SOFT-DECISION REDUNDANT RESIDUE NUMBER SYSTEM BASED ERROR CORRECTION CODING

T.H. Liew, L-L. Yang, L. Hanzo

Dept. of Electr. and Comp. Sc., Univ. of Southampton, SO17 1BJ, UK.
Tel: +44-703-593 125, Fax: +44-703-593 045
Email: th197r@ecs.soton.ac.uk,lh@ecs.soton.ac.uk
http://www-mobile.ecs.soton.ac.uk

ABSTRACT

Soft-decision based redundant residue number system (RRNS) assisted error control coding is proposed and its performance is evaluated. An RRNS(n,k) code is a maximum-minimum distance block code, exhibiting identical distance properties to Reed-Solomon (RS) codes. Hence their error correction capability is given by $t = (n-k)/2$. Different bit mapping methods are proposed, which result in systematic and non-systematic RRNS encoders. We show that the classic Chase algorithm can be invoked, in order to contrive soft-decision detection for RRNS codes and to exploit the soft channel outputs, which provide the relative reliability of each of the received binary digits. We found that soft decision based RRNS decoding is at least 1.5dB better as compared to hard decision assisted RRNS decoding.

1. INTRODUCTION

Since their introduction, redundant residue number systems (RRNS) have been considered to constitute a promising way of supporting fast arithmetic operations [1]–[9]. The arithmetic advantages accrue from the property that the RNS has the ability to add, subtract or multiply in parallel, regardless of the size of the numbers involved, without generating intermediate carry forward digits or internal delays [1, 2]. Furthermore, RRNSs have been studied extensively for the fault-tolerant protection of arithmetic operations in digital filters as well as in general purpose computers [2]–[8].

A coding theoretical approach to error control coding invoking the RRNS has been developed in [7, 8]. The concepts of Hamming weight, minimum distance, weight distribution, error detection capabilities and error correction capabilities were introduced. A computationally efficient procedure is described for example in [8], for correcting multiple errors.

In this paper, we propose an approach to exploiting the error control properties of the RRNS(n,k) code using powerful soft-decisions. Since a RRNS based code constitutes a non-binary block code, we propose two methods for mapping the binary source bits to the non-binary RRNS code symbols, which result in a so-called non-systematic and systematic RRNS code. At the receiver, we combine the RRNS decoder proposed in [8] with the classic Chase Algorithm [10] in order to improve the performance of the code using soft-decisions.

In Section 2 we give a brief introduction to RRNSs. Both systematic and non-systematic encoders are described in Section 3 and the decoder is detailed in Section 4. In Section 5 we propose a novel soft decoding method. Finally, our simulation results are presented in Section 6 and 7.

2. REDUNDANT RESIDUE NUMBER SYSTEM

A RRNS is defined in terms of an n-tuple of pairwise relative prime positive integers, m_1, m_2, ..., m_k, m_{k+1}, ..., m_n, referred to as moduli. The moduli m_1, m_2, ..., m_k are considered to be non-redundant moduli. The remaining $(n-k)$ moduli, m_{k+1}, m_{k+2}, ..., m_n, form the set of redundant moduli that allows error detection and correction in the RRNS. The product of the non-redundant moduli represents the so-called dynamic range, M_k, of the RRNS, which is given by:

$$M_k = \prod_{j=1}^{k} m_j \ . \tag{1}$$

The interval $[0, M_k - 1]$ is also often referred to as the *legitimate range*, while the interval $[M_k, M_n - 1]$ as the *illegitimate range*, where $M_n = \prod_{j=1}^{n} m_j$.

Any positive integer X, where $0 \leq X < M_k$, can be represented by an n-tuple residue sequence given by:

$$X \longleftrightarrow (x_1, x_2, ..., x_k, x_{k+1}, ..., x_n) \ , \tag{2}$$

where the quantity x_j is the lowest positive integer remainder of the division X by m_j and is designated as the residue of X mod (modulo) m_j or $|X|_{m_j}$. The positive integer x_j is also termed the j-th residue digit of X.

0-7803-5435-4/99/$10.00 © 1999 IEEE

Given the n-component residue vector, x_1, x_2, ..., x_n, we can reconstruct the integer X from the residues using a procedure known as the Chinese Remainder Theorem (CRT) [1,2], according to:

$$X = \left[\sum_{j=1}^{n} M_j |x_j L_j|_{m_j} \right] \bmod M_n \ , \qquad (3)$$

where $M_j = \frac{M_n}{m_j}$ and L_j is the multiplicative inverse of $M_j \bmod m_j$, $|L_j M_j|_{m_j} = 1$.

The so-called Mixed Radix Conversion (MRC) can also be used to replace the CRT, representing the integer X in the form of:

$$X = \sum_{i=1}^{n} a_i \prod_{j=1}^{i-1} m_j \ , \qquad (4)$$

where $0 \leq a_i < m_i$ and $\prod_{j=1}^{0} m_j = 1$. In the MRC algorithm, the digits a_1, a_2, ..., a_k are referred to as the mixed radix information digits, and a_{k+1}, ..., a_n will be termed as the mixed radix parity digits.

3. RRNS ENCODER

In the previous section we stated that an RRNS code is given by a set of residues with respect to a pre-defined set of moduli. Since the moduli and the residues can assume positive integers - representing by an arbitrary number of binary bits - the RRNS constitutes a non-binary code, based on transmitting the residues conveying a number of bits. In this section, we propose two different mapping methods from the binary source bits to the non-binary RRNS code, which result in a so-called non-systematic and a systematic RRNS code, respectively.

3.1. Non-systematic Encoder

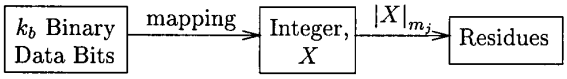

Figure 1: Non-systematic encoding procedures.

We summarised the non-systematic encoding process in Figure 1. The non-systematic encoder accepts k_b number of binary data bits each time, where the integer 2^{k_b} must be in the range $[0, M_k - 1]$ and M_k was defined by Equation 1 and hence has to obey:

$$\max_{k_b} \left\{ 2^{k_b} \right\} < M_k \ . \qquad (5)$$

The data bits are then mapped to an integer X, which has to be in the range of $[0, 2^{k_b} - 1]$. Note that the full range of the RRNS might not be used. Using the moduli in the RRNS, the residues x_j are simply obtained by taking the modulus, as shown in Figure 1.

3.2. Systematic Encoder

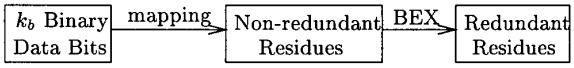

Figure 2: Systematic encoding procedures.

Figure 2 characterises the systematic encoding process. Unlike the non-systematic encoder, which maps all the data bits to be transmitted to an integer X, the systematic encoder divides the bit sequence to be encoded into shorter groups of bits, each of which is represented by a non-redundant residue x_j. Here we state that the mapping rule we proposed is not necessarily unique, but with the aid of the proposed soft-decision algorithm we will be able to resolve the associated ambiguity with a high probability. The reason for introducing this non-unique mapping is to ensure that the dynamic range offered by the RRNS code used is exploited in bit-mapping terms as efficiently as possible. This issue will be further clarified during our forthcoming discussions. Explicitly, we define the systematic mapping rule such that the number of binary data bits k_{b_j} assigned to each non-redundant residue x_j must be the smallest number, satisfying $2^{k_{b_j}} \geq m_j$, obeying:

$$\min_{k_{b_j}} \left\{ 2^{k_{b_j}} \right\} \geq m_j \ . \qquad (6)$$

As a consequence of the above allocation of data bits, there might exist a number of integers X, which cannot be uniquely represented by k_{b_j} data bits, since they are equal to or greater than the modulus, i.e. $X \geq m_j$. Hence, we define the mapping rule as:

$$x_j = \begin{cases} X & \text{if } X < m_j \\ 2^{k_{b_j}} - 1 - X & \text{if } X \geq m_j \end{cases} \ , \qquad (7)$$

where X is the integer represented by a group of k_{b_j} data bits. The mapping rule of Equation 7 implements bitwise complement computations, if $X \geq m_j$, in order to achieve maximum Hamming distance separation between the two bit patterns that are mapped to x_j according to Equation 7. At the receiver - since these integers exhibit maximum Hamming distance separation - we can calculate the Euclidean distance of both integers from the received integer, in order to determine the more likely transmitted integer. The total number of data bits that the systematic encoder encodes each time becomes $k_b = \sum_{j=1}^{k} k_{b_j}$.

Accordingly, as shown in Figure 2, the data bit sequences to be encoded are mapped to the non-redundant residues directly. Then the so-called Base Extension (BEX) algorithm can be invoked [1], in order to compute the redundant residues from the known non-redundant ones.

0-7803-5435-4/99/$10.00 © 1999 IEEE

4. RRNS DECODER

The minimum distance d_{min} is a fundamental parameter associated with any error control code. In [7,8], Krishna et. al. derived the necessary and sufficient conditions concerning the redundant moduli for an RRNS code to exhibit a minimum distance of d_{min}. The minimum distance of an RRNS code is d_{min}, if and only if the product of redundant moduli satisfies the following relation [7,8]:

$$\max\left\{\prod_{i=1}^{d_{min}} m_{j_i}\right\} > M_{n-k} \geq \max\left\{\prod_{i=1}^{d_{min}-1} m_{j_i}\right\} , \tag{8}$$

where $M_{n-k} = \prod_{j=k+1}^{n} m_j$ represents the 'redundant dynamic range' of the code and m_{j_i} is any of the n moduli of the RRNS code, for $1 \leq j_i \leq n$. Similarly to Reed-Solomon (RS) codes, the error correcting capability of an RRNS is also given by [7]:

$$t = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor , \tag{9}$$

where $\lfloor \bullet \rfloor$ means the largest integer not exceeding $\bullet$. From Equation 8, the smallest value of M_{n-k} for a minimum distance d_{min} is obtained by setting

$$M_{n-k} = \max\left\{\prod_{i=1}^{d_{min}-1} m_{j_i}\right\} . \tag{10}$$

Equation 10 shows that the left hand side inequality of Equation 8 is satisfied trivially. It also shows that an optimal RRNS, which is associated with the minimum necessary redundant dynamic range of M_{n-k} for achieving a minimum distance of d_{min} has the largest modulus of $d_{min}-1$ as the redundant modulus. Therefore, we can write that:

$$\begin{aligned} d_{min}-1 &= n-k \\ n &= k+d_{min}-1 . \end{aligned} \tag{11}$$

Using the standard coding theoretical terminology, we will refer to an RRNS that satisfies Equation 11 as a maximum distance separable RRNS (MDS-RRNS) code.

The RRNS decoder invoked in this paper was proposed in [8]. The multiple error correction procedures in [8] are extensions of those in [5,9]. In [5,9], the algorithms for locating a single residue digit error are based on the properties of modulus projection and Mixed Radix Conversion (MRC). However, the proposed RRNS decoder assumes that the output of the demodulator is binary. This implies that the algorithm is incapable of exploiting the soft outputs provided by the demodulator at the receiver. Here, we propose soft decoding of RRNS codes by combining the classic Chase algorithm [10] with the hard decision based RRNS decoder of [8].

5. SOFT DECODING OF RRNS CODES

We consider the transmission of block coded binary symbols $\{-1,+1\}$ using BPSK modulation over an AWGN channel. At the receiver, the demodulator provides the received signal values $\underline{y}$ for the RRNS decoder. A maximum-likelihood decoder is capable of finding the codeword that satisfies:

$$\min_{j} \text{ weight}(|\underline{y} - \underline{x}_j|^2) , \tag{12}$$

where $x_{ji} \in \{-1,+1\}$ are the transmitted binary coded symbols and the range of j is over all possible legitimate codewords. The decision given by Equation 12 is optimum, but the computational complexity increases exponentially with k and becomes prohibitive for block codes with $k > 6$. As a remedy, the Chase algorithm was proposed for near maximum-likelihood decoding of block codes [10]. The algorithm is sub-optimum, but it offers a significantly reduced complexity.

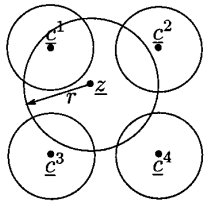

Figure 3: Stylised illustration of the Chase algorithm.

At the demodulator, the received bits $\underline{y}$ are sampled, yielding the sequence $\underline{z}$ and the associated confidence values $|\underline{y}|$ are fed to the Chase Algorithm. Accordingly, the sampled sequence $\underline{z}$ is intentionally perturbed by a set of test patterns TP, which is a binary sequence that contains binary 1s in the bit positions that are to be tentatively inverted. By modulo two adding this test pattern to the received sequence a new sequence $\underline{z}'$ is obtained, where:

$$\underline{z}' = \underline{z} \oplus TP . \tag{13}$$

Using the test patterns, the perturbed received sequence $\underline{z}'$ falls within the decoding sphere of a number of valid codewords, namely in that of for example $\underline{c}^1 \ldots \underline{c}^4$, as shown in Figure 3. In the figure r represents the maximum distance of the perturbed received sequence $\underline{z}'$ from the original sampled received sequence $\underline{z}$. If we increase r, the perturbed received sequence $\underline{z}'$ will fall within the decoding sphere of more valid codewords. In order to reduce the associated decoding complexity, only l bit positions associated with the least reliable confidence values $|\underline{y}|$ are considered for perturbation.

If the perturbed received sequence $\underline{z}'$ falls within the decoding sphere of a valid codeword, by hard decision RRNS decoding a new error pattern $\underline{e}'$ is obtained, which may be an all-zero or a non-zero tuple.

0-7803-5435-4/99/$10.00 © 1999 IEEE

The actual error pattern $\underline{e}$ associated with the received sequence $\underline{z}$ is given by

$$\underline{e} = \underline{e}' \oplus TP, \tag{14}$$

which may or may not be different from the original test pattern TP depending on whether or not the perturbed received sequence $\underline{z}'$ falls into the decoding sphere of a valid codeword. However, only those perturbed received sequences $\underline{z}'$ that fall into the decoding sphere of a valid codeword are considered. In this case, we are concerned with finding the error pattern $\underline{e}$ of minimum 'analogue weight', where the analogue weight of an error sequence $\underline{e}$ is defined as:

$$W(\underline{e}) = \sum_{i=1}^{n} e_i |y_i| \ . \tag{15}$$

The generated test pattern TP will be stored, if the associated analogue weight W is found to be lower, than the previously registered analogue weights. The above procedure will be repeated for the maximum number of test patterns, which is tolerable in complexity terms. Upon completing this loop, the memory is checked in order to determine, whether any error pattern has been stored, and if so, the corrected decoded sequence will be $\underline{z} \oplus \underline{e}$. Otherwise, the decoded sequence is the same as the received sequence $\underline{z}$.

6. SIMULATION RESULTS

As an example, we have quantified the performance of the RRNS(28, 24) code in our simulations. The moduli chosen were 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 217, 223, 227, 229, 233, 239, 241, 247, 251, 253, 255 and 256. Moduli 251, 253, 255 and 256 are the redundant moduli. Applying Equation 8, we can show that the minimum distance d_{min} is equal to 5. Therefore, we can calculate the error correction capability t using Equation 9, which is equal to 2. Besides, the RRNS(28, 24) code also satisfies Equation 11 and hence it is a maximum separable code.

Figure 4 shows the performance comparison between non-systematic and systematic RRNS encoders, which were described in Section 3. Due to their different mapping methods, the code rate for the systematic encoder is $R = 0.86$ as compared to $R = 0.81$ for the non-systematic encoder. The performance of the systematic encoder is about 1.0 dB better than the non-systematic encoder. The figure also shows the performance of the systematic Reed Solomon code RS(28,24) code over $GF(8)$ in comparison to the systematic RRNS-(28,24) code. It can be seen that the performance of the systematic RS(28,24) code and the systematic RRNS(28,24) code is similar.

Figure 5 shows the associated performance curves of systematic RRNS soft decoding for different number of test positions l. At a BER of 10^{-5}, there is

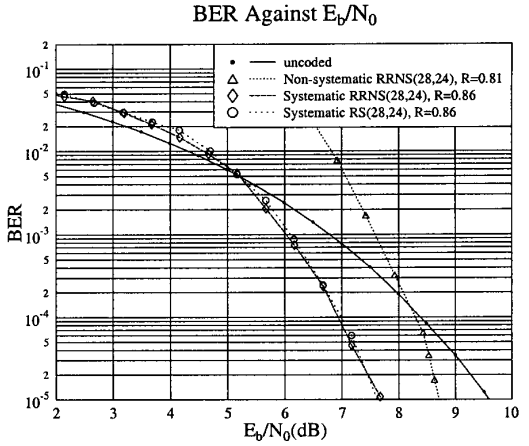

Figure 4: Performance comparison between non-systematic and systematic RRNS hard decoding using BPSK modulation over AWGN channels. The performance of RS(28,24) Reed-Solomon code over $GF(8)$ is also included.

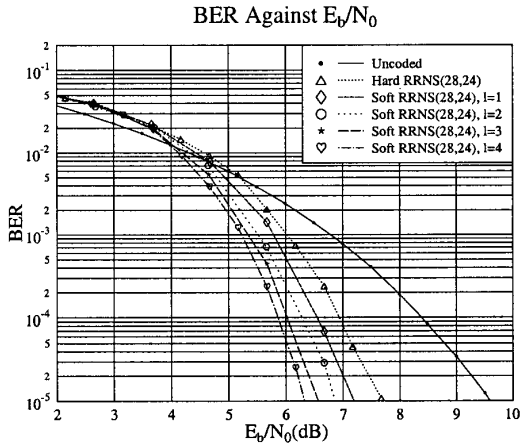

Figure 5: Performance of systematic RRNS soft decoding for different number of test positions l using BPSK modulation over AWGN channels. The performance of systematic RRNS hard decision decoding is also shown for comparison.

0-7803-5435-4/99/$10.00 © 1999 IEEE

a coding gain of 2.3 dB for $l = 1$. Upon increasing the number of test positions l, the larger the subset of tentatively decoded words, the more likely that it contains the transmitted codeword, and hence the better the coding gain. However, the improvement becomes smaller, as the number of test positions l increases. Furthermore, the complexity of the algorithm increases exponentially, since the number of test patterns is equal to 2^l. The performance of systematic RRNS hard decoding is also shown in Figure 5 for comparison. For $l = 4$, i.e. for 16 test patterns, the coding gain of RRNS soft decoding is about 3.2 dB at a BER= 10^{-5}.

7. PERFORMANCE OF RRNS CODES OVER RAYLEIGH CHANNELS

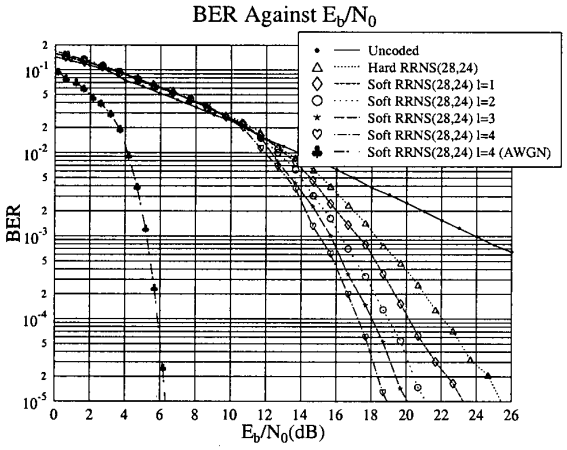

Figure 6: Performance of systematic RRNS soft decoding for different number of test positions l using BPSK modulation over uncorrelated Rayleigh fading channels assuming perfect fading inversion. The performance of systematic RRNS hard decoding is also shown for comparison.

The performance of the RRNS code investigated was also evaluated using BPSK modulation over the so-called perfectly interleaved or uncorrelated Rayleigh fading channel assuming perfect Rayleigh-fading envelope inversion. In Figure 6, we compared the soft decoding RRNS coding performance over both AWGN channels and Rayleigh fading channels under the investigated conditions. At BER= 10^{-5} the performance of RRNS(28,24) code with $l = 4$ is about 6 dB better, than that of hard decision decoding.

8. CONCLUSION

We have proposed a novel non-binary error control code, namely the RRNS code. Its performance was found to be similar to Reed Solomon codes. The Chase algorithm was then used to implement the soft decision decoding of RRNS codes. In our future work, we are researching turbo RRNS codes.

9. ACKNOWLEDGEMENTS

The financial support of the European Union under the auspices of the Pan-European FIRST project and that of Motorola ECID, Swindon UK is gratefully acknowledged. The authors also wish to thank the members of the FIRST consortium for helpful discussions and for their friendship.

10. REFERENCES

[1] Fred J. Taylor, "Residue Arithmetic: A Tutorial with Examples," *IEEE Computer Magazine*, pp. 50-62, May 1984.

[2] Nicholas S. Szabo and Richard I. Tanaka, "Residue Arithmetic and Its Applications to Computer Technology," *McGraw-Hill Book Company*, New York, 1967.

[3] R. W. Watson and C. W. Hastings, "Self-Checked Computation Using Residue Arithmetic," *Proceedings of the IEEE*, vol. 54, No 12, pp. 1920-1931, Dec 1966.

[4] David Mandelbaum, "Error Correction in Residue Arithmetic," *IEEE Transactions on Computers*, vol. C-21, No 6, pp. 1538-543, June 1972.

[5] Ferruccio Barsi and Piero Maestrini, "Error Correcting Properties of Redundant Residue Number Systems," *IEEE Transactions on Computers*, vol. C-22, No 3, pp. 307-315, March 1973.

[6] Stephen Sik-Sang Yau and Yu-Cheng Liu, "Error Correction in Redundant Residue Number Systems," *IEEE Transactions on Computers*, vol. C-22, No 1, pp. 5-11, January 1984.

[7] Hari Krishna, Kuo-Yu Lin and Jenn-Dong Sun, "A Coding Theory Approach to Error Control in Redundant Residue Number Systems - Part I: Theory and Single Error Correction," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 39, No 1, pp. 8-17, January 1992.

[8] Jenn-Dong Sun and Hari Krishna, "A Coding Theory Approach to Error Control in Redundant Residue Number Systems - Part II: Multiple Error Detection and Correction," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 39, No 1, pp. 18-34, January 1992.

[9] Mark H. Etzel and W. K. Jenkins "Redundant Residue Number Systems for Error Detection and Correction in Digital Filters," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-28, No 5, pp. 538-544, October 1980.

[10] David Chase, "A Class of Algorithms for Decoding Block Codes With Channel Measurement Information," *IEEE Transactions of Information Theory*, vol. 18, No 1, pp. 170-182, January 1972.