On the Stopping Distance and the Stopping Redundancy of Codes

Moshe Schwartz, *Member, IEEE,* and Alexander Vardy, *Fellow, IEEE*

***Abstract*—It is now well known that the performance of a linear code $\mathbb{C}$ under iterative decoding on a binary erasure channel (and other channels) is determined by the size of the smallest stopping set in the Tanner graph for $\mathbb{C}$. Several recent papers refer to this parameter as the *stopping distance* s of $\mathbb{C}$. This is somewhat of a misnomer since the size of the smallest stopping set in the Tanner graph for $\mathbb{C}$ depends on the corresponding choice of a parity-check matrix. It is easy to see that $s \leq d$, where d is the minimum Hamming distance of $\mathbb{C}$, and we show that it is always possible to choose a parity-check matrix for $\mathbb{C}$ (with sufficiently many dependent rows) such that $s = d$. We thus introduce a new parameter, the *stopping redundancy* of $\mathbb{C}$, defined as the minimum number of rows in a parity-check matrix H for $\mathbb{C}$ such that the corresponding stopping distance $s(H)$ attains its largest possible value, namely, $s(H) = d$. We then derive general bounds on the stopping redundancy of linear codes. We also examine several simple ways of constructing codes from other codes, and study the effect of these constructions on the stopping redundancy. Specifically, for the family of binary Reed–Muller codes (of all orders), we prove that their stopping redundancy is at most a constant times their conventional redundancy. We show that the stopping redundancies of the binary and ternary extended Golay codes are at most 34 and 22, respectively. Finally, we provide upper and lower bounds on the stopping redundancy of MDS codes.**

***Index Terms*—Erasure channels, Golay codes, iterative decoding, linear codes, maximum distance separable (MDS) codes, Reed–Muller codes, stopping sets.**

I. Introduction

THE recent surge of interest in the binary erasure channel (BEC) is due in large part to the fact that it is the prime example of a channel over which the performance of iterative decoding algorithms can be analyzed precisely. In particular, it was shown by Di, Proietti, Telatar, Richardson, and Urbanke [7] that the performance of a low-density parity-check code (and, in fact, any linear code) under iterative decoding on the BEC is completely determined by certain combinatorial structures called *stopping sets*. A stopping set $\mathcal{S}$ in a code $\mathbb{C}$ is a subset of the variable nodes in a Tanner graph for $\mathbb{C}$ such that all the neighbors of $\mathcal{S}$ are connected to $\mathcal{S}$ at least twice. The size s of the smallest stopping set was termed the *stopping distance* of $\mathbb{C}$ in a number of recent papers [14], [19]. The stopping distance plays an important role in understanding the performance of a code under iterative decoding over the BEC, akin to the role played by the minimum Hamming distance d for maximum-likelihood and/or algebraic decoding. Just as one would like to maximize the minimum distance d if maximum-likelihood or algebraic decoding is to be used, so one should try to maximize the stopping distance s in the case of iterative decoding.

There is, however, an important difference between the minimum distance d and the stopping distance s. While the former is a property of a code $\mathbb{C}$, the latter depends on the specific Tanner graph for $\mathbb{C}$ or, equivalently, on the specific choice of a parity-check matrix H for $\mathbb{C}$. In order to emphasize this, we will henceforth use $s(H)$ to denote the stopping distance and $d(\mathbb{C})$ to denote the minimum distance.

In algebraic coding theory, a parity-check matrix H for a linear code $\mathbb{C}$ usually has $n - \dim(\mathbb{C})$ linearly independent rows. However, in the context of iterative decoding, it has been already observed in [20], [24], and other papers that adding linearly dependent rows to H can be advantageous. Certainly, this can increase the stopping distance $s(H)$. Thus, throughout this paper, a *parity-check matrix* for $\mathbb{C}$ should be understood as any matrix H whose rows span the dual code $\mathbb{C}^{\perp}$. Then the *redundancy* $r(\mathbb{C})$ of $\mathbb{C}$ may be defined as the minimum number of rows in a parity-check matrix for $\mathbb{C}$. Analogously, we define the *stopping redundancy* $\rho(\mathbb{C})$ of $\mathbb{C}$ as the minimum number of rows in a parity-check matrix H for $\mathbb{C}$ such that $s(H) = d(\mathbb{C})$. This work may be thought of as the first investigation of the tradeoff between the parameters $\rho(\mathbb{C}), r(\mathbb{C})$, and $d(\mathbb{C})$.

In the next section, we first show that the stopping redundancy $\rho(\mathbb{C})$ is well defined. That is, given any linear code $\mathbb{C}$, it is always possible to find a parity-check matrix H for $\mathbb{C}$ such that $s(H) = d(\mathbb{C})$. In fact, the parity-check matrix consisting of *all* the nonzero codewords of the dual code $\mathbb{C}^{\perp}$ has this property. Hence, $\rho(\mathbb{C}) \leq 2^{r(\mathbb{C})} - 1$ for all binary linear codes. We then show in Section II that if $d(\mathbb{C}) \leq 3$, then *any* parity-check matrix H for $\mathbb{C}$ satisfies $s(H) = d(\mathbb{C})$, so $\rho(\mathbb{C}) = r(\mathbb{C})$ in this case. The main result of Section II is an extension of this simple observation to a *general upper bound* on the stopping redundancy of binary linear codes (Theorem 4). We also derive in Section II a general *lower bound* on the stopping redundancy of linear codes (Theorem 5).

In Section III, we study several simple ways of constructing codes from other codes, such as the direct-sum construction and code extension by adding an overall parity check. We investigate

Manuscript received March 17, 2005; revised December 9, 2005. This work was supported in part by the National Science Foundation and in part by the David and Lucile Packard Foundation. The material in this paper was presented in part at the IEEE International Symposium on Information Theory, Adelaide, Australia, September 2005.

M. Schwartz was with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA. He is now with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: moosh@paradise.caltech.edu).

A. Vardy is with the Department of Electrical and Computer Engineering, the Department of Computer Science and Engineering, and the Department of Mathematics, the University of California San Diego, La Jolla, CA 92093-0407 USA (e-mail: vardy@kilimanjaro.ucsd.edu).

Communicated by G. Zémor, Associate Editor for Coding Theory.
Digital Object Identifier 10.1109/TIT.2005.864441

the effect of these constructions on the stopping redundancy. It should be pointed out that although we have focused our discussion on binary codes in Sections II and III, most of the results therein extend straightforwardly to linear codes over an arbitrary finite field.

We continue in Section IV with an in-depth analysis of the well-known $(u, u+v)$ construction, and in particular its application in the recursive definition [17, p. 374] of binary Reed–Muller codes. By slightly modifying this construction, we establish a strong upper bound on the stopping redundancy of Reed–Muller codes of arbitrary orders. Specifically, we prove that if $\mathbb{C}$ is a Reed–Muller code of length 2^m and order r, then $\rho(\mathbb{C}) \le d(\mathbb{C})r(\mathbb{C})/2$. Thus, for any constant $d(\mathbb{C})$, we have an increase in redundancy by only a constant factor.

In Section V, we study the $(24, 12, 8)$ extended binary Golay code $\mathcal{G}_{24}$ and the $(12, 6, 6)$ extended ternary Golay code $\mathcal{G}_{12}$. We prove that $\rho(\mathcal{G}_{24}) \le 34$ and $\rho(\mathcal{G}_{12}) \le 22$ by providing specific parity-check matrices for these codes. We take $\mathcal{G}_{24}$ as a test case, and compare the performance of three different decoders: a maximum-likelihood decoder, an iterative decoder using the conventional 12×24 double-circulant parity-check matrix of [17, p. 65] and an iterative decoder using the 34×24 parity-check matrix with maximum stopping distance. In each case, exact analytic expressions for the probability of decoding failure are derived using a computer program.

In Section VI, we consider MDS codes over $\mathbb{F}_q$ with $q \ge 2$. It is easy to extend the general bounds of Section II to q-ary codes. However, in Section VI, we establish much better upper and lower bounds on the stopping redundancy of MDS codes. Notably, all these bounds are independent of the field size q.

This paper only scratches the surface of the many interesting and important questions that arise in the investigation of the stopping redundancy. We conclude in Section VII with a brief discussion and a list of open problems.

II. General Bounds

We begin with rigorous definitions of the stopping distance and the stopping redundancy. Let $\mathbb{C}$ be a binary linear code and let $H = [h_{i,j}]$ be a parity-check matrix for $\mathbb{C}$. The corresponding Tanner graph T for $\mathbb{C}$ is a bipartite graph with each column of H represented by a *variable node* and each row of H represented by a *check node* in such a way that the jth variable node is connected to the ith check node if and only if $h_{i,j} \neq 0$. As already mentioned, a stopping set in T is a subset $\mathcal{S}$ of the variable nodes such that all the check nodes that are neighbors of a node in $\mathcal{S}$ are connected to *at least two nodes* in $\mathcal{S}$. We dispense with this graphical representation of stopping sets in favor of an equivalent definition directly in terms of the underlying parity-check matrix H. Thus, we say that a *stopping set* is a set of columns of H with the property that the projection of H onto these columns does not contain a row of weight one.[1] The resulting definition of the stopping distance—the smallest size of a stopping set—bears a striking resemblance to the definition of the minimum Hamming distance of a linear code.

Recall that the minimum distance of a linear code $\mathbb{C}$ can be defined as the largest integer $d(\mathbb{C})$ such that every $d(\mathbb{C}) - 1$ or less columns of H are linearly independent. For binary codes, this is equivalent to saying that $d(\mathbb{C})$ is the largest integer such that every set of $d(\mathbb{C}) - 1$ or less columns of H contains at least one row of odd weight.

Definition 1: Let $\mathbb{C}$ be a linear code (not necessarily binary) and let H be a parity-check matrix for $\mathbb{C}$. Then the ***stopping distance*** of H is defined as the the largest integer $s(H)$ such that every set of $s(H) - 1$ or less columns of H contains at least one row of weight one.

The following corollary is an immediate consequence of juxtaposing the definitions of $s(H)$ and $d(\mathbb{C})$ above.

Corollary 1: Let $\mathbb{C}$ be a linear code and let H be an arbitrary parity-check matrix for $\mathbb{C}$. Then $s(H) \le d(\mathbb{C})$.

Indeed, it is well known [7], [9], [14] that the support of every codeword is a stopping set, which is another way to see that $s(H) \le d(\mathbb{C})$ regardless of the choice of H. Thus, given a linear code $\mathbb{C}$, the largest stopping distance one could hope for is $d(\mathbb{C})$, no matter how cleverly the Tanner graph for $\mathbb{C}$ is constructed. The point is that this bound can be *always* achieved by adding dependent rows to H (see Theorem 2). This makes the notion of the stopping distance, as a property of a code $\mathbb{C}$, somewhat meaningless: without restricting the number of rows in a parity-check matrix for $\mathbb{C}$, we cannot distinguish between the stopping distance and the conventional minimum distance. This observation, in turn, leads to the following definition.

Definition 2: Let $\mathbb{C}$ be a linear code with minimum Hamming distance $d(\mathbb{C})$. Then the ***stopping redundancy*** of $\mathbb{C}$ is defined as the the smallest integer $\rho(\mathbb{C})$ such that there exists a parity-check matrix H for $\mathbb{C}$ with $\rho(\mathbb{C})$ rows and $s(H) = d(\mathbb{C})$.

The following theorem shows that the stopping redundancy is, indeed, well defined.

Theorem 2: Let $\mathbb{C}$ be a linear code, and let H^* denote the parity-check matrix for $\mathbb{C}$ consisting of all the nonzero codewords of the dual code $\mathbb{C}^\perp$. Then $s(H^*) = d(\mathbb{C})$.

Proof: Let $[\mathbb{C}^\perp]$ denote the $|\mathbb{C}^\perp| \times n$ matrix consisting of all the codewords of $\mathbb{C}^\perp$. It is well known (cf. [17, p. 139]) that $[\mathbb{C}^\perp]$ is an orthogonal array of strength $d(\mathbb{C}) - 1$. This means that any set of $t \le d(\mathbb{C}) - 1$ columns of $[\mathbb{C}^\perp]$ contains all the vectors of length t among its rows, each vector appearing the same number of times. In particular, any set of $d(\mathbb{C}) - 1$ or less columns of $[\mathbb{C}^\perp]$ contains all the vectors of weight one among its rows. Clearly, the all-zero row can be removed from $[\mathbb{C}^\perp]$ to obtain H^*, while preserving this property. □

Theorem 2 also provides a trivial upper bound on the stopping redundancy. In particular, it follows from Theorem 2 that $\rho(\mathbb{C}) \le 2^{r(\mathbb{C})} - 1$ for any binary linear code $\mathbb{C}$. This bound holds with equality in the degenerate case of the single parity-check code. The next theorem determines $\rho(\mathbb{C})$ exactly for *all* binary linear codes with minimum distance $d(\mathbb{C}) \le 3$.

[1] This explains why stopping sets *stop* the progress of an iterative decoder. A row of weight one—equivalently, a check node of degree one—would determine unambiguously an erased symbol. However, if an entire stopping set is erased, then all the neighboring check nodes are connected to these erasures at least twice, and thus form an underconstrained system of linear equations. In this case, an iterative decoder has no way of determining the erased values.

Theorem 3: Let $\mathbb{C}$ be a binary linear code with minimum distance $d(\mathbb{C}) \le 3$. Then ***any*** parity-check matrix H for $\mathbb{C}$ satisfies $s(H) = d(\mathbb{C})$, and therefore $\rho(\mathbb{C}) = r(\mathbb{C})$.

Proof: If H contains an all-zero column, then it is obvious that $s(H) = d(\mathbb{C}) = 1$. Otherwise $s(H) \ge 2$, since then every single column of H must contain a row of weight one. Now, if $d(\mathbb{C}) = 3$, then every two columns of H are distinct. This implies that these two columns must contain either the 01 row or the 10 row (or both). Hence, $s(H) = 3$. □

The following theorem, which is our main result in this section, shows that Theorem 3 is, in fact, a special case of a general upper bound on the stopping redundancy of linear codes.

Theorem 4: Let $\mathbb{C}$ be a binary linear code with minimum distance $d(\mathbb{C}) \ge 3$. Then

$$\rho(\mathbb{C}) \le \binom{r(\mathbb{C})}{1} + \binom{r(\mathbb{C})}{2} + \cdots + \binom{r(\mathbb{C})}{d(\mathbb{C})-2}. \qquad (1)$$

Proof: We first prove a slightly weaker result, which is conceptually simpler. Namely, let us show that

$$\rho(\mathbb{C}) \le \binom{r(\mathbb{C})}{1} + \binom{r(\mathbb{C})}{2} + \cdots + \binom{r(\mathbb{C})}{d(\mathbb{C})-1}. \qquad (2)$$

Let H be an arbitrary parity-check matrix for $\mathbb{C}$ with $r(\mathbb{C})$ linearly independent rows. Construct another parity-check matrix H' whose rows are all the linear combinations of t rows of H, for all $t = 1, 2, \ldots, d(\mathbb{C}) - 1$. Clearly, the number of rows of H' is given by the right-hand side of (2). Now let H_t, respectively H'_t, denote a matrix consisting of some t columns of H, respectively the corresponding t columns of H'. Observe that for all $t \le d(\mathbb{C}) - 1$, the t columns of H_t are linearly independent. This implies that the row-rank of H_t is t, and therefore, some t rows of H_t must form a basis for $\mathbb{F}_2^t$. Hence, the $2^t - 1$ nonzero linear combinations of these t rows of H_t generate all the nonzero vectors in $\mathbb{F}_2^t$, including all the vectors of weight one. But for $t \le d(\mathbb{C}) - 1$, the $2^t - 1$ nonzero linear combinations of *any* t rows of H_t are among the rows of H'_t by construction. This proves that $s(H') = d(\mathbb{C})$ and establishes (2).

To transition from (2) to (1), observe that we do not need to have all the nonzero vectors of $\mathbb{F}_2^t$ among the rows of H'_t; it would suffice to have at least one vector of weight one. Given a set $\mathcal{S} \subseteq \mathbb{F}_2^t$ and a positive integer m, let $m\mathcal{S}$ denote the set of all vectors obtained as a linear combination of at most m vectors from $\mathcal{S}$. Define $\mu(t)$ as the smallest integer with the property that for any basis B of $\mathbb{F}_2^t$, the set $\mu(t)B$ contains at least one vector of weight one. Then in the construction of H', it would suffice to take all the linear combinations of at most $\mu(d(\mathbb{C})-1)$ rows of H. Clearly, $\mu(t) \le t - 1$ for all t (in fact, $\mu(t) = t - 1$ for all t), and the theorem follows. □

The bound of (1), while much better than $\rho(\mathbb{C}) \le 2^{r(\mathbb{C})} - 1$, is still too general to be tight for most codes. Nevertheless, we can conclude from Theorem 4 that when $d(\mathbb{C})$ is a constant, the stopping redundancy is only polynomial in the (conventional) redundancy and, hence, in the length of the code.

In the next theorem, we provide a general *lower bound* on the stopping redundancy of linear codes.

Theorem 5: Let $\mathbb{C}$ be an arbitrary linear code of length n. For each $i = 1, 2, \ldots, d(\mathbb{C}) - 1$, define

$$w_i \overset{\text{def}}{=} \max\left\{ \left\lceil \frac{n+1}{i} \right\rceil - 1,\, d(\mathbb{C}^\perp) \right\}. \qquad (3)$$

Then

$$\rho(\mathbb{C}) \ge \binom{n}{i} \Big/ w_i \binom{n - w_i}{i-1}, \qquad \text{for } i = 1, 2, \ldots, d(\mathbb{C}) - 1.$$

Proof: Let H be a parity-check matrix for $\mathbb{C}$ and let $\mathcal{I}$ be an arbitrary set of i column indices. We say that $\mathcal{I}$ is an *i-set.* We also say that a row $\boldsymbol{h}$ of H *covers* $\mathcal{I}$ if the projection of $\boldsymbol{h}$ onto $\mathcal{I}$ has weight one. If $s(H) = d(\mathbb{C})$, then each of the $\binom{n}{i}$ i-sets must be covered by at least one row of the parity-check matrix, for all $i = 1, 2, \ldots, d(\mathbb{C}) - 1$. Any single row of H of weight $w \le n - i + 1$ covers exactly

$$f_n(i, w) \overset{\text{def}}{=} w \binom{n-w}{i-1} \qquad (4)$$

i-sets. It is not difficult to see that the expression in (4) increases monotonically as w decreases until $f_n(i, w)$ reaches its maximum at $w = \lceil (n+1)/i \rceil - 1$. But $\mathrm{wt}(\boldsymbol{h}) \ge d(\mathbb{C}^\perp)$ for all rows $\boldsymbol{h}$ of H. Thus, each row of H covers at most $f_n(i, w_i)$ i-sets, where w_i is defined in (3), and the theorem follows. □

Is there an asymptotically good sequence of linear codes $\mathbb{C}_1, \mathbb{C}_2, \ldots$ such that the stopping redundancy $\rho(\mathbb{C}_i)$ grows only polynomially fast with the length? The answer to this question is unknown at the present time. However, if the dual sequence $\mathbb{C}_1^\perp, \mathbb{C}_2^\perp, \ldots$ is also asymptotically good, we can use Theorem 5 to settle this question in the negative.

Corollary 6: Let $\mathbb{C}_1, \mathbb{C}_2, \ldots$ be an infinite sequence of linear codes of strictly increasing length n_i and fixed rate $k_i/n_i = R$, with $0<R<1$, such that $d(\mathbb{C}_i)/n_i \ge \delta_1$ for all i, with $\delta_1 > 0$. If also $d(\mathbb{C}_i^\perp)/n_i \ge \delta_2$ for all i, with $\delta_2 > 0$, then

$$\rho(\mathbb{C}_i) \;=\; \Omega\left(n^{-1} 2^{n\left[H_2(\delta_1) - (1-\delta_2) H_2\left(\frac{\delta_1}{1-\delta_2}\right)\right]} \right)$$

where $n = n_i$ and $H_2(x) = x \log_2 x^{-1} + (1-x)\log_2 (1-x)^{-1}$ is the binary entropy function.

Proof: We apply the bound of Theorem 5 with the size of an i-set given by $d(\mathbb{C})-1$. It is easy to see that if $d(\mathbb{C}_i)/n_i \ge \delta_1$ and $d(\mathbb{C}_i^\perp)/n_i \ge \delta_2$ for all i, then the maximum in (3) is attained at $d(\mathbb{C}_i^\perp)$ for all sufficiently large i. Thus,

$$\begin{aligned}\rho(\mathbb{C}_i) &\ge \binom{n}{d(\mathbb{C}_i) - 1} \Big/ d(\mathbb{C}_i^\perp) \binom{n - d(\mathbb{C}_i^\perp)}{d(\mathbb{C}_i) - 2} \\ &\ge \frac{2^{n f(\delta_1, \delta_2)}}{n} \cdot \frac{\delta_1}{\delta_2} \sqrt{\frac{\pi(1-\delta_1-\delta_2)}{4(1-\delta_1)(1-\delta_2)}}\end{aligned}$$

where $f(\delta_1, \delta_2) = H_2(\delta_1) - (1-\delta_2) H_2(\delta_1/(1-\delta_2))$ and the second inequality follows from well-known bounds [17, p. 309] on binomial coefficients in terms of $H_2(\,\cdot\,)$. □

We observe that the function $f(\delta_1, \delta_2)$ defined in the proof of Corollary 6 is always positive, and therefore $\rho(\mathbb{C}_i)$ indeed grows exponentially with the length n. Note that several well-known families of asymptotically good codes (for example, the self-dual codes [16]) satisfy the condition of Corollary 6.

III. Constructions of Codes From Other Codes

In this section, we examine several simple ways of constructing codes from other codes. While for most such constructions, it is trivial to determine the redundancy of the resulting code, we find it considerably more difficult to determine the resulting stopping redundancy, and resort to bounding it.

We start with two simple examples. The first example (Theorem 7) is the well-known direct-sum construction or, equivalently, the (u,v) construction. The second one (Theorem 8) is the (u,u) construction, or concatenation of a code with itself.

Theorem 7: Let $\mathbb{C}_1, \mathbb{C}_2$ be $(n_1,k_1,d_1), (n_2,k_2,d_2)$ binary linear codes, respectively. Then $\mathbb{C}_3 = \{(u,v) : u \in \mathbb{C}_1, v \in \mathbb{C}_2\}$ is an $(n_1+n_2, k_1+k_2, \min\{d_1,d_2\})$ code with

$$\rho(\mathbb{C}_3) \leq \rho(\mathbb{C}_1) + \rho(\mathbb{C}_2). \tag{5}$$

Proof: Let H_1 be an arbitrary $\rho(\mathbb{C}_1) \times n$ parity-check matrix for $\mathbb{C}_1$ with $s(H_1) = d_1$, and let H_2 be an arbitrary $\rho(\mathbb{C}_2) \times n$ parity-check matrix for $\mathbb{C}_2$ with $s(H_2) = d_2$. Then

$$H_3 = \begin{pmatrix} H_1 & \mathbf{0} \\ \mathbf{0} & H_2 \end{pmatrix}$$

is a parity-check matrix for $\mathbb{C}_3$. Assume without loss of generality (w.l.o.g.) that $d_1 \leq d_2$, so $d(\mathbb{C}_3) = d_1$. Label the columns of H_3 by $1, 2, \ldots, n_1+n_2$, and let $\mathcal{I}$ be an arbitrary set of at most $d(\mathbb{C}_3) - 1$ column indices. If $\mathcal{I} \cap \{1,2,\ldots,n_1\} \neq \varnothing$, then the fact that $s(H_1) = d(\mathbb{C}_3)$ implies that there is a row of weight one in the projection of H_3 onto $\mathcal{I}$. Otherwise, $\mathcal{I} \subset \{n_1{+}1, n_2{+}2, \ldots, n_1+n_2\}$, and the same conclusion follows from $s(H_2) = d_2 \geq d(\mathbb{C}_3)$. □

Theorem 8: Let $\mathbb{C}_1$ be an (n,k,d) binary linear code. Then the code $\mathbb{C}_2 = \{(u,u) : u \in \mathbb{C}_1\}$ is a $(2n,k,2d)$ code with

$$\rho(\mathbb{C}_2) \leq \rho(\mathbb{C}_1) + n. \tag{6}$$

Proof: Let H_1 be a $\rho(\mathbb{C}_1) \times n$ parity-check matrix for $\mathbb{C}_1$ with $s(H_1) = d$. Construct a parity-check matrix for $\mathbb{C}_2$ as

$$H_2 = \begin{pmatrix} H_1 & \mathbf{0} \\ I_n & I_n \end{pmatrix}$$

where I_n is the $n \times n$ identity matrix. Label the columns of H_2 by $0, 1, \ldots, 2n-1$, and assume to the contrary there exists a set $\mathcal{I} \subset \{0,1,\ldots,2n-1\}$ such that $|\mathcal{I}| \leq 2d-1$ and there is no row of weight one in the projection of H_2 onto $\mathcal{I}$. Let $H_2(\mathcal{I})$ denote this projection. First, note that the two identity matrices in H_2 imply that if $j \in \mathcal{I}$, then also $(j{+}n) \bmod 2n$ is in $\mathcal{I}$, since, otherwise, $H_2(\mathcal{I})$ contains a row of weight one. It follows that $\mathcal{I} \cap \{0,1,\ldots,n{-}1\} \neq \varnothing$. But $s(H_1) = d$, so the top part of H_2 implies that $|\mathcal{I} \cap \{0,1,\ldots,n{-}1\}| \geq d$, otherwise, $H_2(\mathcal{I})$ again contains a row of weight one. By the first observation, we now conclude that $|\mathcal{I}| \geq 2d$, a contradiction. □

Here is an interesting observation about Theorems 7 and 8. It follows from (5) and (6) that if the constituent codes are optimal, in the sense that their stopping redundancy is equal to their redundancy, then the resulting code is also optimal. This indicates that the bounds in (5) and (6) are tight.

In contrast, the innocuous construction of *extending* a linear code $\mathbb{C}$ by adding an overall parity check [17, p. 27] appears to be much more difficult to handle. The next theorem deals only with the special case where $d(\mathbb{C}) = 3$.

Theorem 9: Let $\mathbb{C}$ be an $(n,k,3)$ binary linear code. Then the extended code $\mathbb{C}'$ is an $(n+1,k,4)$ code with

$$\rho(\mathbb{C}') \leq 2\rho(\mathbb{C}) = 2r(\mathbb{C}') - 2.$$

Proof: Let H be an arbitrary $r(\mathbb{C}) \times n$ parity-check matrix for $\mathbb{C}$. We construct a parity-check matrix for $\mathbb{C}'$ as follows:

$$H' = \begin{pmatrix} H & \mathbf{0} \\ \bar{H} & \mathbf{1} \end{pmatrix}$$

where $\bar{H}$ is the bitwise complement of H, while $\mathbf{0}$ and $\mathbf{1}$ are the all-zero and the all-one column vectors, respectively. Label the columns in H' by $1, 2, \ldots, n+1$, and let $\mathcal{I}$ be a subset of $\{1,2,\ldots,n+1\}$ with $|\mathcal{I}| \leq 3$. In fact, it would suffice to consider the case where $\mathcal{I} \subset \{1,2,\ldots,n\}$ and $|\mathcal{I}| = 3$; all other cases easily follow from the fact that $s(H) = 3$ by Theorem 3.

Let $H(\mathcal{I})$ and $\bar{H}(\mathcal{I})$ denote the projections of H and $\bar{H}$, respectively, on the three positions in $\mathcal{I}$. If $H(\mathcal{I})$ contains a row of weight one, we are done. If $H(\mathcal{I})$ contains a row of weight two, we are also done—then the corresponding row in $\bar{H}(\mathcal{I})$ has weight one. But otherwise, the only rows in $H(\mathcal{I})$ are 000 and 111, which means that the three columns in $H(\mathcal{I})$ are identical, a contradiction since $d(\mathbb{C}) = 3$. □

The construction in Theorem 9 is not optimal. For example, if $\mathbb{C}'$ is the $(8,4,4)$ extended Hamming code, it produces a parity-check matrix for $\mathbb{C}'$ with six rows. But $\mathbb{C}'$ is also the Reed–Muller code $\mathcal{R}(1,3)$ for which we give in the next section a parity-check matrix H with $s(H) = 4$ and only five rows.

IV. Reed–Muller Codes

We now focus on the well-known $(u, u+v)$ construction, in particular in connection with the recursive definition of binary Reed–Muller codes. Our goal is to derive a constructive upper bound on the stopping redundancy of $\mathcal{R}(r,m)$—the binary Reed–Muller code of order r and length 2^m.

We begin by recalling several well-known facts. The reader is referred to [17, Ch. 13] for a proof of all these facts. First, for all $r = 0, 1, \ldots, m$, the dimension of $\mathcal{R}(r,m)$ is $k = \sum_{i=0}^{r} \binom{m}{i}$ and its minimum distance is $d = 2^{m-r}$. Let $G(r,m)$ be a generator matrix for $\mathcal{R}(r,m)$. Then, using the $(u, u{+}v)$ construction, $G(r,m)$ can be defined recursively, as follows:

$$G(r,m) \overset{\text{def}}{=} \begin{pmatrix} G(r,m-1) & G(r,m-1) \\ \mathbf{0} & G(r{-}1,m{-}1) \end{pmatrix} \tag{7}$$

with the recursion in (7) being bootstrapped by $G(m,m) = I_{2^m}$ and $G(0,m) = (11\cdots1)$ for all m. By convention, the code $\mathcal{R}(-1,m)$ is the set $\{\mathbf{0}\}$ for all m. Then

$$\mathcal{R}(r,m)^{\perp} = \mathcal{R}(m-r-1,m) \tag{8}$$

for all m and all $r = -1, 0, 1, \dots, m$. It follows from (8) that $G(r,m)$ is a parity-check matrix for $\mathcal{R}(m-r-1,m)$, a code with minimum distance 2^{r+1}. Hence, every $2^{r+1}-1$ columns of $G(r,m)$ are linearly independent.

Our objective in what follows is to construct an alternative parity-check matrix $H(r,m)$ for $\mathcal{R}(m-r-1,m) = \mathcal{R}(r,m)^{\perp}$ such that $s(H(r,m)) = 2^{r+1}$. Then the number of rows in $H(r,m)$ gives an upper bound on the stopping redundancy of $\mathcal{R}(m-r-1,m)$ (and the number of rows in $H(m-r-1,m)$ is an upper bound on the stopping redundancy of $\mathcal{R}(r,m)$). Here is the recursive construction that we will use.

Recursive Construction A: For all positive integers m and for all $r = 1, 2, \dots, m-2$, we define

$$H(r,m) = \left(\begin{array}{c} H_{\text{top}} \\ \hline H_{\text{bot}} \end{array}\right) \overset{\text{def}}{=} \left(\begin{array}{cc} H(r,m-1) & H(r,m-1) \\ \mathbf{0} & H(r-1,m-1) \\ \hline H(r-1,m-1) & \mathbf{0} \end{array}\right) \tag{9}$$

with the recursion in (9) being bootstrapped as follows: for all $m = 0, 1, \dots$, the matrices $H(0,m), H(m-1,m), H(m,m)$ are defined by

$$H(0,m) \overset{\text{def}}{=} G(0,m) = (11\cdots 1) \tag{10}$$

$$H(m-1,m) \overset{\text{def}}{=} G(m-1,m) \tag{11}$$

$$H(m,m) \overset{\text{def}}{=} G(m,m) = I_{2^m} \tag{12}$$

Proposition 10: $H(r,m)$ is a generator matrix for $\mathcal{R}(r,m)$ and, hence, a parity-check matrix for $\mathcal{R}(m-r-1,m)$.

Proof: The proof is by induction on m and r. Equations (10) to (12) establish the induction base. For the induction step, we need to prove that (9) generates $\mathcal{R}(r,m)$, assuming that $H(r,m-1)$ generates $\mathcal{R}(r,m-1)$ and $H(r-1,m-1)$ generates $\mathcal{R}(r-1,m-1)$. It follows immediately from (7) that H_{top} already generates $\mathcal{R}(r,m)$. Thus, it would suffice to show that all the rows of H_{bot} belong to $\mathcal{R}(r,m)$. To this end, we write

$$\mathcal{R}(r,m) = \{(u, u+v) : u \in \mathbb{C}_1, v \in \mathbb{C}_2\} \tag{13}$$

where $\mathbb{C}_1 = \mathcal{R}(r,m-1)$ and $\mathbb{C}_2 = \mathcal{R}(r-1,m-1)$. Observe that each row of H_{bot} can be written as

$$(v,\mathbf{0}) = (v,v) + (\mathbf{0},v)$$

where $v \in \mathbb{C}_2$. The fact that $(\mathbf{0},v) \in \mathcal{R}(r,m)$ follows immediately from (13) for $u = \mathbf{0}$. The fact that $(v,v) \in \mathcal{R}(r,m)$ also follows from (13) in conjunction with the well-known fact that $\mathbb{C}_2 \subset \mathbb{C}_1$ (take $u := v$ and $v := \mathbf{0}$). Hence, all the rows of H_{bot} belong to $\mathcal{R}(r,m)$, and the induction step is complete. □

It remains to show that the stopping distance of $H(r,m)$ is indeed 2^{r+1}. We again prove this by induction on m and r. Let us first establish the induction base. Trivially, the stopping distance of $H(0,m)$ is 2, since $H(0,m) = (11\cdots 1)$ by (12).

Lemma 11: The stopping distance of $H(m-1,m)$ is 2^m.

Proof: The proof is by induction on m. Start with m=1, in which case we have $s(H(0,1)) = 2$, as desired. For the induction step, observe that

$$H(m-1,m) = \begin{pmatrix} I_{2^{m-1}} & I_{2^{m-1}} \\ \mathbf{0} & H(m-2,m-1) \end{pmatrix}.$$

The situation here is exactly the same as the one we had in the proof of Theorem 8, and the result follows in the same manner. As in Theorem 8, assume to the contrary that there exists a set $\mathcal{I} \subset \{0,1,\dots,2^m-1\}$ such that $|\mathcal{I}| \le 2^m - 1$ and there is no row of weight one in the projection of $H(m-1,m)$ on $\mathcal{I}$. Then $j \in \mathcal{I}$ implies that $(j + 2^{m-1}) \bmod 2^m$ is in $\mathcal{I}$. Hence, $\mathcal{I} \cap \{2^{m-1},\dots,2^m-1\} \ne \varnothing$. But the stopping distance of $H(m-2,m-1)$ is 2^{m-1} by induction hypothesis, which implies that $\{2^{m-1},\dots,2^m-1\} \subseteq \mathcal{I}$. By the earlier observation, this means that $\mathcal{I} = \{0,1,\dots,2^m-1\}$, a contradiction. □

Proposition 12: The stopping distance of $H(r,m)$ is 2^{r+1} for all positive integers m and for all $r = 0,1,\dots,m-1$,

Proof: The proof is by induction on m and r. Lemma 11 in conjunction with the fact that the stopping distance of $H(0,m)$ is 2 establish the induction base. For the induction step, assume that $\mathcal{I} \subseteq \{0,1,\dots,2^m-1\}$ is a set of column indices such that $|\mathcal{I}| \le 2^{r+1}-1$. We distinguish between three easy cases.

Case 1: $\mathcal{I} \cap \{0,1,\dots,2^{m-1}-1\} = \varnothing$.
Then $\mathcal{I} \subseteq \{2^{m-1}, 2^{m-1}+1, \dots, 2^m-1\}$. By induction hypothesis, the stopping distance of $H(r,m-1)$ is 2^{r+1}. Hence, the top row in (9) implies that the projection of $H(r,m)$ onto $\mathcal{I}$ contains a row of weight one.

Case 2: $1 \le |\mathcal{I} \cap \{0,1,\dots,2^{m-1}-1\}| \le 2^r - 1$.
By induction hypothesis, $H(r-1,m-1)$ has a stopping distance of 2^r. Hence, the bottom row in (9) implies that the projection of $H(r,m)$ onto $\mathcal{I}$ contains a row of weight one.

Case 3: $|\mathcal{I} \cap \{0,1,\dots,2^{m-1}-1\}| \ge 2^r$.
Then $|\mathcal{I} \cap \{2^{m-1}, 2^{m-1}+1, \dots, 2^m-1\}| \le 2^r - 1$, and we are in a case that is symmetric to either Case 2 or Case 1. □

The remaining task is to compute the number of rows in the matrix $H(r,m)$. We denote this number as $g(r,m)$.

Lemma 13: For all $r = 0,1,\dots,m-1$, the number of rows in $H(r,m)$ is given by

$$g(r,m) = \sum_{i=0}^{r} \binom{m-r-1+i}{i} 2^i.$$

Proof: Consider the following generating function:

$$f(x,y) = \sum_{m=0}^{\infty} \sum_{r=0}^{m} g(r,m+1) x^r y^m.$$

Note that $H(m,m+1) = G(m,m+1)$ for all $m \ge 0$, in view of (11). Hence, $g(m,m+1) = 2^{m+1}-1$. Using the recursion $g(r,m+1) = g(r,m) + 2g(r-1,m)$, which follows immediately from (9), along with this initial condition, we obtain

$$f(x,y) \;=\; yf(x,y) + 2xyf(x,y) + \sum_{i=0}^{\infty} x^i y^i. \tag{14}$$

Upon rearranging, (14) becomes

$$f(x,y) = \frac{1}{1-y(1+2x)} \sum_{i=0}^{\infty} x^i y^i$$
$$= \left(\sum_{i=0}^{\infty} x^i y^i \right) \left(\sum_{i=0}^{\infty} y^i \sum_{j=0}^{i} \binom{i}{j} 2^j x^j \right). \quad (15)$$

The lemma now follows by observing that $g(r,m)$ is the coefficient of $x^r y^{m-1}$ in (15). □

We are now in a position to summarize the results of this section in the following theorem.

Theorem 14: For all $m = 1, 2, \ldots$ and all $r = 0, 1, \ldots, m$, the stopping redundancy of $\mathcal{R}(r,m)$ is upper-bounded by

$$\rho(\mathcal{R}(r,m)) \leq \sum_{i=0}^{m-r-1} \binom{r+i}{i} 2^i. \quad (16)$$

Proof: Follows immediately from (8), Proposition 10, Proposition 12, and Lemma 13. □

To see how far the bound of Theorem 14 is from the (conventional) redundancy of Reed–Muller codes, we first need the following technical lemma.

Lemma 15: For all positive integers m and $0 \leq r \leq m-1$, we have

$$\sum_{i=0}^{r} \binom{m-r-1+i}{i} 2^{r-i} = \sum_{i=0}^{r} \binom{m}{i}. \quad (17)$$

Proof: Denote the sum $\sum_{i=0}^{r} \binom{m}{i}$ by $S(m,r)$. Using the $\binom{m}{i} = \binom{m-1}{i-1} + \binom{m-1}{i}$ recursion, we obtain

$$S(m,r) = \binom{m-1}{r} + 2 \sum_{i=0}^{r-1} \binom{m-1}{i}$$

and recognize the second term above as $2S(m-1, r-1)$. The result now follows by induction on m and r. □

Using Lemma 15, we can establish a relation between the redundancy of Reed–Muller codes and their stopping redundancy. For this, it will be more convenient to work with the dual code $\mathbb{C} = \mathcal{R}(r,m)^{\perp}$. Recall that $r(\mathbb{C}) = \sum_{i=0}^{r} \binom{m}{i}$. Comparing this to the bound on $\rho(\mathbb{C})$ in Theorem 14, we find that

$$\rho(\mathbb{C}) \leq \sum_{i=0}^{r} \binom{m-r-1+i}{i} 2^i \leq 2^r \sum_{i=0}^{r} \binom{m}{i} = 2^r r(\mathbb{C})$$

where the second inequality follows from (17). Therefore, for any fixed order r, the stopping redundancy of $\mathcal{R}(r,m)^{\perp}$ is at most the redundancy of $\mathcal{R}(r,m)^{\perp}$ times a constant. Alternatively, if we take $\mathbb{C} = \mathcal{R}(r,m)$, then Theorem 14 implies that $\rho(\mathbb{C}) \leq d(\mathbb{C}) r(\mathbb{C})/2$. Thus, for any fixed $d(\mathbb{C})$, the increase in redundancy is by a constant factor.

V. Golay Codes

The $(24,12,8)$ binary Golay code $\mathcal{G}_{24}$ is arguably the most remarkable binary block code. It is often used as a benchmark in studies of code structure and decoding algorithms.

There are several "canonical" parity-check matrices for $\mathcal{G}_{24}$, see [3], [4], [23] and other papers. Our starting point is the systematic double-circulant matrix H_{24} given in MacWilliams and Sloane [17, p. 65] and shown in Table I. It can be readily verified that $s(H_{24}) = 4$, which means that H_{24} achieves only half of the maximum possible stopping distance. Curiously, the stopping distance of the two "trellis-oriented" parity-check matrices for $\mathcal{G}_{24}$, given in [23, p. 2060] and [3, p. 1441], is also 4.

TABLE I
Two Parity-Check Matrices for the $(24,12,8)$ Golay Code $\mathcal{G}_{24}$

$$H_{24} = \begin{pmatrix}
1&1&0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&1&1&0&0&0&1&0\\
1&0&1&0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&1&1&0&0&0&1\\
1&0&0&1&0&0&0&0&0&0&0&0&0&1&0&1&1&0&1&1&1&0&0&0\\
1&0&0&0&1&0&0&0&0&0&0&0&0&0&1&0&1&1&0&1&1&1&0&0\\
1&0&0&0&0&1&0&0&0&0&0&0&0&0&0&1&0&1&1&0&1&1&1&0\\
1&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&1&0&1&1&0&1&1&1\\
1&0&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&1&0&1&1&0&1&1\\
1&0&0&0&0&0&0&0&1&0&0&0&0&1&1&0&0&0&1&0&1&1&0&1\\
1&0&0&0&0&0&0&0&0&1&0&0&0&1&1&1&0&0&0&1&0&1&1&0\\
1&0&0&0&0&0&0&0&0&0&1&0&0&0&1&1&1&0&0&0&1&0&1&1\\
1&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1&1&1&0&0&0&1&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&1&1&1
\end{pmatrix}$$

$$H'_{24} = \begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&1&1&0&1&1&0&0&1&0&0&1&1&1&0\\
0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&1&1&0&1&1&0&0&0&1\\
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&1&0&0&1&1&1&0&1&1\\
0&0&0&0&0&0&0&1&1&0&0&0&1&1&0&1&1&0&0&0&1&0&0&1\\
0&0&0&0&0&0&1&0&0&1&1&1&1&1&1&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&1&1&0&1&0&1&1&0&0&1&1&1&0&0&0&0&0&0\\
0&0&0&0&1&0&1&1&0&1&0&0&0&0&0&1&0&0&1&0&0&1&1&0\\
0&0&0&0&1&1&1&0&1&1&0&1&1&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&1&1&1&1&0&0&0&0&1&0&0&0&1&0&1&0&0&0&0&1\\
0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&1&1&1&1&0&0&0&1&1\\
0&0&0&1&1&0&0&0&0&0&0&0&0&1&1&1&0&1&1&0&0&1&0&0\\
0&0&0&1&1&0&1&1&1&0&0&1&0&0&0&0&0&1&1&0&0&0&0&0\\
0&0&0&1&1&1&0&0&0&0&0&1&1&1&0&0&0&0&1&1&0&0&0&0\\
0&0&1&0&0&0&1&1&0&1&0&0&0&0&0&0&1&0&0&0&1&0&1&1\\
0&0&1&0&1&1&1&1&1&0&0&0&0&0&1&0&0&0&0&0&0&0&1&0\\
0&0&1&1&1&0&0&1&0&0&1&1&0&1&1&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&1&0&1&0&1&0&1&0&0&1&0&1&1&0\\
0&1&0&0&0&0&0&0&1&0&1&1&0&1&1&0&1&0&0&0&0&0&0&1\\
0&1&0&0&0&1&0&0&1&0&1&0&0&0&1&0&0&0&1&0&1&0&1&0\\
0&1&0&0&0&1&1&1&1&0&0&1&0&0&0&0&1&0&0&0&1&0&0&0\\
0&1&1&0&0&1&0&0&1&0&0&0&0&0&1&0&1&1&0&1&0&0&0&0\\
0&1&1&1&0&0&0&1&0&0&0&0&0&1&0&0&0&1&1&1&0&0&0&0\\
1&0&0&0&0&0&1&0&0&0&1&1&1&0&0&1&0&0&0&0&0&1&1&0\\
1&0&0&0&0&1&0&0&0&0&1&1&0&1&1&1&0&0&1&0&0&0&0&0\\
1&0&0&1&1&0&0&0&1&0&0&0&0&0&0&1&0&1&0&0&1&0&0&1\\
1&0&1&0&0&0&0&0&0&1&1&0&1&0&0&0&0&0&0&1&0&0&1&1\\
1&0&1&1&0&0&0&0&0&0&0&1&0&0&1&1&0&0&0&0&1&1&0&0\\
1&0&1&1&0&1&0&0&0&0&0&0&1&0&0&0&0&1&0&1&1&0&0&0\\
1&0&1&1&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&1&0&1&0&1\\
1&1&0&0&0&1&1&0&0&0&0&0&1&0&0&0&1&1&0&0&0&1&0&0\\
1&1&0&0&1&0&1&0&0&1&1&0&0&0&0&0&0&0&0&1&0&1&0&0\\
1&1&0&1&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&1&1&0&0\\
1&1&1&0&1&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0\\
1&1&1&1&0&0&0&0&1&1&0&0&0&0&0&1&0&0&0&1&0&0&0&0
\end{pmatrix}$$

Computing the general bounds of Theorems 4 and 5 for the special case of $\mathcal{G}_{24}$ produces the extremely weak result

$$6 \leq \rho(\mathcal{G}_{24}) \leq 2509.$$

Having tried several methods to construct a parity-check matrix for $\mathcal{G}_{24}$ with stopping distance 8, our best result was achieved using a greedy (lexicographic) computer search. Specifically, with the 4095 nonzero vectors of $\mathcal{G}_{24}$ listed lexicographically, we iteratively construct the parity-check matrix H'_{24}, at each iteration adjoining to H'_{24} the first vector on the list with the highest score. Each vector receives i points to its score for each yet uncovered i-set it covers, where $i \in \{1, 2, \ldots, 7\}$ (cf. Theorem 5). The resulting matrix is given in Table I. Since H'_{24} has only 34 rows and $s(H'_{24}) = 8$, it follows that the stopping redundancy of $\mathcal{G}_{24}$ is at most 34.

To evaluate the effect of increasing the stopping distance, it would be interesting to compare the performance of iterative

TABLE II
NUMBER OF UNDECODABLE ERASURE PATTERNS BY WEIGHT w IN THREE DECODERS FOR $\mathcal{G}_{24}$

w	Total Patterns	$\Psi_{\rm ML}(w)$	$\Psi_{H_{24}}(w)$	$\Psi_{H'_{24}}(w)$
0	1	0	0	0
1	24	0	0	0
2	276	0	0	0
3	2024	0	0	0
4	10626	0	110	0
5	42504	0	2277	0
6	134596	0	19723	0
7	346104	0	100397	0
8	735471	759	343035	3598
9	1307504	12144	844459	82138
10	1961256	91080	1568875	585157
11	2496144	425040	2274130	1717082
12	2704156	1313116	2637506	2556402
$\geqslant 13$	$\binom{24}{w}$	$\binom{24}{w}$	$\binom{24}{w}$	$\binom{24}{w}$

decoders for $\mathcal{G}_{24}$ based on H_{24} or H'_{24}, respectively. As a baseline for such a comparison, it would be also useful to have the performance of a maximum-likelihood decoder for $\mathcal{G}_{24}$. In what follows, we give analytic expressions for the performance of the three decoders on the BEC.

Clearly, a maximum-likelihood decoder fails to decode (recover) a given erasure pattern if and only if this pattern contains the support of (at least one) nonzero codeword of $\mathcal{G}_{24}$. Let $\Psi_{\rm ML}$ denote the number of such erasure patterns as a function of their weight w. Then

$$\Pr_{\rm ML}\{\text{decoding failure}\} = \sum_{w=0}^{24} \Psi_{\rm ML}(w) p^w (1-p)^{24-w}$$

where p is the erasure probability of the BEC. In contrast, an iterative decoder (based on H_{24} or H'_{24}) fails if and only if the erasure pattern contains a stopping set. Thus,

$$\Pr_{H_{24}}\{\text{decoding failure}\} = \sum_{w=0}^{24} \Psi_{H_{24}}(w) p^w (1-p)^{24-w}$$

$$\Pr_{H'_{24}}\{\text{decoding failure}\} = \sum_{w=0}^{24} \Psi_{H'_{24}}(w) p^w (1-p)^{24-w}$$

where $\Psi_{H_{24}}(w)$ and $\Psi_{H'_{24}}(w)$ denote the number of erasure patterns of weight w that contain a stopping set of H_{24} and H'_{24}, respectively. It remains to compute $\Psi_{H_{24}}$, $\Psi_{H'_{24}}$, and $\Psi_{\rm ML}$.

Obviously, $\Psi_{\rm ML}(w) = 0$ for $w \le 7$ and $\Psi_{\rm ML}(w) = \binom{24}{w}$ for $w \ge 13$ (any 13 columns of a parity-check matrix for $\mathcal{G}_{24}$ are linearly dependent). For the other values of w, we have

$$\Psi_{\rm ML}(w) = \begin{cases} \binom{16}{w-8} 759, & 8 \le w \le 11 \\ 1771(20+720)+2576, & w = 12 \end{cases}$$

where we made use Table IV of [5] (for $w = 12$, we have $\Psi_{\rm ML}(w) = |X_{12}| + |S_{12}| + |U_{12}|$ in the notation of [5]). To find $\Psi_{H_{24}}(\cdot)$ and $\Psi_{H'_{24}}(\cdot)$, we used exhaustive computer search. These functions are given in Table II. The resulting probabilities of decoding failure are plotted in Fig. 1. Note that while we may add rows to H'_{24} to eliminate more stopping sets, this would have negligible effect since the slope of the performance curve is dominated by the smallest w for which $\Psi_{H'_{24}}(w) \neq 0$.

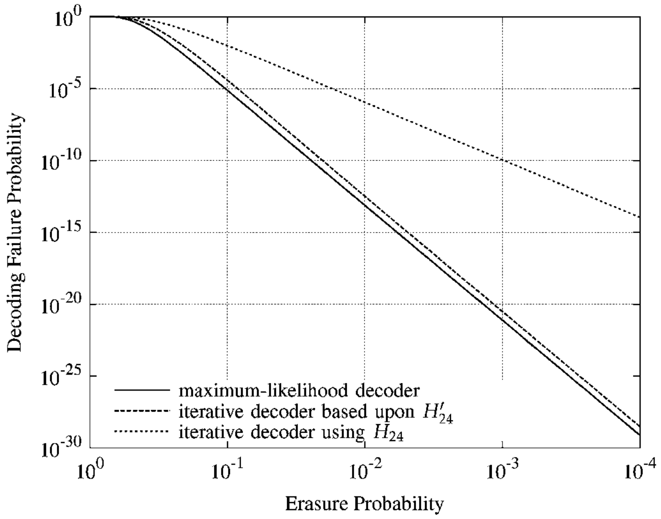

Fig. 1. The decoding failure probability of three decoders for $\mathcal{G}_{24}$: a maximum-likelihood decoder and iterative decoders based upon H_{24} and H'_{24}.

TABLE III
TWO PARITY-CHECK MATRICES FOR THE $(12, 6, 6)$ GOLAY CODE $\mathcal{G}_{12}$

$$H_{12} = \begin{pmatrix} 1&0&0&0&0&0&0&1&1&1&1&1 \\ 0&1&0&0&0&0&1&0&1&-&-&1 \\ 0&0&1&0&0&0&1&1&0&1&-&- \\ 0&0&0&1&0&0&1&-&1&0&1&- \\ 0&0&0&0&1&0&1&-&-&1&0&1 \\ 0&0&0&0&0&1&1&1&-&-&1&0 \end{pmatrix}$$

$$H'_{12} = \begin{pmatrix} 0&0&0&0&0&1&1&1&-&-&1&0 \\ 1&1&-&-&1&0&0&0&0&0&0&- \\ 0&0&0&-&1&0&0&0&1&1&-&- \\ -&-&1&0&0&1&1&1&0&0&0&0 \\ 0&0&-&0&0&1&0&0&-&1&-&1 \\ 1&-&0&1&1&0&1&-&0&0&0&0 \\ 1&1&0&0&0&0&1&1&-&0&0&- \\ 0&1&0&1&0&1&0&0&1&1&1&0 \\ 1&0&-&0&1&0&0&-&0&1&-&0 \\ 0&0&-&1&1&0&1&0&0&0&-&1 \\ 1&0&0&1&1&1&0&0&0&1&0&1 \\ 0&0&1&0&1&1&0&1&1&1&0&0 \\ -&1&0&0&0&0&1&-&0&1&1&0 \\ 0&-&-&1&0&0&-&1&0&0&0&- \\ 0&0&0&0&1&1&-&0&1&0&1&1 \\ -&0&1&1&0&1&0&0&-&-&0&0 \\ 0&-&1&0&-&1&0&0&-&0&1&0 \\ -&0&0&-&1&0&0&-&0&0&1&1 \\ 0&-&0&-&1&0&-&0&0&-&0&1 \\ 1&0&-&0&1&1&1&0&-&0&0&0 \\ 1&0&1&1&1&0&0&0&1&0&1&0 \\ 0&1&0&-&-&1&0&0&0&0&-&1 \end{pmatrix}$$

The $(12, 6, 6)$ extended ternary Golay code $\mathcal{G}_{12}$ is another famous code. A systematic double-circulant parity-check matrix for $\mathcal{G}_{12}$ is given in [17, p. 510]; this matrix is denoted H_{12} in Table III. It is easy to see that $s(H_{12}) = 3$, which is again half of the maximum possible stopping distance. Using greedy lexicographic search, we have constructed a parity-check matrix H'_{12} with stopping distance 6 and only 22 rows. This matrix is also given in Table III. The number of undecodable erasure patterns for a maximum-likelihood decoder and for the iterative decoders based on H_{12} and H'_{12} is given in Table IV.

TABLE IV
NUMBER OF UNDECODABLE ERASURE PATTERNS BY WEIGHT w IN THREE DECODERS FOR $\mathcal{G}_{12}$

w	Total Patterns	$\Psi_{\rm ML}(w)$	$\Psi_{H_{12}}(w)$	$\Psi_{H'_{12}}(w)$
0	1	0	0	0
1	12	0	0	0
2	66	0	0	0
3	220	0	20	0
4	495	0	150	0
5	792	0	456	0
6	924	132	758	377
$\geqslant 7$	$\binom{12}{w}$	$\binom{12}{w}$	$\binom{12}{w}$	$\binom{12}{w}$

VI. MDS CODES

The last family of codes we investigate are the MDS codes. These codes have intricate algebraic and combinatorial structure [17, Ch. 11]. In particular, if $\mathbb{C}$ is an (n,k,d) linear[2] MDS code, then the dual code $\mathbb{C}^\perp$ is also MDS and its distance is $d^\perp = k+1 = n-d+2$. Moreover, every d positions in $\{1,2,\ldots,n\}$ are the support of a codeword of $\mathbb{C}$ of weight d, while every $d^\perp$ positions support a codeword of $\mathbb{C}^\perp$ of weight $d^\perp$. We will use these and other properties of MDS codes to establish sharp upper and lower bounds on their stopping redundancy.

Theorem 16: Let $\mathbb{C}$ be an (n,k,d) MDS code with $d \geq 2$. Then

$$\frac{1}{d-1}\binom{n}{d-2} \leq \rho(\mathbb{C}) \leq \binom{n}{d-2}. \tag{18}$$

Proof: The lower bound is just a special case of Theorem 5. Taking $i = d-1$ in (3), we find that

$$w_{d-1} = d(\mathbb{C}^\perp) = n-d+2$$

whenever $d \geq 2$, so that $n - w_{d-1} = d-2$. The corresponding lower bound in Theorem 5 thus reduces to

$$\rho(\mathbb{C}) \geq \frac{\binom{n}{d-1}}{(n-d+2)\binom{d-2}{d-2}} = \frac{1}{d-1}\binom{n}{d-2}. \tag{19}$$

To prove the upper bound, note that every $d^\perp = n-d+2$ positions support a codeword of $\mathbb{C}^\perp$. We take one such codeword of $\mathbb{C}^\perp$ for every set of $d^\perp$ positions, and use the resulting

$$\binom{n}{d^\perp} = \binom{n}{n-d+2} = \binom{n}{d-2}$$

codewords as rows of a matrix H. We claim that H is a parity-check matrix for $\mathbb{C}$, namely, that $\operatorname{rank}(H) = n-k = d-1$. Indeed, consider a set of $d-1$ positions, say $\{1,2,\ldots,d{-}1\}$. For each $i \in \{1,2,\ldots,d-1\}$, there is a row of H of weight $d^\perp = n-(d-1)+1$ such that the intersection of its support with $\{1,2,\ldots,d-1\}$ is $\{i\}$. The corresponding $d-1$ rows of H thus contain an identity matrix on the first $d-1$ positions; hence, $\operatorname{rank}(H) = d{-}1$. It remains to show that $s(H) = d$. But this follows immediately from what we have already proved: given any set $\mathcal{I} \subset \{1,2,\ldots,n\}$ with $|\mathcal{I}| = d-1$, there is a corresponding set of $d-1$ rows of H whose projection on the positions in $\mathcal{I}$ is the identity matrix. □

Both bounds in Theorem 16 are exact if $d = 2$. Indeed, for $d = 2$ the upper and lower bounds in (18) coincide, yielding $\rho(\mathbb{C}) = 1$. This reflects the degenerate case of the $(n, n-1, 2)$ MDS code $\mathbb{C}$, whose dual is the $(n,1,n)$ repetition code $\mathbb{C}'$. Indeed, any codeword of $\mathbb{C}'$ can serve as a $1 \times n$ parity-check matrix H for $\mathbb{C}$ with $s(H) = 2$. In the case of the $(n,1,n)$ repetition code $\mathbb{C}'$ itself, the bounds in (18) reduce to

$$\frac{n}{2} \leq \rho(\mathbb{C}') \leq \frac{n(n-1)}{2}.$$

The true value is $\rho(\mathbb{C}') = r(\mathbb{C}') = n-1$. To see this, consider an $(n-1) \times n$ parity-check matrix H' for $\mathbb{C}'$ such that the support of the ith row in H' is $\{i, i{+}1\}$ for $i = 1,2,\ldots,n-1$.

Next, we use a combinatorial argument to show that $d = 2$ is the *only case* where the lower bound of Theorem 18 is exact.

Theorem 17: Let $\mathbb{C}$ be an (n,k,d) MDS code with $d \geq 3$. Then

$$\rho(\mathbb{C}) \geq \left\lfloor \frac{1}{d-1}\binom{n}{d-2} \right\rfloor + 1. \tag{20}$$

Proof: Assume to the contrary that there is a parity-check matrix H for $\mathbb{C}$ with $s(H) = d$ and at most $\binom{n}{d-2}/(d-1)$ rows. As in Theorem 5, we say that a given set $\mathcal{I} \subseteq \{1,2,\ldots,n\}$ with $|\mathcal{I}| = i$ is an *i-set*, and that a row $\boldsymbol{h}$ of H *covers* an i-set $\mathcal{I}$ if the projection of $\boldsymbol{h}$ on $\mathcal{I}$ has weight one. The number of $(d-1)$-sets covered by a single row of weight $w \geq d^\perp$ is

$$D_{n,d}(w) = \begin{cases} w\binom{n-w}{d-2}, & w = d^\perp = n-d+2 \\ 0, & w > d^\perp = n-d+2. \end{cases} \tag{21}$$

The total number of $(d-1)$-sets is $\binom{n}{d-1}$ and every one of them must be covered by at least one row of H. But

$$\frac{\binom{n}{d-1}}{\max\limits_{w \geq d^\perp} D_{n,d}(w)} = \frac{\binom{n}{d-1}}{d^\perp\binom{d-2}{d-2}} = \frac{1}{d-1}\binom{n}{d-2} \tag{22}$$

in view of (21). It now follows from (22) that there are exactly $\binom{n}{d-2}/(d-1)$ rows in H, all of weight $w = d^\perp$, and that each $(d-1)$-set is covered by *exactly one* row of H. The latter condition is equivalent to saying that each (complementary) set of $n-(d-1) = d^\perp - 1$ positions is contained in the support of exactly one row of H. In other words, the supports of the rows of H form an $\mathcal{S}(d^\perp{-}1, d^\perp, n)$ Steiner system.[3] Such a Steiner system may or may not exist. If it does not exist we are done, but in many known cases (e.g., $\mathcal{S}(2,3,7), \mathcal{S}(3,4,8), \mathcal{S}(4,5,11)$, etc.) it does; hence, we must proceed to establish another contradiction. To this end, consider a $(d-2)$-set which is the complement of the support of a given row $\boldsymbol{h}_1$ of H. As $s(H) = d$, this

[2] Throughout this section, we deal with linear MDS codes only. Henceforth, whenever we say "an MDS code" we mean a linear MDS code.

[3] An $\mathcal{S}(t,k,v)$ Steiner system is a set of k-subsets of $\{1,2,\ldots,v\}$, called *blocks*, so that each t-subset of $\{1,2,\ldots,v\}$ is contained in exactly one block.

$(d-2)$-set must be covered by some other row of H, say $\boldsymbol{h}_2$. But then

$$|\mathrm{supp}(\boldsymbol{h}_1) \cap \mathrm{supp}(\boldsymbol{h}_2)| = d^{\perp} - 1.$$

The above means that there is a set of $d^{\perp}-1$ positions that is contained in two different blocks of the $\mathcal{S}(d^{\perp}-1, d^{\perp}, n)$ Steiner system, a contradiction. □

Example: The hexacode $\mathcal{H}_6$ is a remarkable $(6,3,4)$ MDS code over $\mathbb{F}_4 = \{0, 1, \omega, \bar{\omega}\}$. It is unique up to monomial equivalence and self-dual under the Hermitian inner product (so the conjugate of a parity-check matrix for $\mathcal{H}_6$ is a generator matrix for $\mathcal{H}_6$). The upper and lower bounds in (18) imply that $5 \le \rho(\mathcal{H}_6) \le 15$. Using one of the covering designs (see below) in [10], we construct the following parity-check matrix:

$$H = \begin{pmatrix} \bar{\omega} & \omega & 0 & 1 & 0 & 1 \\ \bar{\omega} & \omega & 1 & 0 & 1 & 0 \\ 0 & 1 & \bar{\omega} & \omega & 0 & 1 \\ 1 & 0 & \bar{\omega} & \omega & 1 & 0 \\ 0 & 1 & 0 & 1 & \bar{\omega} & \omega \\ 1 & 0 & 1 & 0 & \bar{\omega} & \omega \end{pmatrix} \tag{23}$$

for $\mathcal{H}_6$. It can be easily verified by hand that $s(H) = 4$, and therefore $\rho(\mathcal{H}_6) \le 6$. Finally, the lower bound of Theorem 17 proves that $\rho(\mathcal{H}_6) = 6$. Thus, (20) is exact in this case. □

In general, it follows from the Proof of Theorem 17 that if $\mathbb{C}$ is an (n,k,d) MDS code and H is a parity-check matrix for $\mathbb{C}$ with $s(H) = d$, then the supports of rows of weight $d^{\perp}$ in H form a $(n, d^{\perp}, d^{\perp}-1)$ covering design. A (v,k,t) covering design is collection of subsets of size k of $\{1, 2, \dots, v\}$, called blocks, such that every subset of $\{1, 2, \dots, v\}$ of size t is contained in at least one block (changing "at least one" to "exactly one" thus makes this a Steiner system). The smallest number of blocks in a (v,k,t) covering design is usually denoted by $C(v,k,t)$ and called the *covering number* (see [11], [18], and references therein). Thus, if $\mathbb{C}$ is an (n,k,d) MDS code, then

$$\rho(\mathbb{C}) \ge C(n, d^{\perp}, d^{\perp}-1) = C(n, k{+}1, k). \tag{24}$$

The best general lower bound on the covering number dates back to the work of Schönheim [21], who showed in 1964 that $C(v,k,t) \ge (v/k)C(v-1, k-1, t-1)$. For the special case of (24), this proves that

$$\rho(\mathbb{C}) \ge \left\lceil \frac{n}{k{+}1} \left\lceil \frac{n{-}1}{k} \left\lceil \frac{n{-}2}{k{-}1} \cdots \left\lceil \frac{n{-}k{+}1}{2} \cdots \right\rceil \right\rceil \right\rceil \right\rceil. \tag{25}$$

Notice that if we ignore all the ceilings in (25), then we recover precisely the lower bound in (18). Hence (25) is always at least as strong as the lower bound of Theorem 16. An alternative bound on the covering number is due to de Caen [6] (see also [10, p. 270]). In our case, this bound reduces to

$$\rho(\mathbb{C}) \ \ge \ \frac{k+1}{(k+2)(d-2)} \binom{n}{d-2}. \tag{26}$$

This is better than the lower bound of Theorem 16 if and only if $n > 2(k{+}1) = 2d^{\perp}$. Note that Theorem 17 is sometimes stronger than both (25) and (26), for example, in those cases where $n \le 2d^{\perp}$ and an $\mathcal{S}(k, k{+}1, n)$ Steiner system exists.

We can now summarize most of the results in this section as follows. If $\mathbb{C}$ is an (n,k,d) MDS code over $\mathbb{F}_q$ with $d \ge 3$, the the stopping redundancy of $\mathbb{C}$ is in the range

$$\frac{1}{d-1}\binom{n}{d-2} < \rho(\mathbb{C}) \le \frac{\max\{d^{\perp}, d-1\}}{n}\binom{n}{d-2}$$

(see the Appendix for a proof of the upper bound). These bounds are reasonably close and, notably, do not depend on the size of the field. Determining the stopping redundancy of MDS codes *exactly* appears to be a difficult combinatorial problem. In view of (24), it is likely to be at least as difficult as the problem of determining the covering number $C(n, k{+}1, k)$.

VII. Discussion and Open Problems

This paper only scratches the surface of the many interesting and important problems that arise in the investigation of stopping redundancy. The importance of stopping sets is well understood in the case of the BEC. However, the concept of stopping redundancy is new. Fig. 1 clearly demonstrates that it is the stopping sets of size strictly less than the minimum distance that are responsible for the performance gap between maximum-likelihood and iterative decoding. Thus, eliminating such stopping sets is what we need to do, and the stopping redundancy is the relevant figure of merit.

It would be extremely interesting to understand how relevant stopping redundancy is for other channels. In this regard, it is worth mentioning the following observation of Feldman [9, p. 176]. In the general framework of LP decoding, the support of *any pseudocodeword* is a stopping set for *any channel*. Thus, the stopping redundancy might be the relevant figure of merit in this, very general, context as well.

It is interesting to note that although we have defined and studied the stopping redundancy as a property of linear codes, it turns out to be closely related to a number of well-known combinatorial structures. Steiner systems and covering designs were already discussed in Section VI. A combinatorial structure equivalent to a covering design is the Turán system. For more information on this, we refer the reader to [13], [18], [22]. Another combinatorial concept that is *very* closely related to stopping redundancy is that of k-locally-thin families. A family $\mathcal{F}$ of subsets of the set $\{1, 2, \dots, \rho\}$ is said to be *k-locally-thin* if given any k distinct subsets in $\mathcal{F}$, there is at least one element $i \in \{1, 2, \dots, \rho\}$ that is contained in exactly one of them. The central problem in the study of k-locally-thin families is to determine $M(\rho, k)$, defined as the maximum cardinality of a k-locally-thin family of subsets of $\{1, 2, \dots, \rho\}$. In particular, one would like to determine the sequence

$$t(k) \overset{\text{def}}{=} \limsup_{\rho \to \infty} \frac{\log_2 M(\rho, k)}{\rho}. \tag{27}$$

But $M(\rho, k)$ is also the maximum number of columns in a binary matrix H with ρ rows, distinct columns, and no stopping set of size k. Hence, results on stopping redundancy might be relevant in the study of locally-thin families, and *vice versa*. For example, our construction in Section IV produces a parity-check matrix for the Reed–Muller code $\mathcal{R}(m-2,m)$ of length n, distance 4, and stopping redundancy $(2+o(1))\log_2 n$, thereby showing that $t(3)\geq 1/2$. We point out that estimating $t(k)$ is a notoriously difficult task. In fact, it is not even known whether $t(3)<1$ and whether $t(k)$ decreases monotonically with k. For much more on this, see [1], [2], [15], and references therein.

We have concluded the original version of this paper with a variety of research questions related to our results. Although some of these questions have been since answered (see below), we repeat them here. In Section II, we derived upper and lower bounds on the stopping redundancy of general binary linear codes. Can these general bounds be improved? In particular, is there an asymptotically good family of codes such that their stopping redundancy grows only polynomially fast with their length? In Section III, we have examined only a small sample of the multitude of known ways of combining codes to construct other codes. What can be said of the stopping redundancy of other constructions, in particular constructions involving nonbinary alphabets, such as concatenated/multilevel coding? In Sections IV and V, we investigated the Reed–Muller codes and the Golay codes. Are the constructions provided therein optimal? In particular, is it true that $\rho(\mathcal{G}_{24})=34$? It appears that proving lower bounds on the stopping redundancy, even for specific codes such as $\mathcal{G}_{24}$, is quite difficult. Finally, in Section VI, we considered MDS codes. We conjecture that the stopping redundancy of an (n,k,d) MDS code $\mathbb{C}$ over $\mathbb{F}_q$ *does not depend on the code*, but only on its parameters n and k. In other words, any two (n,k,d) MDS codes have the same stopping redundancy. If this conjecture is true, then it should be possible, in principle, to determine the stopping redundancy of an (n,k,d) MDS code as a function of n and k. However, this appears to be a difficult combinatorial problem.

Finally, we would like to mention two recent papers that are directly inspired by our results, and improve upon them. Etzion [8] studies in detail the stopping redundancy of binary Reed–Muller codes. He proves that the stopping redundancy of $\mathcal{R}(m-2,m)$, which is also the exteded Hamming code of length 2^m, is $2m-1$. This shows that our construction in Section IV is optimal in this case. However, it turns out that this construction is *not* optimal for the first-order Reed–Muller codes $\mathcal{R}(1,m)$; Etzion [8] derives a better upper bound on the stopping redundancy of these codes. Han and Siegel [13] use the "probabilistic method" to establish upper bounds on the stopping redundancy of general linear codes, which improve significantly upon our result in Theorem 4. They also prove upper bounds on the stopping redundancy of MDS codes in terms of Turán numbers, that are stronger than our Corollary 20.

Appendix
An Improved Upper Bound on the Stopping Redundancy of MDS Codes

In this appendix, we improve the upper bound in Theorem 16 using constant-weight codes. An $(n,4,w)$ *constant-weight code* $\mathcal{C}$ is a set of binary vectors of length n and weight w, such that any two elements of $\mathcal{C}$ are at distance ≥ 4 from each other. Let $U(n,w,m)$ denote the largest possible cardinality of a union of m constant-weight codes, each with parameters $(n,4,w)$.

Theoren 18: Let $\mathbb{C}$ be an (n,k,d) MDS code with $d\geq 3$. Set $m=\min\{k,n-k-1\}$. Then

$$\rho(\mathbb{C})\leq\binom{n}{d-2}-U(n,d-2,m). \tag{28}$$

Proof: We start as in the Proof of Theorem 16 by constructing a parity-check matrix H for $\mathbb{C}$ such that the supports[4] of the rows of H are all the binary vectors of length n and weight $d^{\perp}=n-d+2$. Now let $\mathcal{C}_1,\mathcal{C}_2,\ldots,\mathcal{C}_m$ be any m constant-weight codes with parameters $(n,4,n-d+2)$. We remove from H all the rows whose supports belong to $\mathcal{C}_1\cup\mathcal{C}_2\cup\cdots\cup\mathcal{C}_m$. Let H' denote the resulting matrix. Since obviously

$$U(n,n-d+2,m)=U(n,d-2,m)$$

the number of rows remaining in H' is given by the right-hand side of (28), provided $\mathcal{C}_1,\mathcal{C}_2,\ldots,\mathcal{C}_m$ are chosen so as to maximize the cardinality of their union. We claim that $s(H')=d$. To prove this claim, we distinguish between two cases.

Case 1: Consider a $(d-1)$-set. As shown in the Proof of Theorem 16, there are some $d-1$ rows in H such that the projection of their supports on the $(d-1)$-set is the $(d-1)\times(d-1)$ identity matrix. Let $\mathcal{D}\subset\mathbb{F}_2^n$ denote this set of $d-1$ supports. Any two elements of $\mathcal{D}$ are at distance exactly 2 from each other, since $(d-1)+(d^{\perp}-1)=n$. Hence, $|\mathcal{D}\cap\mathcal{C}_i|\leq 1$ for all i. As $m\leq n-k-1=d-2=|\mathcal{D}|-1$, it follows that H' contains at least one row whose support belongs to $\mathcal{D}$.

Case 2: Consider a t-set with $t\leq d-2$ and assume w.l.o.g. that this t-set is $\{1,2,\ldots,t\}$. Note that H contains some $d^{\perp}$ rows whose supports are

$$\{t,t+1,\ldots,t+d^{\perp}\}\setminus\{t+i\},\qquad\text{for } i=1,2,\ldots,d^{\perp}.$$

As before, let $\mathcal{D}$ denote this set of $d^{\perp}$ supports. The intersection of each support in $\mathcal{D}$ with the t-set $\{1,2,\ldots,t\}$ is $\{t\}$, so the projection of each of the corresponding $d^{\perp}$ rows of H onto this t-set has weight one. Moreover, any two elements of $\mathcal{D}$ are at Hamming distance 2 from each other. Hence $|\mathcal{D}\cap\mathcal{C}_i|\leq 1$ for all i, and since $m\leq k=d^{\perp}-1=|\mathcal{D}|-1$, it follows that H' has at least one row whose support is in $\mathcal{D}$.

It remains to show that $\operatorname{rank}(H')=n-k=d-1$. But this follows from the fact that $s(H')=d$. Indeed, up to an appropriate column permutation, there is a row in H' such that the intersection of its support with $\{1,2,\ldots,d-1\}$ is $\{d-1\}$. Then, there is another row in H' such that the intersection of its support with $\{1,2,\ldots,d-2\}$ is $\{d-2\}$, again up to a column permutation. Continuing in this manner, we get a set of $d-1$ rows of H' whose projection on the first $d-1$ positions is an upper-triangular matrix with nonzero entries on the main diagonal. Hence, $\operatorname{rank}(H')=d-1$, and we are done. □

Proposition 19: For all positive integers n and w with $w\leq n$ and for all $m\leq n$

$$U(n,w,m)\geq\frac{m}{n}\binom{n}{w}. \tag{29}$$

[4] We shall regard the support of a row of H interchangeably as a subset of $\{1,2,\ldots,n\}$ or as the corresponding binary vector of length n.

Proof: Graham and Sloane [12, Theorem 1] construct a partition of the set of binary vectors of length n and weight w into n constant-weight codes with parameters $(n, 4, w)$. Taking the m largest codes in such a partition proves (29). □

Corollary 20: Let $\mathbb{C}$ be an (n, k, d) MDS code. Then

$$\rho(\mathbb{C}) \leq \frac{\max\{d^{\perp}, d-1\}}{n}\binom{n}{d-2}. \tag{30}$$

Proof: Follows immediately from Theorem 18 and Proposition 19. Note that (30) coincides with (18) iff $d = 2$. □

Acknowledgment

We are grateful to Ilya Dumer, Tuvi Etzion, Jonathan Feldman, and Paul Siegel for helpful and stimulating discussions.

References

[1] N. Alon, E. Fachini, and J. Körner, "Locally thin set families," *Comb., Probab., Comput.*, vol. 9, pp. 481–488, Nov. 2000.

[2] N. Alon, J. Körner, and A. Monti, "String quartets in binary," *Comb, Probab., Computing*, vol. 9, pp. 381–390, Sep. 2000.

[3] A. R. Calderbank, G. D. Forney Jr., and A. Vardy, "Minimal tail-biting trellises: The Golay code and more," *IEEE Trans. Inf. Theory*, vol. 45, no. 4, pp. 1435–1455, Jul. 1999.

[4] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*. New York: Springer-Verlag, 1993.

[5] ——, "Orbit and coset analysis of the golay and related codes," *IEEE Trans. Inf. Theory*, vol. 36, no. 5, pp. 1038–1050, Sep. 1990.

[6] D. de Caen, "Extension of a theorem of Moon and Moser on complete subgraphs," *Ars Comb.*, vol. 16, pp. 5–10, 1983.

[7] C. Di, D. Proietti, İ. E. Telatar, T. J. Richardson, and R. L. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570–1579, Jun. 2002.

[8] T. Etzion, "On the stopping redundancy of Reed-Muller codes," preprint, Oct. 2005.

[9] J. Feldman, "Decoding error-correcting codes via linear programming," Ph.D. dissertation, MIT, Cambridge, MA, Sep. 2003.

[10] D. Gordon, G. Kuperberg, and O. Patashnik, "New constructions for covering designs," *J. Comb. Des.*, vol. 3, pp. 269–284, 1995.

[11] D. Gordon, G. Kuperberg, O. Patashnik, and J. Spencer, "Asymptotically optimal covering designs," *J. Comb. Theory*, vol. 75, pp. 270–280, 1996.

[12] R. L. Graham and N. J. A. Sloane, "Lower bounds for constant weight codes," *IEEE Trans. Inf. Theory*, vol. IT-26, no. 1, pp. 37–43, Jan. 1980.

[13] J. Han and P. H. Siegel, "Improved upper bounds on stopping redundancy," *IEEE Trans. Inf. Theory*, submitted for publication.

[14] N. Kashyap and A. Vardy, "Stopping sets in codes from designs," in *Proc. IEEE Int. Symp. Information Theory*, Yokohama, Japan, Jun./Jul. 2003, p. 1222.

[15] A. Kostochka, "Extremal problems on delta-systems," in *Numbers, Information and Complexity*. Boston, MA: Kluwer, 2000, pp. 143–150.

[16] F. J. MacWilliams, N. J. A. Sloane, and J. G. Thompson, "Good self-dual codes exist," *Discr. Math.*, vol. 3, pp. 153–162, 1972.

[17] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North-Holland, 1978.

[18] W. H. Mills and R. C. Mullin, "Coverings and packings," in *Contemporary Design Theory: A Collection of Surveys*, J. H. Dinitz and D. R. Stinson, Eds. New York: Wiley, 1992, pp. 371–399.

[19] A. Orlitsky, K. Viswanathan, and J. Zhang, "Stopping set distribution of LDPC code ensembles," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 929–953, Mar. 2005.

[20] N. Santhi and A. Vardy, "On the effect of parity-check weights in iterative decoding," in *Proc. IEEE Int. Symp. Information Theory*, Chicago, IL, Jun./Jul. 2004, p. 322.

[21] J. Schönheim, "On coverings," *Pacific J. Math.*, vol. 14, pp. 1405–1411, 1964.

[22] A. Sidorenko, "Upper bounds for Turán numbers," *J. Combin. Theory, Ser. A*, vol. 77, pp. 134–147, 1997.

[23] A. Vardy, "Trellis structure of codes," in *Handbook of Coding Theory*, V. Pless and W. C. Huffman, Eds. Amsterdam, The Netherlands: Elsevier, 1998, ch. 24.

[24] J. S. Yedidia, J. Chen, and M. Fossorier, "Generating code representations suitable for belief propagation decoding," in *Proc. 40th Allerton Conf. Communications, Control, and Computing*, Monticello, IL., Oct. 2002.