [16] H. Imai, Ed., Essentials of Error-Control Coding Techniques. San Diego, CA: Academic, 1990.
- [17] C. L. Chen, "Symbol error-correcting codes for computer memory systems," IEEE Trans. Comp., vol. 41, pp. 252–256, Feb. 1992.

Aru, Okereke Eze is a lecturer in the Department of Computer Engineering, Michael Okpara University of Agriculture, Umuahia, Abia State, Nigeria. His research interests include Computational Intelligence, Security system design, Expert systems, Design of Microcontroller and Microprocessor based system, digital systems design using microcontrollers and other computer related subjects.

Emole Chukwuemeka Enoch is a Postgraduate Student of Electrical/Electronic Engineering, Michael Okpara University of Agriculture, Umuahia, Abia State, Nigeria. Her research interests include Electronic and Digital Systems, Data Communication, etc.

Enyenihi Henry Johnson is a lecturer in the Head, Department of Electrical/ Electronics Engineering, Akwa Ibom State University in Uyo, Akwa Ibom State, Nigeria. His research interests include Computer Hardware design and maintenance, Electronic and Communication Systems, Electronic Security system designs etc.