On the Voronoi Neighbor Ratio for Binary Linear Block Codes

Erik Agrell

***Abstract*—**Soft-decision decoding of block codes is regarded as the geometrical problem of identifying the Voronoi region within which a given input vector lies. A measure, called the neighbor ratio, is proposed to characterize how many facets a Voronoi region has. Theory and algorithms are presented to determine the neighbor ratio for binary linear block codes and results are given for several types of codes. An asymptotic analysis for long codes reveals that the neighbor ratio depends on whether the code rate is less than 1/2 or not. For rates below this threshold, all pairs of codewords tend to share a Voronoi facet; for higher rates, a relatively small fraction of them do.

***Index Terms*—**Binary linear block codes, Gaussian channel, Voronoi regions, neighbor ratio, asymptotic properties, soft-decision decoding.

I. Introduction

A channel decoder is, in its common form, a device that receives a sequence of values from the demodulator and outputs another sequence, selected from a predefined set of codewords. This form, the *hard decision* decoder, assumes that the demodulator has made a decision on each transmitted symbol. However, 2–3 dB can be gained if more information from the demodulator can be utilized [1, pp. 518–522], [2, p. 141]. We could let the demodulator output not only the detected symbols, but also a measure on how reliable each detection is. An alternative way to achieve the same effect would be to let the demodulator deliver an unquantized *estimate* of the symbol in each time interval [3, pp. 464–473]. All decisions are postponed to the channel decoder. This approach leads to the *soft-decision* decoder.

In soft-decision decoding, the channel decoder makes no decision until it has received a sequence of symbol estimates corresponding to a whole codeword from the demodulator. Herein lies the strength of soft-decision decoding. The price for the coding gain is increased decoder complexity and real-valued computations. A large number of soft-decision decoding algorithms have been conceived since the sixties. Good summaries and literature surveys are given in [2, ch. 4], [4], and [5].

Most work on soft-decision decoding has been done for the *Gaussian channel*.[1] With this channel model, the sequence of estimates that the demodulator outputs can be regarded as a codeword with an added random noise vector, whose components are Gaussian and independent. Hence, the optimum channel decoder has a closest point problem to solve: For each input sequence, find the codeword with the minimum Euclidean distance to the sequence. This assumes that codewords are interpreted as points in Euclidean space, not just as strings of bits.

The minimum distance decoding rule partitions Euclidean space into a number of regions, one around each codeword. The regions, nowadays normally called *Voronoi regions* [6], can be defined as the set of all sequences that would be decoded as the same codeword. Soft-decision decoding is hence to determine which Voronoi region a received sequence belongs to.

We will confine the present study to the class of *binary linear block codes*, which possesses some appealing properties for soft-decision decoding, theoretically as well as practically. Firstly, a binary code is, when regarded as a set of points in Euclidean space, a spherical code, whose Voronoi regions were characterized already by Shannon in 1959 [7]. They are pyramidal polytopes, with flat sides and infinite size, and all of them share the same apex. Secondly, a linear code has (still assuming that it is binary) the property that all its Voronoi regions are congruent, so it is sufficient to analyze one of them, see below. A class of codes with this property was studied by Slepian in 1965 [8] and more generally in 1968 [9], and, even more generally, by Forney in 1991 [10]. The topic of codes for the Gaussian channel and their Voronoi regions is also addressed in, e.g., [11], [12], and references therein.

In Section II, the terminology to be used is introduced, and the neighbor ratio is defined. Some known methods to determine Voronoi regions are summarized in Section III, and a new method is presented that complements the old ones nicely. The neighbor ratio and related properties of some common codes are determined in Section IV. These results are generalized to longer codes in Section V, and some asymptotic properties are discerned. The two next Sections, VI and VII, consider in detail the asymptotic neighbor ratio as a function of the rate. Section VIII contains a discussion and a short summary.

II. Preliminaries

If $\mathcal{C}$ is a binary linear block code, we denote by $M(\mathcal{C})$ the number of codewords and by $n(\mathcal{C})$ the number of bits in a codeword (the code length).[2] The rate is defined as $R(\mathcal{C}) = k(\mathcal{C})/n(\mathcal{C})$, where $k(\mathcal{C}) = \log_2 M(\mathcal{C})$. The (Hamming) weight $w(\mathbf{c})$ of a binary codeword $\mathbf{c}$ is defined as the number of ones in it; if the bit values 0 and 1 are interpreted as coordinates in Euclidean space, the relation between weight and Euclidean norm is $w(\mathbf{c}) = \|\mathbf{c}\|^2$. The minimum weight of the code is

$$d(\mathcal{C}) = \min_{\mathbf{c} \in \mathcal{C} \setminus \{\mathbf{0}\}} w(\mathbf{c}) \tag{1}$$

where $\mathbf{0}$ is the all-zero vector.[3] The triplet $[n(\mathcal{C}), k(\mathcal{C}), d(\mathcal{C})]$ summarizes the most fundamental properties of a code $\mathcal{C}$; sometimes we will also use the abbreviated form $[n(\mathcal{C}), k(\mathcal{C})]$. The functions $\mathbf{G}(\mathcal{C})$ and $\mathbf{H}(\mathcal{C})$ represent any generator matrix and parity-check matrix, respectively, for the considered code.

With this nomenclature, the Voronoi region of $\mathbf{0}$ is the n-dimensional body

$$\mathcal{V}_0 = \left\{ \mathbf{x} \in \mathbb{R}^n : \ \|\mathbf{x}\|^2 \le \|\mathbf{x} - \mathbf{c}\|^2 \text{ for all } \mathbf{c} \in \mathcal{C} \right\}. \tag{2}$$

It can be shown that the Voronoi region of any codeword $\mathbf{c} \in \mathcal{C}$ can

This work was completed while the author was with the Department of Information Theory, Chalmers University of Technology, Göteborg, Sweden. He is now with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093-0407 USA.

[1] Traditionally, the term "Gaussian channel" also includes the requirements of additive and white noise.

[2] When there is no doubt of which code is concerned, we simply write these functions as M, n, etc., omitting the argument.

[3] Later, the symbol $\mathbf{0}$ will represent all-zero matrices, too. All-one vectors will be denoted with $\mathbf{1}$.

be expressed as[4]

$$\{\mathbf{c} + \mathbf{x}(\mathbf{I} - 2\ \mathrm{diag}(\mathbf{c})) : \quad \mathbf{x} \in \mathcal{Q}_0'\} \tag{3}$$

where diag(**c**) is the diagonal matrix having the elements of the vector **c** along the diagonal. Since this region is just a translation, rotation and reflection of $\mathcal{Q}_0'$, all Voronoi regions in a binary linear block code have the same shape, and $\mathcal{Q}_0'$ can be used as a representative of all of them. All significant properties of a code can be derived from $\mathcal{Q}_0'$ [10]. This correspondence is about the structure of $\mathcal{Q}_0'$ for various codes.

The complete specification of a multidimensional polytope, including lists of the vertices, edges, etc., and their interrelations, is surprisingly complex [13, chs. 1 and 6], and memory considerations alone are sufficient reason to abandon that data structure for n in the order of 10 or greater [14]. On the other hand, a Voronoi region is fully defined through a list of its sides, or *facets*, only, and such a list suffices for most purposes, including error analysis and soft-decision decoding. Edges and vertices do not play as important roles, as will be discussed in the following.

Consider the inequalities in (2) again; each one of them contributes one or zero facets to $\mathcal{Q}_0'$. The codewords whose inequalities define a facet are called *0-neighbors* in the code $\mathcal{C}$. Moreover, two codewords whose Voronoi regions share a common facet are called *neighbors*. If $\mathbf{c}_1$ and $\mathbf{c}_2$ are neighbors, then $\mathbf{c}_1 \oplus \mathbf{c}_2$ is a 0-neighbor, where $\oplus$ denotes addition over $GF(2)$. Denoting the set of 0-neighbors with $\mathcal{N}_0(\mathcal{C})$, the Voronoi region $\mathcal{Q}_0'$ can be equivalently written as

$$\mathcal{Q}_0' = \left\{\mathbf{x} \in \mathbb{R}^n : \quad \|\mathbf{x}\|^2 \le \|\mathbf{x} - \mathbf{c}\|^2 \text{ for all } \mathbf{c} \in \mathcal{N}_0(\mathcal{C})\right\} \tag{4}$$

which in a sense has removed all redundancy from (2).

The main results of this correspondence concern the number of neighbors in various codes. We propose the *neighbor ratio*

$$\Gamma(\mathcal{C}) = \frac{|\mathcal{N}_0(\mathcal{C})|}{M(\mathcal{C})} \tag{5}$$

to characterize the code in this sense. The neighbor ratio lies between 0 and 1, these values excluded. Of special interest is what happens with the neighbor ratio when the code length increases. Suppose that a constant $0 < R \le 1$ is given, and consider an infinite sequence of codes $\mathcal{C}_i$, $i = 1, 2, \cdots$, whose parameters satisfy $n(\mathcal{C}_i) \to \infty$ and $k(\mathcal{C}_i)/n(\mathcal{C}_i) \to R$ as $i \to \infty$. Then the *asymptotic neighbor ratio* $\Gamma_\infty(R)$ for this sequence of codes is defined as

$$\Gamma_\infty(R) = \lim_{i \to \infty} \Gamma(\mathcal{C}_i). \tag{6}$$

provided that the limit exists.

It is normally not possible to obtain an exact expression for the error probability P_e of soft-decision decoding, because it involves computing an n-dimensional integral over the polytope $\mathcal{Q}_0'$. A standard approximation is the *union bound*, which is

$$\begin{aligned} P_e &\le \sum_{\mathbf{c} \in \mathcal{C} \setminus \{\mathbf{0}\}} Q\left(\sqrt{w(\mathbf{c})\frac{2E_b}{N_0}}\right) \\ &= \sum_{i=1}^{n} A_i(\mathcal{C}) Q\left(\sqrt{i\frac{2E_b}{N_0}}\right) \end{aligned} \tag{7}$$

assuming equiprobable codewords and biorthogonal modulation (say, BPSK or QPSK) with bit energy E_b [2, pp. 29–30], [11]. The variance of the discrete-time Gaussian noise is $N_0/2$, $Q(x)$ denotes the integral $\int_x^\infty (2\pi)^{-1/2} \exp\left(-z^2/2\right) dz$, and $A_i(\mathcal{C})$ denotes a component of the *weight distribution*

$$A_i(\mathcal{C}) = |\{\mathbf{c} \in \mathcal{C} : w(\mathbf{c}) = i\}| \quad \text{for } i = 0, \cdots, n(\mathcal{C}) \tag{8}$$

where $|\cdot|$ means the cardinal number of a set. Hence, the weight distribution gives the number of codewords with a certain weight. Weight distributions of many codes were tabulated in [15, ch. 16]. Another important characteristic of a code is the *local weight distribution* with components

$$L_i(\mathcal{C}) = |\{\mathbf{c} \in \mathcal{N}_0(\mathcal{C}) : w(\mathbf{c}) = i\}| \quad \text{for } i = 0, \cdots, n(\mathcal{C}) \tag{9}$$

which gives the number of 0-neighbors with a certain weight. Straightforward combination of (4), (7) and (9) yields another bound on the error probability,

$$\begin{aligned} P_e &\le \sum_{\mathbf{c} \in \mathcal{N}_0(\mathcal{C})} Q\left(\sqrt{w(\mathbf{c})\frac{2E_b}{N_0}}\right) \\ &= \sum_{i=1}^{n} L_i(\mathcal{C}) Q\left(\sqrt{i\frac{2E_b}{N_0}}\right) \end{aligned} \tag{10}$$

which is tighter than the usual union bound [10]. However, this bound requires that the local weight distribution $L_i(\mathcal{C})$ can be found, which is much harder than to find $A_i(\mathcal{C})$. Methods to compute $L_i(\mathcal{C})$ are summarized in the next section, where also a third bound is given, being tighter than (7) but just as easy to compute.

Besides being a valuable theoretical instrument, the Voronoi regions in a code can be employed in the decoding process itself. This was suggested by Landau [16], who also pointed out the significance of the number of facets of the Voronoi regions. An iterative algorithm was developed by Hwang [17] and investigated in more detail by Butovitsch [18, pts. D–E]. The idea, which has also been considered for other point sets and in other applications [19], [20], is basically as follows: *(i)* Select a codeword **c** and compute its distance δ to the input **x**. *(ii)* Compute the distance δ' from a new codeword $\mathbf{c}' = \mathbf{c} \oplus \mathbf{a}$ to **x**, where $\mathbf{a} \in \mathcal{N}_0(\mathcal{C})$. If $\delta' < \delta$, set $\mathbf{c} := \mathbf{c}'$ and go to (ii). *(iii)* If $\mathcal{N}_0(\mathcal{C})$ contains unexamined codewords, go to (ii). Otherwise, output **c**. The complexity of this algorithm, which performs maximum-likelihood decoding, is proportional to $\Gamma(\mathcal{C})$.

III. Identification of Neighbors

In this section, rules to determine the neighbors in a binary linear block code are summarized. First, we adopt a geometric viewpoint on the problem and interpret some basic theory in geometric terms. Then, rules will be given to determine whether a given codeword is a 0-neighbor, and in some cases, to identify 0-neighbors and 0-nonneighbors based on their weights alone.

The n-dimensional binary space $\{0,1\}^n$ describes the vertices of an n-dimensional hypercube, and an $[n,k]$ binary code $\mathcal{C}$ forms a subset of these vertices. This special structure makes it possible to analyze the geometry of a binary code in much more detail than is possible for a general point set, which in turn leads to explicit results about the neighbors. The following properties characterize any binary linear code, regarded as a point set. (Actually, the two first properties apply to binary nonlinear codes as well.) The proofs, which we omit, follow immediately from the theory in [12].

(*i*) No three codewords can form an obtuse angle.

(*ii*) If the dots in Figure 1 are two codewords in the code, all codewords lie within the shaded region.

(*iii*) If and only if two given codewords are the only codewords on the sphere of Figure 1, they are neighbors.

[4] All vectors are row vectors.

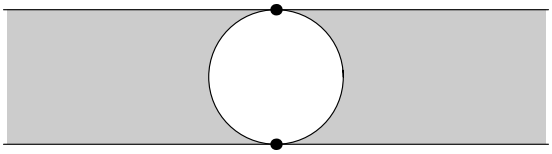

Fig. 1. Two parallel hyperplanes touching a hypersphere, where the points of contact are defined by any two codewords (dots). All codewords in a binary code lie between the planes, outside the sphere.

(*iv*) If three codewords form a right angle, there is a fourth codeword completing the rectangle.

(*v*) Two codewords forming the diagonal of a rectangle are nonneighbors. Two codewords that do not form any diagonal are neighbors.

Again regarding codewords as strings of bits, the last of these statements can be translated into the following important rule, which was first given in [17], though not in Voronoi terminology. To show that $\mathcal{N}_0(\mathcal{C})$ is equal to the "projecting set" of [17], compare Corollary 1 in [12] with Definition 2 in [17].

C rule: A codeword is a 0-neighbor if and only if it covers[5] no other nonzero codeword.

To establish whether a given codeword $\mathbf{c} \in \mathcal{C}$ is a 0-neighbor or not, it is not necessary to generate all codewords one by one and check if they are covered by $\mathbf{c}$. Instead, the test implied by the C rule can be performed by a sequence of row operations on the generator matrix $\mathbf{G}(\mathcal{C})$. An explicit algorithm for this purpose was presented in [12]. A condensed version of the algorithm can be formulated as the following rule.

G rule: Let $\mathbf{G}_0(\mathcal{C},\mathbf{c})$ denote the matrix formed by the columns of $\mathbf{G}(\mathcal{C})$ corresponding to positions where a given codeword $\mathbf{c} \in \mathcal{C}$ has zeros. Then $\mathbf{c}$ is a 0-neighbor if and only if $\operatorname{rank} \mathbf{G}_0(\mathcal{C},\mathbf{c}) = k(\mathcal{C}) - 1$.

We now introduce a useful dual of this rule. It can, just as the G rule, be proved through the C rule.

H rule: Let $\mathbf{H}_1(\mathcal{C},\mathbf{c})$ denote the matrix formed by the columns of $\mathbf{H}(\mathcal{C})$ corresponding to positions where a given codeword $\mathbf{c} \in \mathcal{C}$ has ones. Then $\mathbf{c}$ is a 0-neighbor if and only if $\operatorname{rank} \mathbf{H}_1(\mathcal{C},\mathbf{c}) = w(\mathbf{c}) - 1$.

The G and H rules complement each other nicely. Both of them yield the same results, but the complexity involved varies with the code parameters and with the weight of the codeword to test. When $\mathbf{G}_0(\mathcal{C},\mathbf{c})$ is a large matrix, $\mathbf{H}_1(\mathcal{C},\mathbf{c})$ is small. Generally, the speed of a G rule test increases for lower rates and higher weights, and vice versa.

Example: Is

$$\mathbf{c} = [1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0]$$

a 0-neighbor in the [7,4,3] Hamming code with

$$\mathbf{G}(\mathcal{C}) = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

as a generator matrix? The G rule answers in the affirmative, because

$$\mathbf{G}_0(\mathcal{C},\mathbf{c}) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

has a rank of $3 = k(\mathcal{C}) - 1$. Equivalently, we can employ the H rule. Selecting the proper columns of the parity-check matrix

$$\mathbf{H}(\mathcal{C}) = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

yields

$$\mathbf{H}_1(\mathcal{C},\mathbf{c}) = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

whose rank is $2 = w(\mathbf{c}) - 1$, which confirms that $\mathbf{c}$ is a 0-neighbor. ❑

Even though the time required to decide whether one given codeword is a 0-neighbor or not is moderate, the accumulated classification of all codewords in a code can be prohibitive. Thus, we would need a method to classify large sets of codewords simultaneously. Such a method can be based on the weight of a codeword, which sometimes provides enough information to make the decision. Specifically, the C rule assures that codewords with low enough weights are always 0-neighbors, and by upper-bounding the rank of $\mathbf{G}_0(\mathcal{C},\mathbf{c})$ or $\mathbf{H}_1(\mathcal{C},\mathbf{c})$, it can be shown that codewords with sufficiently high weights are never 0-neighbors. The two bounds thus obtained, which were given by Hwang [17], can be summarized in the following theorem.

Theorem 1: For any binary linear block code $\mathcal{C}$,

$$\{\mathbf{c} \in \mathcal{C}: \quad 1 \le w(\mathbf{c}) \le 2d(\mathcal{C}) - 1\} \subseteq \mathcal{N}_0(\mathcal{C}) \subseteq \{\mathbf{c} \in \mathcal{C}: \quad 1 \le w(\mathbf{c}) \le n(\mathcal{C}) - k(\mathcal{C}) + 1\}. \tag{11}$$

The left-hand side of the theorem states that the weight of any nonzero 0-nonneighbor $\mathbf{c}$ satisfies $w(\mathbf{c}) \ge 2d(\mathcal{C})$. A generalization of this property is possible. Suppose that a weight w is present in the weight distribution of a code (i.e., $A_w(\mathcal{C}) \ge 1$) and that we want to analyze the codewords with this weight. We can then attempt to write w as the sum of two nonzero integers, both of which are present in the same weight distribution. This can be done for all 0-nonneighbors, which is proved through the following theorem. It can be done for some 0-neighbors as well, so the theorem is not useful for identifying 0-nonneighbors.

Theorem 2: If the weight w of a nonzero codeword $\mathbf{c} \in \mathcal{C}$ cannot be written as $w = i + j$, where $i \ge 1$, $j \ge 1$, $A_i(\mathcal{C}) \ge 1$, and $A_j(\mathcal{C}) \ge 1$, then $\mathbf{c}$ is a 0-neighbor.

Proof of Theorem 2: From property (*v*) we see that any 0-nonneighbor $\mathbf{c} \in \mathcal{C} \setminus \{\mathbf{0}\}$ can be written as $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_2$, where $\mathbf{c}_1 \in \mathcal{C} \setminus \{\mathbf{0}\}$, $\mathbf{c}_2 \in \mathcal{C} \setminus \{\mathbf{0}\}$, and $\mathbf{c}_1 \cdot \mathbf{c}_2 = 0$. Thus, $w(\mathbf{c}) = \|\mathbf{c}\|^2 = \|\mathbf{c}_1\|^2 + \|\mathbf{c}_2\|^2 + 2\mathbf{c}_1 \cdot \mathbf{c}_2 = w(\mathbf{c}_1) + w(\mathbf{c}_2)$. ❑

To find the 0-neighbors $\mathcal{N}_0(\mathcal{C})$ in a code $\mathcal{C}$, the first thing to do would be to find the weight distribution [15, ch. 16] and combine it with Theorems 1 and 2. For some codes, especially short ones, Theorem 1 yields a complete description of $\mathcal{N}_0(\mathcal{C})$; otherwise, it leaves a subset of the codewords for individual examination. (Both cases will be illustrated in the next section.) Then either the G or the H rule is applied to the remaining subset, which possibly can be further reduced using the automorphism group of the code, see [12].

As an example of the information provided by the weight distribution, consider the [64,22,16] Reed-Muller code. It contains codewords with weights 0, 16, 24, 28, 32, 36, 40, 48, and 64. According to Theorem 1, all codewords with weights 16, 24, and 28 are 0-neighbors, and no codewords with weights 48 and 64 are. Furthermore, Theorem 2 classifies the weight-36 codewords as 0-neighbors. Weights 32 and 40 remain to be examined with the G

[5] A binary codeword $\mathbf{c}_1$ is said to *cover* another one, $\mathbf{c}_2$, if $\mathbf{c}_1$ has a one in all positions where $\mathbf{c}_2$ has a one [21, p. 63].

rule, which is less complex than the H rule in this case. (It turns out that all codewords with weight 40 are 0-nonneighbors, whereas both 0-neighbors and 0-nonneighbors have weight 32.)

Finally, an interesting consequence of the right-hand side of (11) is that the summation interval in (7) can be reduced. The obtained bound on the error probability,

$$P_e \le \sum_{i=1}^{n-k+1} A_i(\mathcal{C}) Q\left(\sqrt{i\frac{2E_b}{N_0}}\right), \tag{12}$$

combines the advantages of (7) and (10). The sum does not require more information about the code than does the usual union bound, but the resulting bound is tighter.

IV. Some Codes and Their 0-Neighbors

The tools discussed in the previous section were applied to several well-known codes. In anticipation of the asymptotic theory in the next sections, we summarize some general properties of code families below and list results for some specific codes in Table I. In the table, $w_- = 2d$ is the lowest weight possible for a nonzero 0-nonneighbor, $w_+ = n-k+1$ is the highest possible weight for a 0-neighbor, and $\mathcal{W} = \{\mathbf{c} \in \mathcal{C}: \quad w_- \le w(\mathbf{c}) \le w_+\}$ is the set bounded by these values. Hence, $|\mathcal{W}|/M$ is the proportion of codewords not being identified through Theorem 1. The table also gives the number of 0-neighbors, $|\mathcal{N}_0|$, the number of 0-neighbors at minimum distance, $L_d = A_d$, and the neighbor ratio, Γ. Other investigations of the neighbors in some codes are included in [17], [18], and [12].

Trivial codes: The 0-neighbors of the $[n,n,1]$ universe code, the $[n,n-1,2]$ even-weight code, and the $[n,1,n]$ repetition code are all determined by Theorem 1. The number of 0-neighbors is, respectively, n, $n(n-1)/2$, and 1.

Hamming codes: The Hamming codes and their relatives are almost completely determined by Theorem 1. The $\left[2^m-1, 2^m-m-1, 3\right]$ Hamming code has $w_- = 5$ and $w_+ = m+1$, so the gap between the bounds is relatively narrow. The neighbor ratio rapidly tends to zero with increasing code length. Conditions are similar for the $\left[2^m, 2^m-m-1, 4\right]$ extended Hamming code. Theorem 1 identifies all 0-neighbors for the $[32,26,4]$ and shorter extended Hamming codes and leaves only one weight for further analysis in the codes with lengths 64 and 128. In the $\left[2^m-1, m, 2^{m-1}\right]$ dual Hamming code, or simplex code, all nonzero codewords have the same weight. Hence, all of them are 0-neighbors, according to the C rule.

BCH codes: The Bose-Chaudhuri-Hocquenghem (BCH) codes, provide valuable information for the present study, because there is

TABLE I
PARAMETERS OF SOME COMMON CODES.

Type	$[n,k,d]$	R	w_-	w_+	$\lvert\mathcal{W}\rvert/M$	$\lvert\mathcal{N}_0\rvert$	L_d	Γ
Universe	$[n,n,1]$	1	2	1	0	n	n	$n2^{-n}$
Even-weight	$[n,n-1,2]$	$1-1/n$	4	2	0	$n(n-1)/2$	$n(n-1)/2$	$n(n-1)2^{-n}$
Repetition	$[n,1,n]$	$1/n$	$2n$	n	0	1	1	1/2
Hamming	[7,4,3]	0.571	6	4	0	14	7	0.875
Hamming	[15,11,3]	0.733	6	5	0	308	35	0.150
Hamming	[31,26,3]	0.839	6	6	$3.4\cdot10^{-4}$	20,336	155	0.000
Ext. Hamming	[8,4,4]	0.500	8	5	0	14	14	0.875
Ext. Hamming	[16,11,4]	0.688	8	6	0	588	140	0.287
Ext. Hamming	[32,26,4]	0.813	8	7	0	29,016	1,240	0.000
Dual Hamming	[7,3,4]	0.429	8	5	0	7	7	0.875
Dual Hamming	[15,4,8]	0.267	16	12	0	15	15	0.938
Dual Hamming	[31,5,16]	0.161	32	27	0	31	31	0.969
Golay	[23,12,7]	0.522	14	12	0	3,335	253	0.814
Ext. Golay	[24,12,8]	0.500	16	13	0	3,335	759	0.814
BCH	[15,5,7]	0.333	14	11	0	30	15	0.938
BCH	[15,7,5]	0.467	10	9	0	108	18	0.844
BCH	[31,6,15]	0.194	30	26	0	62	31	0.969
BCH	[31,11,11]	0.355	22	21	0	2,046	186	0.999
BCH	[31,16,7]	0.516	14	16	0.56	42,284	155	0.645
BCH	[31,21,5]	0.677	10	11	$6.1\cdot10^{-2}$	107,198	186	0.051
BCH	[63,7,31]	0.111	62	57	0	126	63	0.984
BCH	[63,10,27]	0.159	54	54	0	1,022	196	0.998
BCH	[63,16,23]	0.254	46	48	0	65,534	1,890	1.000
BCH	[63,18,21]	0.286	42	46	$5.5\cdot10^{-3}$	262,139	1,452	1.000
BCH	[63,24,15]	0.381	30	40	0.68	15,840,940	651	0.944
BCH	[63,30,13]	0.476	26	34	0.75	695,053,516	1,764	0.647
BCH	[63,36,11]	0.571	22	28	0.22	10,198,908,660	5,670	0.148
RM	[16,5,8]	0.313	16	12	0	30	30	0.938
RM	[32,6,16]	0.188	32	27	0	62	62	0.969
RM	[32,16,8]	0.500	16	17	0.56	42,284	620	0.645
RM	[64,7,32]	0.109	64	58	0	126	126	0.984
RM	[64,22,16]	0.344	32	43	0.72	3,821,804	2,604	0.911

a relatively large number of codes with the same length within the family. Several examples are listed in the table. We save the comments and generalizations until the next section.

RM codes: The Reed-Muller (RM) codes constitute another large family, which contains several other types of codes as special cases. A general RM code is denoted $\mathscr{R}(u,m)$, where u is the *order*, and its parameters are

$$\left[2^m, \sum_{i=0}^{r}\binom{m}{i}, 2^{m-u}\right]. \tag{13}$$

$\mathscr{R}(0,m)$ is the $\left[2^m,1,2^m\right]$ repetition code, $\mathscr{R}(m-2,m)$ is the $\left[2^m,2^m-m-1,4\right]$ extended Hamming code, and $\mathscr{R}(m-1,m)$ is the $\left[2^m,2^m-1,2\right]$ even-weight code. The first-order RM code $\mathscr{R}(1,m)$ is the dual of the extended Hamming code. All codewords in $\mathscr{R}(1,m)$, except $\mathbf{0}$ and $\mathbf{1}$, have the same weight, 2^{m-1}, and these $M-2$ are thus the 0-neighbors.

V. The Neighbor Ratio of Long Codes

Exact values of the neighbor ratio were computed for many codes in the previous section. In this section, we study the neighbor ratio for long codes, that is, codes for which $n(\mathscr{C})$ is large. We will observe that low-rate ($R(\mathscr{C})<1/2$) and high-rate ($R(\mathscr{C})>1/2$) codes[6] behave quite differently, which will be explained in subsequent sections.

For codes with many codewords (say, $k(\mathscr{C})>35$), it is too time-consuming to find exact neighbor ratios through the methods described is Section III. However, an accurate estimate of the neighbor ratio can be obtained by studying random codewords, equiprobably selected from $\mathscr{C}$. The probability that such a codeword is a 0-neighbor is equal to $\Gamma(\mathscr{C})$.

Figure 2 extends some of the results of Table I to longer codes, suggesting an asymptotic behavior of the neighbor ratio. In the diagram, the neighbor ratio of primitive binary BCH codes has been estimated using 1 million randomly chosen codewords from each code. It can be observed that for increasing code length, $\Gamma(\mathscr{C})$ tends to either 0 or 1, depending on the rate. The threshold appears to be at a rate equal to $1/2$:

$$\Gamma_\infty(R)=\begin{cases}1 & \text{if } R<1/2\\ 0 & \text{if } R>1/2\end{cases}. \tag{14}$$

Note also that the neighbor ratio is *not* a monotonically decreasing function of the rate. This is because the low-rate BCH codes have only two 0-nonneighbors, $\mathbf{0}$ and $\mathbf{1}$. For such codes, $\Gamma(\mathscr{C})=(M(\mathscr{C})-2)/M(\mathscr{C})=1-2^{1-R(\mathscr{C})n(\mathscr{C})}$, which, for a constant $n(\mathscr{C})$, is a slowly increasing function of $R(\mathscr{C})$.

Figure 3 presents the corresponding results for the RM codes $\mathscr{R}(u,m)$. The appearance of the curves divides the set of RM codes into two types, depending on whether m is even or odd. Only the latter type contains codes with $R(\mathscr{C})=1/2$, which explains the perceived difference between the two types. However, both of them approach the same step function, namely, the function that was also observed for BCH codes above.

This raises the question how general the pattern (14) is. Does it hold for more codes than just BCH and RM codes?

VI. Asymptotic Analysis: High-Rate Case

A partial answer to the question above is that the first part of (14)

[6] This precise definition of "low-rate" and "high-rate" follows Forney [22].

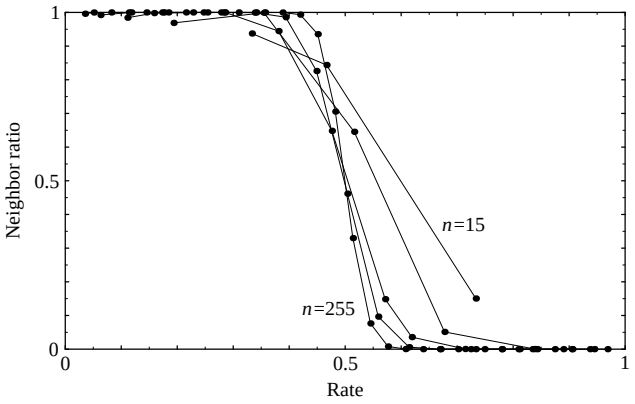

Fig. 2. The estimated neighbor ratio $\Gamma(\mathscr{C})$ vs. the rate $R(\mathscr{C})$ for various primitive binary BCH codes. The lines connect codes with $n(\mathscr{C})=15$, 31, 63, 127, and 255, respectively.

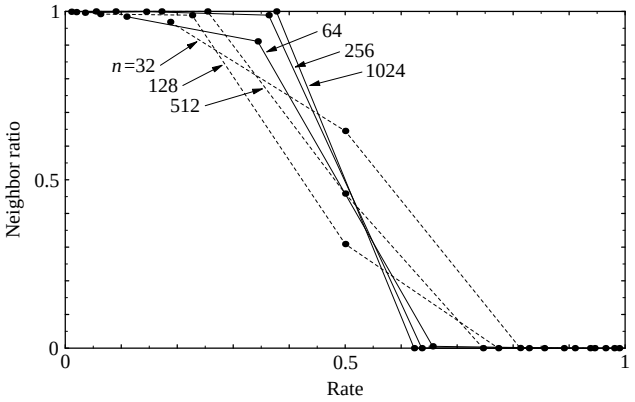

Fig. 3. The neighbor ratio $\Gamma(\mathscr{C})$ vs. the rate $R(\mathscr{C})$ for RM codes. Lines connect codes with the same length $n=2^m$; solid lines for even values of m and dashed lines for odd.

holds quite generally: $\Gamma_\infty(R)=0$ if $R>1/2$, for any binary linear block code. This will be proved in the following; we will return to the low-rate case, which is more complicated, in the next section.

As an aid in the analysis, it is convenient to define two types of codes, compressible and incompressible. With a *compressible* code, we mean a code that satisfies one (or both) of the following conditions:

(*i*) There is a position in which all codewords are zero.

(*ii*) There is a pair of positions in which all codewords have two equal bits.

An *incompressible* code is a code for which neither of these statements is true.

Obviously, a compressible code $\mathscr{C}$ can be made incompressible by removing every all-zero column and every duplicate column from the $\mathbf{G}$ matrix. The parameters of the incompressible code $\mathscr{C}'$ thus created satisfy $n(\mathscr{C}')<n(\mathscr{C})$, $k(\mathscr{C}')=k(\mathscr{C})$, and $d(\mathscr{C}')\le d(\mathscr{C})$. Furthermore, the neighbor ratios are the same for the two codes, $\Gamma(\mathscr{C}')=\Gamma(\mathscr{C})$, which is easily shown using the G rule.

The theoretical usefulness of the concept of compressible codes comes from the following theorem, which implies that in long, incompressible codes, almost all codewords have a weight close to $n/2$. Introducing the *relative weight* $\rho(\mathbf{c})=w(\mathbf{c})/n$, a proof can be formulated based on [21].

Theorem 3: For any incompressible binary linear block code $\mathscr{C}$, the proportion of the codewords $\mathbf{c}\in\mathscr{C}$ whose relative weights $\rho(\mathbf{c})$ satisfy $|\rho(\mathbf{c})-1/2|\ge\varepsilon$ is less than or equal to $\left(4\varepsilon^2 n(\mathscr{C})\right)^{-1}$ for any

$\varepsilon > 0$.

Proof of Theorem 3: The minimum number of linearly dependent columns in **G** gives the minimum distance of the dual code; for an incompressible code, this value is greater than 2. Thus [21, p. 131], the relative weights have a "mean"

$$\bar{\rho} = \frac{1}{M(\mathcal{C})} \sum_{\mathbf{c}\in\mathcal{C}} \rho(\mathbf{c}) = \frac{1}{2} \tag{15}$$

and a "variance"

$$\sigma_\rho^2 = \frac{1}{M(\mathcal{C})} \sum_{\mathbf{c}\in\mathcal{C}} (\rho(\mathbf{c}) - \bar{\rho})^2 = \frac{1}{4n(\mathcal{C})}. \tag{16}$$

Applying Chebyshev's inequality (or, more precisely, a deterministic variant thereof) completes the proof. ❑

Returning to the neighbor ratio, we can now concretize its behavior for long codes. First, Theorem 4 gives an upper bound on the neighbor ratio of high-rate incompressible codes. The upper bound converges to zero with increasing code length $n(\mathcal{C})$. Theorem 5 is a generalization to allow for compressible codes, too.

Theorem 4: For an incompressible binary linear block code whose parameters satisfy

$$\frac{k}{n} > \frac{1}{2} + \frac{1}{n} \tag{17}$$

the neighbor ratio is upper-bounded by

$$\Gamma \le \left(\frac{k}{n} - \frac{1}{2} - \frac{1}{n}\right)^{-2} (4n)^{-1}. \tag{18}$$

Proof of Theorem 4: According to the right-hand side of Theorem 1, the set $\mathcal{N}_0$ of all 0-neighbors of **0** satisfies

$$\begin{aligned} \mathcal{N}_0 &\subseteq \{\mathbf{c}\in\mathcal{C}: \quad w(\mathbf{c}) \le n-k+1\} \\ &= \left\{\mathbf{c}\in\mathcal{C}: \quad \rho(\mathbf{c}) \le \frac{1}{2} - \left(\frac{k}{n} - \frac{1}{2} - \frac{1}{n}\right)\right\} \\ &\subseteq \left\{\mathbf{c}\in\mathcal{C}: \quad \left|\rho(\mathbf{c}) - \frac{1}{2}\right| \ge \frac{k}{n} - \frac{1}{2} - \frac{1}{n}\right\}. \end{aligned} \tag{19}$$

Theorem 3 is now applicable to the set given by the last line. The proportion of $\mathcal{C}$ that is contained in this set is upper-bounded by $\left(4\varepsilon^2 n\right)^{-1}$, where $\varepsilon = k/n - 1/2 - 1/n$. The proportion of $\mathcal{C}$ that is contained in the set on the left-hand side, $\mathcal{N}_0$, is by definition $\Gamma(n,k)$, which completes the proof. ❑

The constraint to incompressible codes can be immediately relaxed, which is done in the following theorem.

Theorem 5: For *any* binary linear block code whose parameters satisfy

$$\frac{k}{n} > \frac{1}{2} + \frac{1}{n} \tag{20}$$

the neighbor ratio is upper-bounded by

$$\Gamma \le \left(\frac{k}{n} - \frac{1}{2} - \frac{1}{n}\right)^{-2} (4n)^{-1}. \tag{21}$$

Proof of Theorem 5: As mentioned above, there is for any binary linear block code $\mathcal{C}$ an incompressible code $\mathcal{C}'$ with the same number of 0-neighbors, where the parameters satisfy $n(\mathcal{C}') \le n(\mathcal{C})$ and $k(\mathcal{C}') = k(\mathcal{C})$. Hence, (20) implies

$$0 < 2k(\mathcal{C}) - n(\mathcal{C}) - 2 \le 2k(\mathcal{C}') - n(\mathcal{C}') - 2 \tag{22}$$

or

$$\frac{k(\mathcal{C}')}{n(\mathcal{C}')} > \frac{1}{2} + \frac{1}{n(\mathcal{C}')}. \tag{23}$$

This inequality shows that Theorem 4 is applicable to the incompressible code $\mathcal{C}'$. Since $\mathcal{C}$ and $\mathcal{C}'$ have the same neighbor ratio,

$$\begin{aligned} \Gamma(\mathcal{C}) &= \Gamma(\mathcal{C}') \\ &\le \left(\frac{k(\mathcal{C}')}{n(\mathcal{C}')} - \frac{1}{2} - \frac{1}{n(\mathcal{C}')}\right)^{-2} \left(4n(\mathcal{C}')\right)^{-1} \\ &= \frac{n(\mathcal{C}')}{n(\mathcal{C})} \left(\frac{2k(\mathcal{C}) - n(\mathcal{C}) - 2}{2k(\mathcal{C}') - n(\mathcal{C}') - 2}\right)^2 \cdot \\ &\qquad \frac{1}{4n(\mathcal{C})\left(k(\mathcal{C})/n(\mathcal{C}) - 1/2 - 1/n(\mathcal{C})\right)^2} \\ &\le \left(\frac{k(\mathcal{C})}{n(\mathcal{C})} - \frac{1}{2} - \frac{1}{n(\mathcal{C})}\right)^{-2} \left(4n(\mathcal{C})\right)^{-1} \end{aligned} \tag{24}$$

where the last inequality follows from (22). ❑

Now we study (21) for increasing code length n, where the ratio k/n approaches a constant rate R. Then Γ tends to $\Gamma_\infty(R)$, $(k/n - 1/2 - 1/n)^{-2}$ tends to the positive constant $(R-1/2)^{-2}$, and $(4n)^{-1}$ tends to zero. This proves the following important corollary.

Corollary 6: The asymptotic neighbor ratio satisfies

$$\Gamma_\infty(R) = 0 \quad \text{if } R > 1/2 \tag{25}$$

for any sequence of binary linear block codes such that $\Gamma_\infty(R)$ exists.

Thereby the high-rate study is complete, and the right-hand sides of the diagrams in Figures 2 and 3 have been explained.

VII. Asymptotic Analysis: Low-Rate Case

Now we turn to the left-hand side of the diagrams, that is, we consider low-rate codes. It is tempting to suggest that $\Gamma_\infty(R) = 1$ whenever $R < 1/2$. This would be a nice counterpart to Corollary 6, and Figures 2 and 3 indeed support the suggestion for two common families of codes.

Unfortunately, the hypothesis is false, though not very often. In practice, we have observed a threshold at $R = 1/2$ for many sequences of codes—indeed, for all codes except those that were explicitly conceived to violate the hypothesis. We will try to explain this behavior in the following. First, we study the set of all binary linear block codes and show that almost all codes in this set have the $R = 1/2$ threshold. Then, to emphasize *almost*, a class of codes is given for which $\Gamma_\infty(R) = 0$ for all $0 < R < 1$. The section is concluded by some observations on the relation between minimum weight and asymptotic neighbor ratio.

To investigate the asymptotic neighbor ratio for low rates, an averaging argument is employed over a large number of codes of the same size. Such arguments have been successfully employed in the past, ever since Shannon employed a random coding argument to prove the channel coding theorem [23], [24, pp. 198–203]. That the method can produce quite strong results is to some extent explained by Pierce's results [25], according to which the Gilbert-Varshamov bound[7] is tight for *almost all* binary linear block codes, if $n(\mathcal{C})$ is large. Hence, a random code is a good code. (See also [23] and [27] regarding the error probability of random codes.) We will now verify the observation in Section V, that $\Gamma(\mathcal{C})$ is close to 1 for $R(\mathcal{C}) < 1/2$ and large $n(\mathcal{C})$, for random codes. We first give a theorem about a random codeword in a random code in $\mathcal{L}(n,R)$, which denotes the set of all binary linear block codes $\mathcal{C}$ for which

[7] The Gilbert-Varshamov bound [26, ch. 4] is still the best known lower bound on the highest possible $d(\mathcal{C})/n(\mathcal{C})$, as a function of the code rate.

$n(\mathcal{C}) = n$ and $R(\mathcal{C}) = R$. The proof, which is lengthy and not too enlightening per se, is only sketched.

Theorem 7: If a code is selected equiprobably from $\mathscr{L}(n,R)$, where $R < 1/2$, and a codeword is selected equiprobably from this code, the probability that this codeword is a 0-neighbor tends to 1 as $n \to \infty$.

Outline of proof of Theorem 7: Suppose that two constants n and k are given and consider a pair $(\mathbf{H},\mathbf{c})$, where $\mathbf{H} \in \{0,1\}^{n\times(n-k)}$ and $\mathbf{c} \in \{0,1\}^n$. Let $\mathbf{H}_1$ denote the $w(\mathbf{c}) \times (n-k)$ matrix that consists of the columns of $\mathbf{H}$ in the positions where $\mathbf{c}$ has ones. For $\mathbf{c}$ to be a 0-neighbor in a code with parity check matrix $\mathbf{H}$, the following three conditions must be satisfied:

(A) rank $\mathbf{H} = n-k$. ($\mathbf{H}$ is the parity check matrix of an $[n,k]$ code).

(B) $\mathbf{1}\,\mathbf{H}_1^T = \mathbf{0}$ ($\mathbf{c}$ is a codeword).

(C) rank $\mathbf{H}_1 = w(\mathbf{c}) - 1$ ($\mathbf{c}$ is a 0-neighbor).

If $\mathbf{H}$ and $\mathbf{c}$ are random variables, equiprobably selected from their respective set, the probability P_n that a random codeword in a random code is a 0-neighbor can be expressed in terms of the events A, B, and C:

$$P_n = \Pr\{C | A \wedge B\}. \tag{26}$$

Using standard probability rules, P_n can be lower-bounded by

$$P_n = 1 - \frac{\Pr\{A \vee C | B\} - \Pr\{C|B\}}{\Pr\{A|B\}} \geq 1 - \frac{1 - \Pr\{C|B\}}{\Pr\{A|B\}}. \tag{27}$$

The two conditional probabilities in the bound can be evaluated by counting the total number of pairs $(\mathbf{H},\mathbf{c})$ that satisfy B, A and B, and B and C. The results that come forth after a tedious excursion into combinatorics are that $\Pr\{A|B\} \to 1$ as $n \to \infty$ and $k \to \infty$, and $\Pr\{C|B\} \to 1$ as $n \to \infty$ if $k < n/2$. Insertion of these limits into (27) completes the proof. ❑

This theorem can be translated into the neighbor ratio of codes, which is done in Theorem 8. The essence is that almost all codes have a ratio close to one.

Theorem 8: For any $R < 1/2$ and any $\varepsilon > 0$, the proportion of codes $\mathcal{C} \in \mathscr{L}(n,R)$ that satisfy $\Gamma(\mathcal{C}) > 1-\varepsilon$ tends to 1 as $n \to \infty$.

Proof of Theorem 8: The probability P_n that a random codeword of a random code in $\mathscr{L}(n,R)$ is a 0-neighbor is, assuming equiprobable selection,

$$P_n = \frac{1}{|\mathscr{L}(n,R)|} \sum_{\mathcal{C}\in\mathscr{L}(n,R)} \Gamma(\mathcal{C}). \tag{28}$$

This probability tends to 1 for large n, according to Theorem 7. The set $\mathscr{L}(n,R)$ is split into $\mathscr{L}_\varepsilon$ and $\overline{\mathscr{L}_\varepsilon}$ such that

$$\mathscr{L}_\varepsilon = \{\mathcal{C} \in \mathscr{L}(n,R): \quad \Gamma(\mathcal{C}) > 1-\varepsilon\}, \tag{29}$$

$$\overline{\mathscr{L}_\varepsilon} = \{\mathcal{C} \in \mathscr{L}(n,R): \quad \Gamma(\mathcal{C}) \leq 1-\varepsilon\}. \tag{30}$$

These subsets are now employed to bound P_n:

$$\begin{aligned} P_n &= \frac{1}{|\mathscr{L}(n,R)|}\left(\sum_{\mathcal{C}\in\mathscr{L}_\varepsilon}\Gamma(\mathcal{C}) + \sum_{\mathcal{C}\in\overline{\mathscr{L}_\varepsilon}}\Gamma(\mathcal{C})\right) \\ &\leq \frac{1}{|\mathscr{L}(n,R)|}\left(\sum_{\mathcal{C}\in\mathscr{L}_\varepsilon} 1 + \sum_{\mathcal{C}\in\overline{\mathscr{L}_\varepsilon}}(1-\varepsilon)\right) \\ &= 1 - \varepsilon\frac{|\overline{\mathscr{L}_\varepsilon}|}{|\mathscr{L}(n,R)|} \end{aligned} \tag{31}$$

or

$$\frac{|\overline{\mathscr{L}_\varepsilon}|}{|\mathscr{L}(n,R)|} \leq \frac{1}{\varepsilon}(1-P_n). \tag{32}$$

As n grows to infinity, Theorem 7 states that P_n tends to 1 for any $R < 1/2$. Hence, the right-hand side of the inequality tends to 0 for any given positive constant ε, and so does the left-hand side, which measures the proportion of $\mathscr{L}(n,R)$ that does *not* satisfy $\Gamma(\mathcal{C}) > 1-\varepsilon$. ❑

If we constrain our interest to *systematic* codes only, properties similar to Theorems 7 and 8 can be derived for such a set of codes, too. A useful method to modify the theory was given in [25].

Theorem 8 complements Corollary 6, and together they characterize the curves of Figures 2 and 3. However, as mentioned above, there exist indeed exceptions to the rule of low-rate codes having a high neighbor ratio. This will be demonstrated through an example.

Consider a code $\mathcal{C}$ that is the direct sum of two identical codes $\mathcal{C}'$, that is, the generator matrix $\mathbf{G}$ of $\mathcal{C}$ is formed as [28], [21, p. 76]

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}' & \mathbf{0} \\ \mathbf{0} & \mathbf{G}' \end{bmatrix} \tag{33}$$

where $\mathbf{G}'$ refers to the code $\mathcal{C}'$. The parameters of $\mathcal{C}$ satisfy $n(\mathcal{C}) = 2n(\mathcal{C}')$, $k(\mathcal{C}) = 2k(\mathcal{C}')$, $d(\mathcal{C}) = d(\mathcal{C}')$, and $R(\mathcal{C}) = R(\mathcal{C}')$. The 0-neighbors in $\mathcal{C}$ are given by the C rule, provided that the 0-neighbors in $\mathcal{C}'$ are known. They are

$$\mathscr{N}_0(\mathcal{C}) = \bigcup_{\mathbf{c}\in\mathscr{N}_0(\mathcal{C}')} (\{\mathbf{c} \quad \mathbf{0}\} \cup \{\mathbf{0} \quad \mathbf{c}\}) \tag{34}$$

which tells us that there are only twice as many 0-neighbors in $\mathcal{C}$ as in $\mathcal{C}'$. Hence,

$$M(\mathcal{C})\Gamma(\mathcal{C}) = 2M(\mathcal{C}')\Gamma(\mathcal{C}') \tag{35}$$

or

$$\begin{aligned} \Gamma(\mathcal{C}) &= \frac{2^{k(\mathcal{C}')+1}}{2^{k(\mathcal{C})}}\Gamma(\mathcal{C}') \\ &\leq 2^{1-n(\mathcal{C})R(\mathcal{C})/2}. \end{aligned} \tag{36}$$

The neighbor ratio can now be studied for a sequence of codes $\mathcal{C}$ such that $n(\mathcal{C})$ approaches infinity and $R(\mathcal{C})$ approaches a constant R. The resulting asymptotic neighbor ratio

$$\Gamma_\infty(R) = 0 \quad \text{if } R > 0 \tag{37}$$

contradicts the hypothesis above of $R = 1/2$ being the threshold for all types of binary linear block codes.

The small neighbor ratio of codes of the type (33) can be explained in the light of property (v) in Section III. In a code that contains no right angles, all codewords are neighbors of each other, whereas there are few neighbors in a code with many right angles. There are very many right angles in the code generated by (33), since half of the rows of $\mathbf{G}$ are orthogonal to the other half.

There are cyclic codes as well that contradict the hypothesis. Suppose, for example, that $g'(x)$ is a generator polynomial for a code $\mathcal{C}'$, and consider the code $\mathcal{C}''$ with length $n(\mathcal{C}'') = 2n(\mathcal{C}')$ generated by $g''(x) = (g'(x))^2 = g'(x^2)$. This code is identical to the direct sum code $\mathcal{C}$ given by (33), except for a reordering of the bits. Other aspects of binary cyclic codes with even lengths are discussed in [29] and [30].

The purpose of this example is to point out the possibility to design codes with $R(\mathcal{C}) < 1/2$ for which the Voronoi region (4) is less complex. However, the considered code is not a very good one. Its minimum weight is equal to the minimum weight of a half as long code. Hence, since $d(\mathcal{C}')/n(\mathcal{C}') \leq 1/2$ (see, e.g., [31, p. 167]), $d(\mathcal{C})/n(\mathcal{C}) \leq 1/4$. In fact, a minimum weight as low as this is a necessary requirement for incompressible codes with few neighbors. Any class of incompressible codes with higher minimum distance

follows (14), having a threshold at $R = 1/2$, according to the following theorem.

Theorem 9: $\Gamma_\infty(R) = 1$ for any class of incompressible binary linear block codes $\mathcal{C}$ that satisfies $d(\mathcal{C})/n(\mathcal{C}) > 1/4$.

The proof of this theorem is similar to the proof of Theorem 4. The set $\mathcal{C} \setminus \mathcal{N}_0(\mathcal{C})$ is bounded using the left-hand side of Theorem 1 in combination with Theorem 3, for incompressible codes. The proof is completed by letting $n(\mathcal{C})$ approach infinity, as in Corollary 6. We omit the details.

In a sense, Theorem 9 is the complement of Corollary 6, being derived from the other half of Theorem 1. However, note that Corollary 6 is not constrained to incompressible codes; this constraint was removed through Theorem 5. Somewhat surprisingly, the same generalization is not possible for Theorem 9. As a counterexample, consider a code $\mathcal{C}$ that is formed as the direct sum of a $\left[2^a - 1, a, 2^{a-1}\right]$ dual Hamming code and a $\left[2^{a-1}, 1, 2^{a-1}\right]$ repetition code. The created code, which is compressible, satisfies $d(\mathcal{C})/n(\mathcal{C}) > 1/3$ and $\Gamma(\mathcal{C}) = 1/2$ for any a.

Theorem 9 might appear to hold for codes with any rate $R(\mathcal{C})$, thus contradicting Corollary 6. However, a constraint on $R(\mathcal{C})$ for long codes is implicit in the condition $d(\mathcal{C})/n(\mathcal{C}) > 1/4$. The McEliece-Rodemich-Rumsey-Welch bound [32], which is the best known upper bound on the rate of long binary codes with a given ratio $d(\mathcal{C})/n(\mathcal{C})$ [33, p. 81], assures that if $d(\mathcal{C})/n(\mathcal{C}) > 1/4$, then $R(\mathcal{C}) < 0.354$ as $n(\mathcal{C}) \to \infty$.

VIII. SUMMARY

The Voronoi regions of binary linear block codes have been studied, with special emphasis on their number of facets. The neighbor ratio, which is the normalized number of facets of a Voronoi region, was investigated as a function of the rate. For almost all long codes with $R < 1/2$, the neighbor ratio is close to 1. This was showed by averaging the number of facets over the set of all binary linear block codes of the same length and rate. A counterexample was also given to demonstrate that there indeed exists some long low-rate codes with a low neighbor ratio, even though such codes are generally bad in terms of the minimum distance.

For rate values that satisfy $R > 1/2$, however, the neighbor ratio is close to 0 for *all* sufficiently long codes. In contrast to the low-rate case, this property holds without any exceptions, as proved through Theorem 5.

In summary, the asymptotic neighbor ratio as a function of the rate is a step function for almost all classes of codes, with a transition from 1 to 0 at a rate equal to one half. This theoretical results was well predicted by the computation of the Voronoi neighbors in a number of well-known codes. It was observed that the longer the code, the closer the asymptotic neighbor ratio adheres to the step function.

REFERENCES

[1] S. G. Wilson, *Digital Modulation and Coding*. Upper Saddle River, NJ: Prentice-Hall, 1996.

[2] G. C. Clark, Jr. and J. B. Cain, *Error-Correction Coding for Digital Communications*. New York, NY: Plenum Press, 1981.

[3] R. E. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley Publishing Company, 1983.

[4] 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. IT-32, no. 1, pp. 41–50, Jan. 1986.

[5] M. P. C. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Trans. Inform. Theory*, vol. 41, no. 5, pp. 1379–1396, Sept. 1995.

[6] A. Okabe, B. Boots, and K. Sugihara, *Spatial Tessellations. Concepts and Applications of Voronoi Diagrams*. Chichester, England, U.K.: John Wiley & Sons, 1992.

[7] C. E. Shannon, "Probability of error for optimal codes in a Gaussian channel," *Bell Syst. Tech. J.*, vol. 38, no. 3, pp. 611–656, May 1959.

[8] D. Slepian, "Permutation modulation," *Proc. IEEE*, vol. 53, no. 3, pp. 228–236, Mar. 1965.

[9] D. Slepian, "Group codes for the Gaussian channel," *Bell Syst. Tech. J.*, vol. 47, no. 4, pp. 575–602, Apr. 1968.

[10] G. D. Forney, Jr., "Geometrically uniform codes," *IEEE Trans. Inform. Theory*, vol. 37, no. 5, pp. 1241–1260, Sept. 1991.

[11] I. Ingemarsson, "Group codes for the Gaussian channel," in *Topics in Coding Theory. In Honour of Lars H. Zetterberg*, M. Thoma and A. Wyner, Eds. Berlin, Germany: Springer-Verlag, pp. 73–108, 1989.

[12] E. Agrell, "Voronoi regions for binary linear block codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 1, pp. 310–316, Jan. 1996.

[13] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*. Berlin, Germany: Springer-Verlag, 1987.

[14] E. Viterbo and E. Biglieri, "Computing the Voronoi cell of a lattice: the diamond-cutting algorithm," *IEEE Trans. Inform. Theory*, vol. 42, no. 1, pp. 161–171, Jan. 1996.

[15] E. R. Berlekamp, *Algebraic Coding Theory*. New York, NY: McGraw-Hill, 1968.

[16] H. J. Landau, "How does a porcupine separate its quills?," *IEEE Trans. Inform. Theory*, vol. IT-17, no. 2, pp. 157–161, Mar. 1971.

[17] T.-Y. Hwang, "Decoding linear block codes for minimizing word error rate," *IEEE Trans. Inform. Theory*, vol. IT-25, no. 6, pp. 733–737, Nov. 1979.

[18] P. Butovitsch, "Classification of signal sets. The classification capability of multi-layer perceptrons and soft-decision decoding of error-correcting codes," Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden, 1994.

[19] P. J. Green and R. Sibson, "Computing Dirichlet tessellations in the plane," *The Computer J.*, vol. 21, no. 2, pp. 168–173, May 1978.

[20] E. Agrell and P. Hedelin, "How to evaluate search methods for vector quantization," in *Proc. Nordic Signal Processing Symposium*, pp. 258–263, Ålesund, Norway, June 1994.

[21] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, the Netherlands: North-Holland Publishing Company, 1977.

[22] G. D. Forney, Jr., "Dimension/length profiles and trellis complexity of linear block codes," *IEEE Trans. Inform. Theory*, vol. 40, no. 6, pp. 1741–1752, Nov. 1994.

[23] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, nos. 3 and 4, pp. 379–423 and 623–656, July and Oct. 1948.

[24] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: John Wiley & Sons, 1991.

[25] J. N. Pierce, "Limit distribution of the minimum distance of random linear codes," *IEEE Trans. Inform. Theory*, vol. IT-13, no. 4, pp. 595–599, Oct. 1967.

[26] W. W. Peterson, *Error-Correcting Codes*. M.I.T. Press and John Wiley & Sons, 1961.

[27] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the I.R.E.*, vol. 37, no. 1, pp. 10–21, Jan. 1949.

[28] D. Slepian, "Some further theory of group codes," *Bell Syst. Tech. J.*, vol. 39, no. 5, pp. 1219–1252, Sept. 1960.

[29] J. H. van Lint, "Repeated-root cyclic codes," *IEEE Trans. Inform. Theory*, vol. 37, no. 2, pp. 343–345, Mar. 1991.

[30] G. Castagnoli, "On repeated-root cyclic codes," *IEEE Trans. Inform. Theory*, vol. 37, no. 2, pp. 337–342, Mar. 1991.

[31] R. G. Gallager, *Information Theory and Reliable Communication*. New York, NY: John Wiley and Sons, 1968.

[32] R. J. McEliece, E. R. Rodemich, H. Rumsey, Jr., and L. R. Welch, "New upper bounds on the rate of a code via the Delsarte-MacWilliams inequalities," *IEEE Trans. Inform. Theory*, vol. IT-23, no. 2, pp. 157–166, Mar. 1977.

[33] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Englewood Cliffs, NJ: Prentice-Hall, 1995.