BIT LEVEL SOFT DECISION DECODING OF REED–SOLOMON CODES

1.1.3

Alexander Vardy and *Yair Be'ery*

Department of Electronic Communications, Control and Computer Systems,
Tel–Aviv University.

ABSTRACT

In this paper we present a Reed–Solomon decoder that makes use of bit soft decision information. A Reed–Solomon generator matrix which possesses a certain inherent structure in GF(2) is derived. Using this structure representation of the code as a union of cosets, each coset being an interleaver of several binary BCH codes, is obtained. Such partition into cosets provides a clue for efficient bit level soft decision decoding. The proposed decoding algorithms are in many cases orders of magnitude more efficient than conventional techniques.

I. INTRODUCTION

The practical importance of Reed–Solomon codes is well established (see for example [1],[2]). The application of Reed–Solomon codes spans from compact disk systems to deep–space standard[3],[4]. Hard decision decoders for RS codes are readily availiable using algebraic decoding algorithms. Such decoders have been implemented and operate at rates above 120 Mb/s. Soft decision decoding of RS codes is, however, an entirely different matter. Even though the decoder can be supplied with reliable soft decision data relatively easily[1], the high complexity of optimal soft decoding makes full utilization of such data prohibitive. In fact, the availiable soft decoding algorithms, such as Forney's generalized minimum distance decoding [5] and others [6],[7], make use of soft decision information only on the byte level. Namely, the confidence values of the received bits are processed in one way or another [7] to generate the average confidence value of the symbol, which is then transferred to the decoder. Thus, the bit level soft decision information is lost. In this context Berlekamp et al [1] state that "the major drawback with RS codes is that the present generation of decoders do not make full use of bit–based soft decision information". In this paper we present Reed–Solomon decoders that make use of bit soft decision information. The proposed decoding algorithms are in many cases several orders of magnitude more efficient than the existing techniques (say, Viterbi decoding based on the conventional Wolf's trellis [8]). The reduced complexity of our algorithms is due to a certain symmetric structure of the RS generator matrix over GF(2) which is derived in Section II. The bit level soft decision decoders utilizing this structure are presented in Section III.

II. STRUCTURE OF THE GENERATOR MATRIX

Let $\mathcal{R}(N,K)$ be the Reed–Solomon code over GF(2^m) of length $N = 2^m - 1$ and dimension K. We assume that $\mathcal{R}$ is used on a binary channel. Hence the encoder must employ some fixed linear mapping $\phi: \mathrm{GF}(2^m)^N \rightarrow \mathrm{GF}(2)^{mN}$ to convert a sequence of N elements of GF(2^m) into a string of mN binary digits. Namely, a codeword $\underline{c} = (c_0, c_1, \ldots c_{N-1}) \in \mathcal{R}$, $c_i \in \mathrm{GF}(2^m)$ is transmitted as $\phi(\underline{c}) = (c_0^1, c_0^2, \ldots c_0^m, c_1^1, c_1^2, \ldots c_1^m, \ldots c_{N-1}^1, c_{N-1}^2, \ldots c_{N-1}^m)$, where $c_i^j \in \mathrm{GF}(2)$

Now let α be a primitive element of GF(2^m) and let $\alpha^s, \alpha^{s+1} \ldots \alpha^{s+N-K-1}$ be the set of zeroes of $\mathcal{R}$. Denote by $\mathcal{B}$ the binary BCH code of length $n = 2^m - 1$ with zeroes at $\alpha^s, \alpha^{s+1} \ldots \alpha^{s+N-K-1}$ and their cyclotomic conjugates over GF(2). Let $\gamma_1, \gamma_2, \ldots \gamma_m$ be the basis of GF(2^m) over GF(2) employed for the linear mapping ϕ. We define the codes $\mathcal{B}_1, \mathcal{B}_2, \ldots \mathcal{B}_m$ as

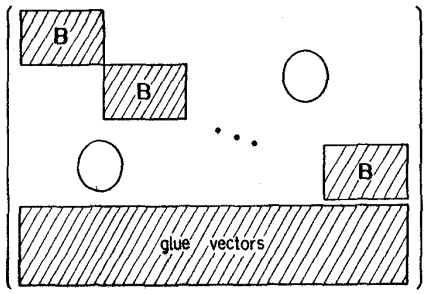
$$\mathcal{B}_j = \{ (\gamma_j b_0, \gamma_j b_1, \ldots \gamma_j b_{N-1}) \mid \underline{b} = (b_0, b_1, \ldots b_{N-1}) \in \mathcal{B} \} \quad (1)$$

where $b_i \in \mathrm{GF}(2)$ and the product $\gamma_j b_i$ is in GF(2^m). It is well known [9] that $\mathcal{B}$ is a subfield subcode of $\mathcal{R}$ and, hence, the m codes defined in (1) are also subcodes of $\mathcal{R}$. Therefore if $\{\underline{v}_1^j, \underline{v}_2^j, \ldots \underline{v}_k^j\}$ is a set of k generators for $\mathcal{B}_j$ we may use the set

$$\bigcup_{j=1}^{m} \{ \phi(\underline{v}_1^j), \phi(\underline{v}_2^j), \ldots \phi(\underline{v}_k^j) \}$$

as the first mk rows of a binary generator matrix for $\mathcal{R}$ By rearranging the columns the following structure is obtained

$$G = \quad (2)$$



where **B** is a $k \times n$ generator matrix of $\mathcal{B}$ This proves our basic theorem.

THEOREM 1. *Let* $\mathbf{B} = [b_{ij}]$ *be a generator matrix of the binary BCH code of length* $n = 2^m - 1$, *dimension* $k < K$ *and designed distance* $d \geq N-K+1$. *Then there exists a binary genarator matrix of* $\mathcal{R}$, $\mathbf{G} = [g_{ij}]$, $0 \leq i \leq mN-1$, $0 \leq j \leq mK-1$, *such that for* $0 \leq j \leq mk-1$

$$1)\ g_{ij} = 0 \qquad \text{if } \lfloor j/k \rfloor \neq \lfloor i/n \rfloor \quad (3a)$$

$$2)\ g_{ij} = b_{\bar{i}\bar{j}} \qquad \text{if } \lfloor j/k \rfloor = \lfloor i/n \rfloor \quad (3b)$$

where $\bar{i} \equiv i \pmod n$ *and* $\bar{j} \equiv j \pmod k$.

Using Theorem 1 $\mathcal{R}$ may be written as a union of cosets

$$\mathcal{R} = \bigcup_{\ell=0}^{2^{\Delta}-1} \mathbf{C}_\ell \quad (4)$$

such that $\Delta = m(K-k)$ and $\mathbf{C}_\ell = \{\underline{r}^\ell + \underline{c} \mid \underline{c} \in \mathbf{C}\}$, where

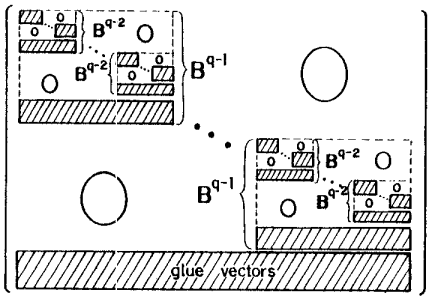
$$\mathbf{C} = \mathcal{B}_1 \oplus \mathcal{B}_2 \oplus \ldots \oplus \mathcal{B}_m$$

is a direct sum of the m codes defined in (1) and the vectors $\underline{r}^\ell$, $\ell = 0,1,\ldots 2^\Delta - 1$ are coset representatives for $\mathbf{C}$ in $\mathcal{R}$.[1] Note that the conditions (3a) and (3b) arrange the columns of $\mathbf{G}$ in such a way that the m bits of each channel symbol are distributed evenly among the m BCH codes. This property of the generator matrix gives a new insight into the inherent burst error correction capability of Reed–Solomon codes, since each given coset of $\mathcal{R}$ may be viewed as an *interleaver*. Assume for simplicity that some message vector $\underline{s}$ is mapped into the coset corresponding to $\underline{r} = \mathbf{0}$. Then for this message $\underline{s}$ employing $\mathcal{R}(N,K)$ on a binary channel is equivalent to interleaving m t–error correcting binary BCH codes of length $2^m - 1$, where $2t+1 = N-K+1$. A coset corresponding to $\underline{r} \neq \mathbf{0}$ is also an interleaver which interleaves m translates of the binary $(n,k,2t+1)$ BCH code. Any channel burst of less than $m(t-1)+2$ bits will be distributed evenly among the m codes, at most t bits to a code, and therefore will be successfully corrected by the hard–decision decoder. This conception of the RS code as a union of cosets, each coset being an interleaver, will be extensively used in the next section.

It is noteworthy that Theorem 1 may be straightforwardly generalized so as to apply to any linear code that contains a subfield subcode. If m is composite we can employ such generalization of Theorem 1 to obtain an additional structure for the binary generator matrix of $\mathcal{R}$. Let $m = p_1 p_2 \ldots p_q$, where $q \geq 2$ and $p_1, p_2, \ldots p_q$ are not necessarily distinct primes. Then GF(2) $\subset$ GF(2^{p_1}) $\subset$ ... $\subset$ GF(2^m). Let $\mathcal{B}^{(j)}$ be the BCH code of length $2^m - 1$ and designed distannce $N-K+1$ over GF($2^{p_1 p_2 \ldots p_j}$). Evidently $\mathcal{B}^{(j_1)}$ is a subfield subcode of $\mathcal{B}^{(j_2)}$, provided that $j_2 > j_1$. Hence, instead of going directly from $\mathcal{R}$ to $\mathcal{B}$ we may use the structure of the whole chain of nested BCH codes over nested fields

$$\mathcal{B} \subset \mathcal{B}^{(1)} \subset \ldots \subset \mathcal{B}^{(q)} = \mathcal{R} \qquad (5)$$

Appllying Theorem 1 successively q times we obtain the following Reed–Solomon generator matrix

$\mathbf{G} =$ 

(6)

where $\mathbf{B}^j$ is the generator matrix of $\mathcal{B}^{(j)}$.

[1] The terminology used here as well as the term glue vectors employed in (2) is from Conway and Sloane [10]. The subcode of $\mathcal{R}$ generated by the Δ glue vectors (the *glue code* in terminology of [10]) consists of precisely the coset representatives for $\mathbf{C}$ in $\mathcal{R}$.

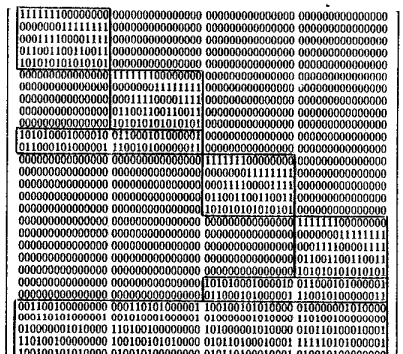
Example. Consider the (15,9,7) RS code over GF(2^4) generated by $g(x) = \prod_{i=1}^{6}(x-\alpha^i)$, where α is a zero of $\pi_1(x) = x^4 + x + 1$. The factorization of m is given by $4 = 2 \cdot 2$. Hence $q = 2$ and $p_1 = p_2 = 2$; $\mathcal{B}^{(q)} = \mathcal{B}^{(2)} = \mathcal{R}(15,9)$; $\mathcal{B}^{(1)}$ is the (15,6,7) BCH code over GF(2^2) $\subset$ GF(2^4) generated by $g_1(x) = g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{12})$; $\mathcal{B}$ is the binary (15,5,7) BCH code generated by $g_0(x) = g_1(x)(x-\alpha^{10})$. A generator matrix for $\mathcal{R}$ (15,9) over GF(2^4) is given by

$$\mathbf{G} = \begin{bmatrix} x^4 g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{10})(x-\alpha^{12}) \\ x^3 g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{10})(x-\alpha^{12}) \\ x^2 g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{10})(x-\alpha^{12}) \\ x g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{10})(x-\alpha^{12}) \\ g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{10})(x-\alpha^{12}) \\ g(x)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{12}) \\ g(x)(x-\alpha^8)(x-\alpha^9) \\ g(x)(x-\alpha^8) \\ g(x) \end{bmatrix}$$

Writing the polynomials $g_0(x)$, $g_1(x)$ and $g(x)$ explicitely we have

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \alpha^5 & 1 & 0 & 0 & \alpha^5 & 1 & \alpha^5 & \alpha^{10} & \alpha^{10} & 1 & 0 & 0 & 0 & 0 & 0 \\ \alpha^8 & \alpha^5 & \alpha^8 & \alpha^{11} & \alpha^6 & \alpha & \alpha^5 & \alpha^3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha^{14} & \alpha^3 & \alpha^4 & \alpha^4 & \alpha^3 & 1 & \alpha & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha^6 & \alpha^9 & \alpha^6 & \alpha^4 & \alpha^{14} & \alpha^{10} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad (7)$$

Note that the sixth row of (7) belongs to GF(2^2) as expected. We arbitrarily choose $\{1,\alpha\}$ as a basis for GF(2^4) over GF(2^2) and $\{1,\beta\}$ as a basis for GF(2^2) over GF(2), where β is a zero of $\pi_2(x) = x^2 + x + 1$. Thus, the basis for GF(2^4) over GF(2) becomes $\{1,\beta,\alpha,\alpha\beta\} = \{1,\alpha^5,\alpha,\alpha^6\}$. Replacing each entry in (7) by its corresponding matrix representation with respect to the above basis and rearranging the columns we obtain the following binary generator matrix for $\mathcal{R}$(15,9)



III. SOFT DECISION DECODING

Suppose that using the linear mapping ϕ a codeword of $\mathcal{R}$ is transmitted through a binary channel. We assume throughout a continious–output, say, additive white gaussian noise (AWGN) channel, characterized by transition probability densities $f_j(v) = f(v/j)$, $j \in GF(2)$, $v \in \mathbb{R}$, where $\mathbb{R}$ is the real line. In case of a discrete channel with output alphabet $\mathbb{F}$, $f_j(v)$ should be replaced by transition probabilities $p_j(v) = p(v/j)$, $j \in GF(2)$, $v \in \mathbb{F}$. Now let the word $\underline{v} = (\underline{v}_0, \underline{v}_1, \ \dots \ \underline{v}_{N-1}) = (v_0^1, v_0^2, \dots v_0^m, v_1^1, v_1^2, \dots v_1^m, \ \dots \ v_{N-1}^1, v_{N-1}^2, \dots v_{N-1}^m)$ be observed at the output. Maximum likelihood decoding consists of finding a codeword $\underline{c} \in \mathcal{R}$ that maximizes $P(\underline{v}/\underline{c})$, that is maximizes the *a posteriori* probability $P(\underline{c}/\underline{v})$, provided that $P(\underline{c})$ is the same for all $\underline{c} \in \mathcal{R}$. On a random noise memoryless channel one may as well search the maximum of

$$M(\underline{c}) = \sum_{i=0}^{N-1} \sum_{j=1}^{m} \log f(v_i^j / c_i^j) .$$

Using the partition into cosets (4) and interchanging the order of summation we may perform the maximization as follows

$$\max_{\underline{c} \in \mathcal{R}} M(\underline{c}) = \max_{\mathbb{C}_\ell} \max_{\underline{c} \in \mathbb{C}_\ell} \sum_{j=1}^{m} \sum_{i=0}^{N-1} \log f(v_i^j / c_i^j) .$$

It follows from the structure of the generator matrix obtained in Theorem 1 that in a given coset the choise of $(c_0^j, c_1^j, \dots c_{N-1}^j)$ may be made independently for each $j = 1,2,\dots m$. Hence we may interchange summation over j with maximization within a coset, i.e.

$$\max_{\underline{c} \in \mathcal{R}} M(\underline{c}) = \max_{\mathbb{C}_\ell} \sum_{j=1}^{m} \left[\max_{\underline{c} \in \mathbb{C}_\ell} \sum_{i=0}^{N-1} \log f(v_i^j / c_i^j) \right] .$$

Yet, the maximization within the square brackets is just the soft decoding of the inner BCH code $\mathcal{B}$. This implies the following decoding algorithm.

ALGORITHM 1

For each coset $\mathbb{C}_\ell$, $\ell = 0,1,\dots 2^{\Delta}-1$, *with coset representative* $\underline{r} = (r_0^1, r_0^2, \dots r_0^m, r_1^1, r_1^2, \dots r_1^m, \ \dots \ r_{N-1}^1, r_{N-1}^2, \dots r_{N-1}^m)$

1. *Find the* m *codewords* $\hat{\underline{b}}_1, \hat{\underline{b}}_2, \dots \hat{\underline{b}}_m$, *where* $\hat{\underline{b}}_j = (\hat{b}_0^j, \hat{b}_1^j, \dots \hat{b}_{N-1}^j) \in \mathcal{B}$ *by maximizing*

$$M_j(\underline{b}) = \sum_{i=0}^{N-1} \log f(v_i^j / b_i + r_i^j) \qquad (8)$$

with respect to all $\underline{b} \in \mathcal{B}$ *for* $j = 1,2,\dots m$.

2. *Evaluate*

$$M(\underline{c}) = \sum_{j=0}^{m} M_j(\hat{\underline{b}}_j) = \sum_{j=1}^{m} \sum_{i=0}^{N-1} \log f(v_i^j / c_i^j) \qquad (9)$$

where $c_i^j = \hat{b}_i^j + r_i^j$.

Decode to the codeword $\hat{\underline{c}} \in \mathcal{R}$ *that maximizes* (9) ■

For decoding the inner BCH code transforms [11],[12], trellises [8],[13] or other efficient methods [14] are availiable. Let Ω denote the computational complexity of either of these methods. Then the number of real addition equivalent operations required by the above decoding algorithm is given by

$$N_1 = [m\Omega + m] \cdot 2^{\Delta} = m(\Omega+1) \cdot 2^{m(K-k)} \qquad (10)$$

Evidently [11], Ω is upper bounded by $k \cdot 2^k$. However, in most cases Ω is much less than that due to a precomputation stage which is common to all the cosets. For instance for the (7,4,3) binary Hamming code $\Omega = 14$ without precomputation, and $\Omega = 3$ if a precomputation stage of 66 real operations is employed (for details see [14]). Comparing (10) with the complexity of the Viterbi algorithm, based on the conventional Wolf's trellis [8] for high–rate codes over GF(2), given by [11]

$$W_1 = [3m(2K - 2^m + 1) + 5] \cdot 2^{m(2^m - K - 1)}$$

we conclude that *Algorithm 1* implies an exponential computational gain for half–rate Reed–Solomon codes and also for many RS codes of higher rate. Obviously, the computational gain would be even greater for extended and doubly extended RS codes.

We may further reduce the complexity of decoding $\mathcal{R}$ by means of a recursive algorithm based on the "recursive" structure of the generator matrix derived in (6). Let k_j, $j = 1,2,\dots q$ denote the dimension of $\mathcal{B}^{(j)}$ (to keep the notation rigorous we also define $\mathcal{B}^{(0)} = \mathcal{B}$ and $k_0 = k$). The main idea of the recursive algorithm is representing each $\mathcal{B}^{(j)} \in \mathcal{R}$ as a union of cosets in a way analogous to (4)

$$\mathcal{B}^{(j)} = \bigcup_{\ell=0}^{2^{\Delta_j}-1} \mathbb{C}_\ell^j$$

such that $\Delta_j = \sum_{i=1}^{j} p_i(k_j - k_{j-1})$ and $\mathbb{C}_\ell^j = \{ \underline{r}^\ell + \underline{c} \mid \underline{c} \in \mathbb{C}^j \}$ where

$$\mathbb{C}^j = \mathcal{B}_1^{(j-1)} \oplus \mathcal{B}_2^{(j-1)} \oplus \dots \oplus \mathcal{B}_{p_j}^{(j-1)}$$

and $\underline{r}^\ell$, $\ell = 0,1,\dots 2^{\Delta_j}-1$ are coset representatives for $\mathbb{C}^j$ in $\mathcal{B}^{(j)}$. Given this recursive partition into cosets we may apply *Algorithm 1* recursively with $\mathcal{B}$ and $\mathcal{R}$ replaced by, respectively, $\mathcal{B}^{(j-1)}$ and $\mathcal{B}^{(j)}$ at each stage of recursion. The complexity of such recursive decoding is upper bounded by

$$N_2 = m(\Omega + q) \cdot 2^{\sum_{j=1}^{q} \Delta_j}$$

Obviously, $\sum_{j=1}^{q} \Delta_j < \Delta$. Thus for instance for the (15,9) RS code over $GF(2^4)$ we have $\sum_{j=1}^{2} \Delta_j = p_1(k_1 - k) + p_1 p_2(K - k_1) = 14$ and $\Delta = m(K-k) = 16$ (see example in Section II).

It should be pointed out that the proposed decoders maximize the sum of bit and not symbol likelihoods and, therefore, do not necessarily provide for the inherent burst error correction capability of Reed–Solomon codes. This is a direct consequense of our initial assumption of a binary *memoryless* channel. However, with only a slight

modification *Algorithm 1* becomes suitable for a "bursty" channel as well. Assuming as in [5],[15], independent noise on each transmitted symbol we may characterize a general binary channel with memory by the 2^m transition probability densities

$$f(v/\xi) = f(v^1,v^2,\ldots v^m/\xi^1,\xi^2,\ldots\xi^m) \qquad (11)$$

where $\xi \in GF(2^m)$ and $(\xi^1,\xi^2,\ldots\xi^m)$ is a radix–2 expansion of ξ. A binary AWGN channel with bursts is a special case of (11). Now recall that each coset of $\mathcal{R}$ is an interleaver. It is well known [16],[17] that interleaving converts a channel with memory (especially a channel with bursts) to one that can be treated as memoryless. Hence, *within a given coset* of $\mathcal{R}$ the maximization may be still performed separately for each of the m interleaved codes and the first step of *Algorithm 1* remains unchanged. At step 2, however, taking into account channel memory, we evaluate

$$M(\underline{c}) = \sum_{i=0}^{N-1} \log f(v_i/c_i) = \sum_{i=0}^{N-1} \log f(v_i^1,v_i^2..v_i^m/\hat{b}_i^1+r_i^1,\hat{b}_i^2+r_i^2,\ldots\hat{b}_i^m+r_i^m) \qquad (12)$$

and decode to the codeword $\hat{\underline{c}} \in \mathcal{R}$ that maximizes (12). Evidently the above modification almost does not affect the decoding complexity which is now given by

$$N_3 = [m\Omega+2^m-1]\cdot 2^{m(K-k)}$$

as compared to the complexity of Viterbi decoding, based on Wolf's trellis over $GF(2^m)$

$$W_2 = (3\cdot 2^{m-1})\frac{2^{m(2^m-K-1)}-1}{2^m-1} + (2K-2^m+1)(2^{m+1}-1)\cdot 2^{m(2^m-K-1)} - 2^m$$

Examples. For the (7,5) RS code over $GF(2^3)$ we have $K = 5$, $k = 4$ and $\Omega = 3$ with a precomputation stage of 66 real operations. Hence $N_3 = 128$ and the total complexity of soft decision decoding is 194 real addition equivalent operations per codeword or about 13 real operations per information bit. Decoding the same code with Viterbi algorithm based on Wolf's trellis we need $W_2 = 3079$ real operations per codeword, whereas straight forward maximization requires about 230,000 operations. For the extended (8,5) RS code we have $N_3 = 136$ and the total complexity of soft decoding is 208 real addition equivalent operations per codeword or 14 operations per information bit as compared to $W_2 = 17{,}031$ real operations per codeword or 1135 operations per information bit. For the (15,13), (15,11) and (15,9) RS codes over $GF(2^4)$ we need approximately 90, 800 and 2000 real operations per information bit, respectively. The same numbers using conventional trellis decoding are, respectively, 1700, 328,000 and 15,000,000 operations per information bit.

As illustrated by the foregoing examples the proposed algorithms are in many cases several orders of magnitude more efficient than any of the existing optimal techniques. Moreover, the structure of the RS generator matrix that was derived in Section II may serve a basis for sub–optimal bit level soft decoding algorithms practically applicable to long RS codes.

ACKNOWLEDGEMENT

The authors are indebted to Jakov Snyders for many constructive discussions. Alexander Vardy wishes to thank Hagit Itzkowich for her invaluable help.

REFERENCES

[1] E.R. Berlekamp, R.E. Peile, and S.P. Pope, "The application of error control to communications," *IEEE Communications Magazine*, Vol.25, No.4, pp. 44–57, 1987.

[2] W.W. Wu, D. Haccoun, R.E. Peile and Y. Hirata, "Coding for satellite communication," *IEEE J. Sel. Areas Comm.*, Vol.5, No.4, pp. 724–785, 1987.

[3] Consultative Committee for Space Data Systems, *Recommendations for Space Data System Standards: Telemetry Channel Coding " Blue Book "*, 1984.

[4] E.R. Berlekamp, J. Shifman and W. Toms, "An application of Reed–Solomon codes to a satellite TDMA system," MILCOM86, Monterey, CA.

[5] G.D. Forney, Jr., "Generalized minimum distance decoding," *IEEE Trans. Inform. Theory*, Vol.12, pp. 125–131, 1966.

[6] L.R. Welch and E.R. Berlekamp, *Error–Correction for Algebraic Block Codes*, U.S. Patent Application No.536951, 1983.

[7] N. Doi, H. Imai, M. Izumita and S. Mita, "Soft desicion decoding for Reed–Solomon codes," *Proc.* GLOBECOM87, pp. 2090–2094.

[8] J.K. Wolf, "Efficient maximum likelihood decoding of linear block codes," *IEEE Trans. Inform. Theory* Vol.24, pp. 76–80, 1978.

[9] F.J. McWilliams and N.J.A. Sloane, *The Theory of Error Correcting Codes*, North–Holland, Amsterdam 1977.

[10] J.H. Conway and N.J.A. Sloane, "Soft decoding techniques for codes and lattices, including the Golay code and the Leech lattice," *IEEE Trans. Inform. Theory*, Vol.32, No.1, pp. 41–50, 1986.

[11] Y. Be'ery and J. Snyders, "Optimal soft decision block decoders based on Fast Hadamard Transform," *IEEE Trans. Inform. Theory*, Vol.32, pp. 355–364, 1986.

[12] Y. Be'ery and J. Snyders, "A recursive Hadamard Transform optimal soft decision decoding algorithm," *SIAM J. Algebraic and Discrete Methods*, Vol.8, No.4, pp. 778–789, 1987.

[13] G.D. Forney, Jr., "Coset codes II : Binary lattices and related codes," *IEEE Trans. Inform. Theory*, Vol.34, No.5, 1988.

[14] J. Snyders and Y. Be'ery, "Maximum likelihood soft decoding of binary block codes and decoders for the Golay codes" *IEEE Trans. Inform. Theory*, to appear.

[15] M. Rice, D.J. Tait and P.G. Farell, "A soft decision Reed–Solonmon decoder," presented at the *IEEE Int. Symp. Inform. Theory*, Kobe, Japan, 1988.

[16] A.J. Viterbi and J.K. Omura, *Principles of Digital Communication and Coding*, McGraw Hill, New York 1977.

[17] J. Wolfowitz, *Coding Theorems of Information Theory*, Springer–Verlag, Berlin 1978.