

# Efficient Bounded-Distance Decoding of the Hexacode and Associated Decoders for the Leech Lattice and the Golay Code

Ofer Amrani and Yair Be'ery

**Abstract**—Two soft-decision decoding algorithms for the (6, 3, 4) quaternary code *hexacode* are presented. Both algorithms realize half the minimum Euclidean distance of the code. The proposed algorithms are most practical. In using them, bounded-distance decoding of the Golay code and the Leech lattice are performed with at most 187 and 519 real-number operations respectively. Compare this to 651, respectively 3595, operations required by the best known maximum likelihood decoders of [5], [6], and 431, respectively 1007, operations required by the bounded-distance decoders of [7]. We present some simulation results for the proposed Leech lattice decoders revealing near-optimal performance. A comparison to known trellis codes is also provided.

## I. INTRODUCTION

THE LEECH LATTICE, being the densest sphere packing lattice in 24 dimensions, is most interesting for two applications: 1) coded modulation techniques for band-limited additive white Gaussian noise (AWGN) channels, where a finite number of lattice points provide a signal set for data transmission [1], [2], and 2) vector quantization, where the lattice points provide a set of possible quantized vectors [3], [4]. For both applications, an algorithm is required to find the closest lattice point to an arbitrary given point in  $R^{24}$ .

The most efficient algorithms known for maximum likelihood as well as bounded-distance decoding of the Leech lattice and the Golay code [5]–[7] are based on their close interrelationship with the *hexacode*,  $H_6$ . The main idea behind these algorithms is to project the codewords of the aforementioned codes onto the 64 hexacodewords, a method which seems to work best. The algorithms in [5]–[7] involve maximum likelihood hexadecoding (i.e., decoding of the hexacode), which requires 179 real-number operations. By contrast, we propose that a slightly sub-optimal yet more efficient bounded-distance decoding of the hexacode be utilized [8], [14].

In Section II, we present two efficient bounded-distance decoding algorithms for the hexacode. Indeed, by substituting these algorithms into the algorithms of [7], as described in Section III, the computational complexities required for decoding the Leech lattice and the Golay code are considerably reduced, while the coding gain loss is negligible with Algorithm 1, and is only about 0.1dB with Algorithm 2. Simulation results and a comparison to known trellis codes, of similar coding-gain

and complexity to that of the Leech lattice, are provided in the last section of this paper.

## II. DECODING THE HEXACODE

### A. Decoding Algorithm 1

The generator matrix of  $H_6$  is given in (1), with  $\{0, 1, \omega, \bar{\omega}\}$  being the symbols of  $GF(4)$

$$\begin{bmatrix} 1 & 0 & 0 & 1 & \bar{\omega} & \omega \\ 0 & 1 & 0 & 1 & \omega & \bar{\omega} \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (1)$$

We assume that a sequence of six symbols corresponding to a hexacodeword  $\mathbf{x} = (x_1, \dots, x_6)$  is transmitted via an AWGN channel, and denote by  $\{r(n)\}_{n=1}^6$  the received version of that codeword. The first decoding algorithm may be described as follows.

**Precomputation:** As the metric of each  $GF(4)$  symbol, in any coordinate  $n$ , take the square Euclidean distance (SED) between the received value  $r(n)$  and the Euclidean representation of that symbol. The mapping of  $GF(4)$  onto the Euclidean space can take any form, however, let us assume an orthogonal mapping since this results with the highest decoding complexity.

**Computation:** Choose the symbol with the minimum metric in each coordinate, denoted  $\hat{x}_n$ , and construct a  $GF(4)^6$  vector  $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \hat{x}_3, \hat{x}_4, \hat{x}_5, \hat{x}_6)$ . If  $\hat{\mathbf{x}} \in H_6$ , select it as the output of the decoder. Otherwise, proceed as follows. Let  $\hat{\mathbf{y}}_1 = (\hat{x}_1, \hat{x}_2, \hat{x}_3)$  and  $\hat{\mathbf{y}}_2 = (\hat{x}_4, \hat{x}_5, \hat{x}_6)$ , be the two "blocks" of symbols, such that  $\hat{\mathbf{x}} = (\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2)$ . Also let  $d_H(\cdot)$  denote the Hamming distance. For each block  $\hat{\mathbf{y}}_i$ , there are ten different  $\mathbf{y}_i \in GF(4)^3$  satisfying

$$\{\mathbf{y}_i: d_H(\mathbf{y}_i, \hat{\mathbf{y}}_i) \leq 1\} \quad i = 1, 2. \quad (2)$$

Each  $\mathbf{y}_i$ , taken as an information vector, uniquely defines a hexacodeword. Thus, construct the 20 hexacodewords from all  $\mathbf{y}_i$  satisfying (2). Finally, among the hexacodewords so constructed, select the codeword with the minimum overall metric as the output of the decoder. Along the lines of [7], the computational complexity of this algorithm is at most 107 real-number operations.  $\square$

### B. Bounded-Distance Property of Algorithm 1

Let  $2d$  be the minimum Euclidean distance between distinct symbols. The guaranteed error-correction radius of the code is thus  $\frac{1}{2}d_{\min}(H_6) = \frac{1}{2}\sqrt{d_H(H_6) \cdot (2d)^2} = 2d$ . Accordingly, if the energy of the noise is smaller than  $4d^2$ , there may occur at most three symbol errors in  $\hat{\mathbf{x}}$ . Hence, at least one

Paper approved by T. Aulin, the Editor for Coding and Communications Theory of the IEEE Communications Society. Manuscript received April 25, 1994; revised March 3, 1995, and August 10, 1995. This paper was presented in part at the International Symposium on Information Theory ISIT'94, Trondheim, Norway, 1994, and at the Israeli-French Workshop on Coding and Applications, Tel-Aviv, Israel, December 1994.

The authors are with the Department of Electrical Engineering-Systems, Tel-Aviv University, Ramat Aviv 69978, Tel-Aviv, Israel.

Publisher Item Identifier S 0090-6778(96)03133-9.

of the two blocks contains not more than one error. Since all events of up to one symbol error per block are considered, the correct codeword must be among the 20 constructed codewords. The correct codeword will obviously be selected as the output of the decoder as it has the minimal metric among all hexacodewords.

### C. Further Complexity Reduction—Algorithm 2

Further reduction in decoding complexity, while maintaining the same guaranteed error correction radius, may be achieved by refining Algorithm 1. One such refinement which decreases decoding complexity to just 57 real operations is straightforward. Instead of constructing the 20 codewords from all  $y_i \in GF(4)^3$  satisfying (2), first locate the *least* reliable symbol in each block

$$\begin{aligned} \mathcal{M}(\hat{x}_{\ell_1}) &= \max\{\mathcal{M}(\hat{x}_1), \mathcal{M}(\hat{x}_2), \mathcal{M}(\hat{x}_3)\} & \ell_1 &\in \{1, 2, 3\} \\ \mathcal{M}(\hat{x}_{\ell_2}) &= \max\{\mathcal{M}(\hat{x}_4), \mathcal{M}(\hat{x}_5), \mathcal{M}(\hat{x}_6)\} & \ell_2 &\in \{4, 5, 6\} \end{aligned}$$

where  $\mathcal{M}(\hat{x}_i)$  denotes the metric of the symbol  $\hat{x}_i$ ;  $i = 1, \dots, 6$ . Then, for each block, construct four hexacodewords: one from each of the three information vectors  $y_i$  that differ from  $\hat{y}_i$  in coordinate  $\ell_i$ , and one from the vector  $\hat{y}_i$  itself. Finally, among the eight codewords so constructed, select the one with the minimum metric as the output of the decoder.  $\square$

Recall that as long as the magnitude of the noise is within the guaranteed error correction radius of the code, its energy is upper bounded by  $4d^2$ . Hence, the energy of the noise that is added to one of the blocks must be smaller than  $2d^2$ , and this block may contain a single symbol-error at the most. Obviously, for correct decoding, Algorithm 2 must locate this error. According to the following lemma, this error is always located.

**Lemma 1:** If the energy of the noise that is added to a block is smaller than  $2d^2$ , then any single symbol error in that block will be located by Algorithm 1.

*Proof:* Without loss of generality, let  $\hat{x}_1$ , with metric  $m_1^2$ , be the symbol in error—i.e.,  $\hat{x}_1 \neq x_1$ . Since the minimum Euclidean distance between symbols was taken to be  $2d$ , the noise energy added to  $x_1$  satisfies  $\varepsilon_1 \geq (2d - m_1)^2$ . Also let  $m_2^2$  be the metric of  $\hat{x}_2$ , where  $\hat{x}_2 = x_2$ . The noise energy added to  $x_2$  satisfies  $\varepsilon_2 = m_2^2$ . Since it is given that the noise energy added to the block satisfies  $\varepsilon_1 + \varepsilon_2 < 2d^2$ , then  $(2d - m_1)^2 + m_2^2 < 2d^2$ . Adding  $m_1^2$  to both sides, the last inequality may be reduced to  $m_1^2 - m_2^2 > 2(m_1 - d)^2$ . Thus obviously,  $m_1^2 - m_2^2 > 0$ , and  $m_1^2 > m_2^2$ . Since Algorithm 2 selects the symbol with the highest metric in the block, the Lemma is proved.  $\square$

At this point we note that the penalty paid for the above complexity reduction is in coding gain. Simulation results of the Leech decoder based on both versions of the above hexadecoders, revealing a very small coding gain loss, are given in Section IV.

### III. ON DECODING THE LEECH LATTICE AND GOLAY CODE

By substituting the proposed hexadecoders into the decoding algorithms of the Golay code,  $C_{24}$ , and the Leech lattice,  $\Lambda_{24}$ , of [7], more efficient decoding algorithms for these codes are

TABLE I  
DECODING OF THE LEECH LATTICE,  $H_{24}$ ; LEECH HALF LATTICE,  $Q_{24}$ ;  
LEECH QUARTER LATTICE [6], AND BD: BOUNDED-DISTANCE

Year	Devisers	Decoding idea	gain loss	complexity
1993	Vardy and Be'ery [6]	ML decoding of $4 \times Q_{24}$ using ML hexadecoder	optimal	3,596
1989	Forney [13]	BD decoding of $2 \times H_{24}$ using ML Golay decoder of [12]	-0.1 dB	~2000
		using Golay decoder of [6]	-0.1 dB	~1500
1993	Amrani, Be'ery, Vardy, Sun and van Tilborg [7]	BD decoding of $4 \times Q_{24}$ using ML hexadecoder	-0.1 dB	1007
now	Amrani and Be'ery	BD decoding of $4 \times Q_{24}$ using BD hexadecoder 1	-0.1 dB	719
		using BD hexadecoder 2	-0.2 dB	519

obtained. Moreover, the obtained algorithms (as those of [7]) have the *same* guaranteed error correction radius as that of maximum-likelihood decoding. This may be easily verified as follows. The algorithms in [7] and the hexacode were shown to have the same guaranteed error correction radius, meaning that any bounded-distance hexadecoder, which decodes correctly at least up to the guaranteed error correction radius of the hexacode, will preserve the guaranteed correction radius of the algorithms of [7].

Table I summarizes the *worst case* computational complexity and the coding gain loss of the obtained Leech decoders in comparison to the best known decoders of the Leech lattice [6], [7]. Specific coding gains of the Leech decoders for various bit-error rate (BER) values may be found in Section IV and Fig. 1. For the Golay code, the worst case complexity of decoding is reduced to 291, respectively 187, real-number operations when substituting Algorithm 1, respectively Algorithm 2, into the Golay decoder of [7]. Compare this to 651, respectively 431, operations required by the best known maximum likelihood, respectively bounded distance, decoders of  $C_{24}$  [5], [7].

It is customary to state the complexity of decoding in terms of real-number operations, since the latter provides a good estimate of the actual decoding time and the hardware implementation complexity. However, it is noteworthy that the proposed algorithms are also most efficient in terms of the number of  $GF(4)$  operations (only 376, respectively 200, operations, as compared to 1200  $GF(4)$  operations in [6], [7]). We note that further reduction in real-number complexity may be obtained for the price of a considerable increase in  $GF(4)$  complexity.

### IV. SIMULATION RESULTS AND COMPARISON

Comprehensive computer simulations were performed for the AWGN channel model. The simulation results of Algorithm 1 exhibit a remarkable resemblance to those of maximum likelihood decoding of  $H_6$ . Indeed, the improved  $\Lambda_{24}$  decoder obtained by substituting Algorithm 1 into the decoder of [7], performs practically the same as the original algorithm of [7]. In fact, when plotted together (in the scale of Fig. 1), their two traces coincide. The results also show that relative to maximum likelihood decoding, the coding gain loss of the improved decoder is uniformly less than 0.1 dB for BER ranging from  $10^{-1}$  to  $10^{-7}$ . The improved bounded-

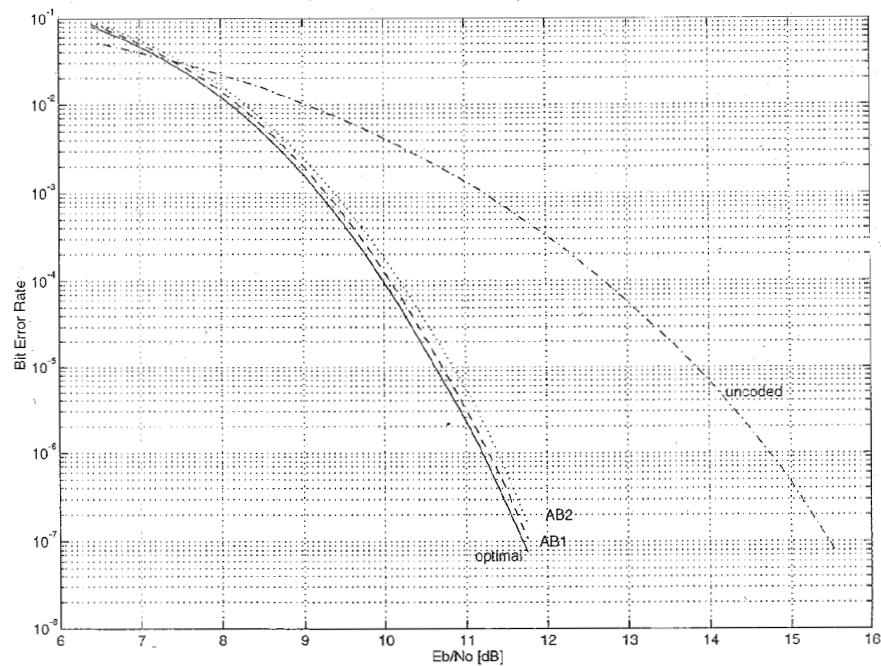


Fig. 1. Bit error rate comparison for the two leech decoders versus optimal and uncoded performance.

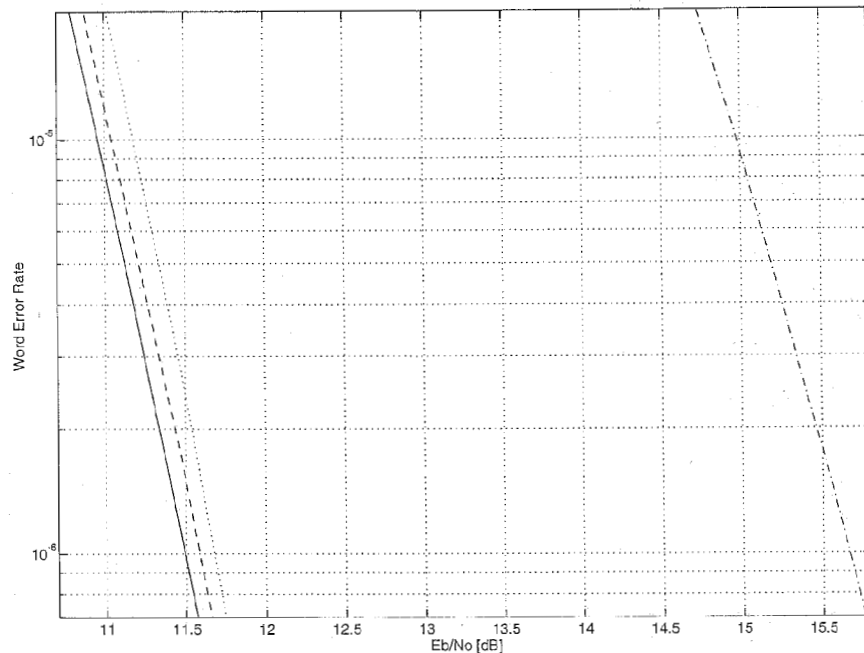


Fig. 2. Word error rate comparison for the leech decoders versus optimal and uncoded performance.

distance decoder for  $\Lambda_{24}$ , based on Algorithm 2, is only 0.1 dB worse than the above algorithm, thus, its coding gain loss is only 0.2 dB relative to the maximum likelihood decoder for the aforementioned BER range. Note that a square 64-QAM constellation was employed by the simulation for generating the  $\Lambda_{24}$  codewords, as in [6], [7], and [10]. In Fig. 1, we present the BER simulation results of the improved bounded-distance Leech lattice decoders versus maximum-

likelihood decoding (solid line). The dashed curve (AB1), denotes the results of the decoder based on Algorithm 1, while the dotted curve (AB2), denotes the results of the decoder based on Algorithm 2. The uncoded curve is given for reference (dashed-dotted line). Fig. 2 presents the **word error rate** curves, enlarged for the range of  $10^{-5}$  to  $10^{-6}$ .

It is interesting to compare these results to some well known trellis codes. Forney [11] presents some tables of known trellis

TABLE II  
COMPARISON OF TRELIS CODES AND THE LEECH LATTICE

Code	Rate	$\tilde{N}_D$	$\gamma$ (dB)	$\gamma_{eff}$ (dB)	
AB2	$\Lambda_{24}$	1/2	44	6.02	4.00
Eyuboglu	2D 8 state	1/2	44	3.52	3.32
AB1	$\Lambda_{24}$	1/2	60	6.02	4.10
Calderbank-Sloane	2D 8 state	1/2	72	4.39	3.79
ABVSvT [7]	$\Lambda_{24}$	1/2	84	6.02	4.10
Eyuboglu	2D 16 state	1/2	128	4.77	4.37
Vardy-Be'ery [6]	$\Lambda_{24}$	1/2	300	6.02	4.20
AB2	$H_{24}$	2/3	22	5.77	3.80
AB1	$H_{24}$	2/3	30	5.77	3.90
Wei	4D 8 state	2/3	44	4.52	3.82
Wei	4D 16 state	2/3	72	4.52	4.20
Gallager-Calderbank-Sloane	4D 8 state	3/4	64	4.52	3.82
Calderbank-Sloane	8D 8 state	3/4	90	5.27	3.75
Wei	8D 16 state	3/4	104	5.27	4.01

codes, along with their important parameters and performance. In Table II, we list some of the best trellis codes found in [11], with similar effective coding gains,  $\gamma_{eff}$ , and normalized (to two dimensions) decoding complexities,  $\tilde{N}_D$ , to those of the proposed algorithms. The table is divided into different basic code rates. For the rate 2/3, we present the results that were obtained for the Leech half-lattice,  $H_{24}$ , by using the same algorithms. Note that the 4D 16 state Wei code is one of the three codes supported by the emerging V.34 (formerly V.fast) ITU-T modem standard. We would like to point out that the effective coding gains were obtained using Forney's estimate [11] for the trellis codes, and from the simulation results for  $\Lambda_{24}$  and  $H_{24}$ . All effective coding gains are given for *word error rates* of the order of  $10^{-6}$ .

## ACKNOWLEDGMENT

The authors wish to thank A. Vardy for helpful discussions as well as all the anonymous reviewers for their comments and suggestions.

## REFERENCES

- [1] R. E. Peile, "A co-designed coding, modulation and equalization scheme for transmission of 155.52 mbit/s data over a 72 mhz intelsat transponder. Part I: coding and modulation performance," *Int. J. Satellite Commun.*, vol. 11, pp. 313-333, 1993.
- [2] G. R. Lang and F. M. Longstaff, "A Leech lattice modem," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 968-973, 1989.
- [3] J. P. Adoul and M. Barth, "Nearest neighbor algorithm for spherical codes from the leech lattice," *IEEE Trans. Inform. Theory*, vol. 34, pp. 1188-1202, 1988.
- [4] M. V. Eyuboglu and G. D. Forney, "Lattice and trellis quantization with lattice- and trellis-bounded codebooks—high-rate theory for memoryless sources," *IEEE Trans. Inform. Theory*, vol. 39, pp. 46-59, 1993.
- [5] A. Vardy and Y. Be'ery, "More efficient soft-decision decoding of the Golay codes," *IEEE Trans. Inform. Theory*, vol. 37, pp. 667-672, 1991.
- [6] ———, "Maximum-likelihood decoding of the Leech lattice," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1435-1444, 1993.
- [7] O. Amrani, Y. Be'ery, A. Vardy, F.-W. Sun, and H. C. A. van Tilborg, "The Leech lattice and the Golay code: bounded-distance decoding and multilevel constructions," *IEEE Trans. Inform. Theory*, vol. 40, pp. 1030-1043, 1994.
- [8] O. Amrani and Y. Be'ery, "Efficient bounded-distance decoding of the Hexacode and associated decoders for the Leech lattice and the Golay code," in *Proc. ISIT*, Trondheim, Norway, 1994, p. 400.
- [9] F.-W. Sun and H. C. A. van Tilborg, "More efficient bounded-distance decoding of the Golay code and the Leech lattice," in *Proc. ISIT*, Trondheim, Norway, 1994, p. 399.
- [10] Y. Be'ery, B. Shahar, and J. Snyders, "Fast decoding of the Leech lattice," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 959-967, 1989.
- [11] G. D. Forney Jr., "Coset codes-Part I: Introduction and geometrical classification," *IEEE Trans. Inform. Theory*, vol. 34, pp. 1123-1151, 1988.
- [12] J. Snyders and Y. Be'ery, "Maximum likelihood soft decoding of binary block codes and decoders for the Golay codes," *IEEE Trans. Inform. Theory*, vol. 35, pp. 963-957, 1989.
- [13] G. D. Forney Jr., "A bounded-distance decoding algorithm for the Leech lattice, with generalizations," *IEEE Trans. Inform. Theory*, vol. 35, pp. 906-909, 1989.
- [14] O. Amrani and Y. Be'ery, "Methods for efficient bounded distance decoding for a family of block codes and associated error correction methods for the hexacode, the golay code, and the leech lattice," Israel Patent Application No. 116087.