Limited-Trial Chase-Like Algorithms Achieving Bounded-Distance Decoding

Jos H. Weber, *Senior Member, IEEE,* and
Marc P. C. Fossorier, *Senior Member, IEEE*

***Abstract*—A soft-decision decoder for an error-correcting block code of Hamming distance d is said to achieve bounded-distance (BD) decoding if its error-correction radius is equal to that of a complete Euclidean distance decoder. The Chase decoding algorithms are reliability-based algorithms achieving BD decoding. The least complex version of the original Chase algorithms ("Chase-3") uses $O(d)$ trials of a conventional binary decoder. In this correspondence, we propose classes of Chase-like BD decoding algorithms of lower complexity than the original Chase-3 algorithm. In particular, the least complex members of these classes require only $O(d^{2/3})$ trials.**

***Index Terms*—Asymptotic optimality, binary linear block codes, bounded-distance (BD) decoding, multitrial decoding, reliability information, soft-decision decoding.**

I. Introduction

Over the years, many *soft-decision* decoding techniques have been proposed for binary linear error-correcting block codes [5]. Although a *maximum-likelihood* (ML) decoding algorithm minimizes the decoding error probability, other suboptimum algorithms are still of interest as well, due to the (prohibitively) high computational complexity of ML decoding for long codes. Of particular interest are the ones which achieve *bounded-distance* (BD) decoding, i.e., for which the error-correction radius is the same as for a complete Euclidean distance decoder. On certain channels, such as the additive white Gaussian noise (AWGN) channel, this property guarantees that an algorithm is asymptotically optimal, i.e., it has the same error performance as ML decoding at high signal-to-noise ratios (SNRs).

Various classes of suboptimum algorithms provide an efficient tradeoff between error performance and decoding complexity. One such class is formed by the *Chase algorithms* [3], running a number of trials of a conventional algebraic binary decoder and thus generating a list of candidate codewords, of which the most likely one is chosen as the final decoding result. In each trial, some of the least reliable symbols are inverted before the actual decoding starts. The inversion patterns, also called test patterns, are taken from a (fixed) test set. Although the Chase decoding approach is rather old, such decoders are still highly relevant. They can not only be used as stand-alone decoders, but also as constituent components in modern techniques like iterative decoding of product codes ("block turbo codes") [8], [1].

All three methods proposed in [3] achieve BD decoding when applied to any binary linear block code $\mathcal{C}$ of length n, dimension k, and Hamming distance d. The numbers of trials are $\binom{n}{\lfloor d/2 \rfloor}$, $2^{\lfloor d/2 \rfloor}$, and $\lfloor d/2 \rfloor + 1$, respectively. Note that for the last one, known as "Chase-3," the number of trials grows only linearly with the Hamming distance d. Limited-trial Chase-like decoding algorithms, not necessarily achieving BD decoding, are proposed in [2]. In particular, [2] provides a $\lceil (d+2)/4 \rceil$-trial BD decoding method. In [9], it is shown that the number of trials can be further reduced to $\lceil d/6 \rceil + 1$ while preserving the BD decoding property. Hence, when considering the ratio between the number of trials and the Hamming distance d when d is approaching infinity, it follows that this ratio can be reduced from $1/2$ (Chase-3) [3] to $1/4$ [2] to $1/6$ [9]. In this correspondence, we present classes of limited-trial Chase-like BD decoding algorithms which show that this ratio can be made arbitrarily small. Most strikingly, it follows that BD decoding is possible using only $O(d^{2/3})$ trials.

The rest of this correspondence is organized as follows. Preliminary matters are given in Section II. Next, limited-trial Chase-like algorithms and their properties are presented in Section III. Finally, the results are discussed in Section IV.

II. Preliminaries

We assume the following setting. A sequence of k information bits is encoded into a codeword $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ according to a binary linear code $\mathcal{C}$. Binary phase-shift keying (BPSK) modulation is used, i.e., a binary codeword $\boldsymbol{x}$ is represented by the vector $\boldsymbol{\chi} = (\chi_1, \chi_2, \ldots, \chi_n)$ through the usual componentwise mapping from $\{0, 1\}$ to $\{\pm 1\}$. Let $\boldsymbol{\rho} = (\rho_1, \rho_2, \ldots, \rho_n)$ denote the received sequence in n-dimensional Euclidean space, and $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ denote the binary hard-decision vector for which each y_i follows merely from the sign of ρ_i. We define the binary error vector by $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{y}$.

A complete channel measurement decoder generates for any received $\boldsymbol{\rho}$ a codeword $\boldsymbol{x}$ minimizing the analog weight of $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{y}$, defined by

$$w_{\boldsymbol{\alpha}}(\boldsymbol{z}) = \sum_{i=1}^{n} \alpha_i z_i \tag{1}$$

where the summation is over the real numbers, and $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ contains the channel measurement information, which is of the format

$$\alpha_i = |\omega_i \rho_i| \tag{2}$$

with the ω_i's being positive weight factors. When choosing $\omega_i = 1$ for all i, then minimizing the analog weight is equivalent to minimizing Euclidean distance. For certain channels, such as the AWGN channel, this is also equivalent to ML decoding. For other channels, fixing the weights ω_i may not be the best thing to do. For example, for the coherent Rayleigh fading channel with channel state information (CSI), ω_i should be set as the fading coefficient [3].

A complete Euclidean distance or channel measurement decoder may be far too complex for practical implementation. The idea behind the Chase decoding approach is to use a binary decoder in a multitrial fashion, in combination with the reliability vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, to obtain a relatively small set of possible binary error patterns, of which one of minimum analog weight is the final decoding result. The algorithms have been designed to work with a conventional (algebraic) binary decoder, which determines the codeword which differs in the least number of places from the decoder's input sequence, provided that this number is not greater than $\lfloor (d-1)/2 \rfloor$. The set of error patterns is obtained by the following procedure. A binary test pattern $\boldsymbol{t} = (t_1, t_2, \ldots, t_n)$ from a test set $\mathcal{T}$ of size l is added to the received vector $\boldsymbol{y}$. The resulting vector

Manuscript received November 20, 2003; revised May 6, 2004. This work was supported by the Dutch Ministry of Economic Affairs, through the Airlink research project (DTC.5961), as part of the policy plan "Competing with ICT Competencies," and by the National Science Foundation under Grant CCR-0098029. The material in this correspondence was presented in part at the IEEE International Symposium on Information Theory, Chicago, IL, June/July 2004.

J. H. Weber is with the Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: j.h.weber@ewi.tudelft.nl).

M. P. C. Fossorier is with the University of Hawaii at Manoa, Honolulu, HI 96822 USA (e-mail: marc@aravis.eng.hawaii.edu).

Communicated by K. A. S. Abdel-Ghaffar, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2004.838103

$\boldsymbol{y}' = \boldsymbol{y} + \boldsymbol{t}$ is fed into the binary decoder, giving either an error pattern $\boldsymbol{z}'$ or a decoding failure. In case an error pattern has been found, the error pattern $\boldsymbol{z}_{\boldsymbol{t}} = \boldsymbol{z}' + \boldsymbol{t}$ relative to $\boldsymbol{y}$ is calculated. After this has been done for all test patterns $\boldsymbol{t}$ from $\mathcal{T}$, a $\boldsymbol{z}_{\boldsymbol{t}}$ with minimum analog weight is determined and added to $\boldsymbol{y}$ to obtain the estimate for the transmitted codeword. In case all l trials result in decoding failures, the decoder output is the received vector $\boldsymbol{y}$ itself. In [3] Chase has proposed three algorithms, i.e., three test sets $\mathcal{T}$. For Algorithm 1 the set $\mathcal{T}$ consists of all binary vectors of length n which contain exactly $\lfloor d/2 \rfloor$ ones. For Algorithm 2 the set $\mathcal{T}$ consists of all binary vectors of length n which have the $n - \lfloor d/2 \rfloor$ most reliable positions equal to zero. For Algorithm 3 the set $\mathcal{T}$ consists of all binary vectors of length n which contain ones in the i least reliable positions and zeros elsewhere, where $i = 0, 2, 4, \ldots, d-1$ if d is odd and $i = 0, 1, 3, 5, \ldots, d-1$ if d is even.

The error-correction radius of a soft-decision decoding algorithm $\mathcal{A}$ is defined as the radius of the largest open sphere in Euclidean space centered at a codeword $\boldsymbol{\chi}$, such that all vectors in the spheres are decoded to $\boldsymbol{\chi}$ by $\mathcal{A}$. Algorithm $\mathcal{A}$ is said to achieve BD decoding if the squared error correction radius Δ equals the code's Hamming distance d. In case of an AWGN channel, $\boldsymbol{\rho}$ is the sum of the transmitted sequence $\boldsymbol{\chi}$ and the noise sequence $\boldsymbol{\nu} = (\nu_1, \nu_2, \ldots, \nu_n)$, where the ν_is are independent zero-mean Gaussian random variables. On such a channel, the asymptotic loss of algorithm $\mathcal{A}$ compared to ML decoding is $10 \log_{10}(d/\Delta)$ dB. Hence, any BD decoding algorithm $\mathcal{A}$ has an asymptotic loss of 0 dB, and is thus asymptotically optimal, i.e., $\log P_{\mathrm{ML}} / \log P_{\mathcal{A}} \rightarrow 1$ as the SNR approaches infinity, where P_{ML} denotes the decoding error probability for an ML decoding algorithm, and P_A denotes the decoding error probability of the suboptimum decoding algorithm $\mathcal{A}$.

III. Limited-Trial Chase-Like Decoding

All three Chase algorithms achieve BD decoding, and give essentially the same performance as ML decoding at high SNR in case of binary antipodal signaling and transmission over the AWGN or coherent Rayleigh fading channel [3]. In this section, we propose Chase-like decoding algorithms of lower complexity, but still achieving BD decoding. Before presenting our algorithms, we first introduce some notations. For $i = 0, 1, \ldots, n$, let $\boldsymbol{t}_i$ denote the test pattern of length n which contains ones in the i most unreliable positions and zeros elsewhere. Let $\boldsymbol{a}^j$ denote the concatenation of j times the vector $\boldsymbol{a}$, e.g., $(01)^3 0^2 = 01010100$. For convenience, we assume without loss of generality throughout the rest of this correspondence that the ordering of the received symbols is such that

$$\alpha_i \geq \alpha_{i+1} \tag{3}$$

for $i = 1, \ldots n-1$. Under this assumption

$$\boldsymbol{t}_i = 0^{n-i} 1^i \tag{4}$$

for $i = 0, 1, \ldots, n$. Note that using this notation, the Chase-3 test set reads $\{\boldsymbol{t}_0, \boldsymbol{t}_2, \boldsymbol{t}_4, \ldots, \boldsymbol{t}_{d-1}\}$ if d is odd, and $\{\boldsymbol{t}_0, \boldsymbol{t}_1, \boldsymbol{t}_3, \ldots, \boldsymbol{t}_{d-1}\}$ if d is even. Obviously, the original Chase-3 inversion patterns were inspired by Forney's Generalized Minimum Distance (GMD) decoding algorithm [4]. In the GMD decoding approach, which is valid for both binary and nonbinary linear codes, unreliable symbols are erased in the various trials. It is most effective to choose the parity of the number of erasures in a trial complementary to the parity of the code's Hamming distance. However, [2] shows that this is not necessarily the case when using inversions instead of erasures, i.e., when applying the Chase decoding approach.

For codes of even Hamming distance d, we allow a small improvement in the binary decoder, like in [6]. A binary decoder for a code of even Hamming distance d can be designed in such a way that any pattern of up to $d/2 - 1$ errors is corrected, while $d/2$ errors are corrected if one of these errors occurs in a predetermined position, for which we choose the most unreliable. Throughout the rest of this correspondence we assume the binary decoder is implemented using this feature, as it allows to unify the cases d even and d odd.

Now we are ready to present two new classes of Chase-like BD decoding algorithms for binary linear block codes. As in the Chase-3 [3] and Arico-Weber [2] algorithms, all test patterns are of the $\boldsymbol{t}_i$ format. The two classes use test sets $\mathcal{T}^1_{d,m}$ and $\mathcal{T}^2_{d,m}$, respectively, where d is the Hamming distance of the code and m is any integer satisfying $m \geq 3$ and $m^2 - m + 1 \leq d$. The test sets in Class 1 have a simple structure and suffice to demonstrate our main result, i.e., they show that BD decoding is possible in $O(d^{2/3})$ trials. The test sets in Class 2 are subsets of the corresponding test sets from Class 1, i.e., $\mathcal{T}^2_{d,m} \subseteq \mathcal{T}^1_{d,m}$. Hence, Class 2 realizes further complexity reductions (but not below $O(d^{2/3})$).

A. Class 1

The test sets in Class 1 are defined by

$$\mathcal{T}^1_{d,m} = \left(\bigcup_{j=0}^{\frac{m^2-3m}{2}} \{\boldsymbol{t}_{d-2j}\} \right) \cup \left(\bigcup_{j=1}^{\left\lceil \frac{d-m^2+m}{2m} \right\rceil} \{\boldsymbol{t}_{d-m^2+3m-2mj}\} \right) \cup \{\boldsymbol{t}_0\}. \tag{5}$$

For $m = 3$ we obtain the test set presented in [9].

Theorem 1: For any binary linear code of length n and Hamming distance $d < n$, and any integer m such that $m \geq 3$ and $m^2 - m + 1 \leq d$, the Chase-like decoder with test set $\mathcal{T}^1_{d,m}$ and reliability values set as $\alpha_i = |\rho_i|$ for all i achieves BD decoding.

This result follows by applying the method for evaluating the error-correction radius of reliability-based soft-decision decoding algorithms proposed in [5]. The main steps of the procedure are given in the Appendix .

Note from (5) that

$$|\mathcal{T}^1_{d,m}| = 2 + \frac{m^2 - 3m}{2} + \left\lceil \frac{d - m^2 + m}{2m} \right\rceil. \tag{6}$$

Consequently, for any $m \geq 3$

$$\lim_{d \to \infty} \frac{|\mathcal{T}^1_{d,m}|}{d} = \frac{1}{2m} \tag{7}$$

i.e., the ratio between the number of trials and the Hamming distance d is $1/(2m)$ in case d is approaching infinity. Hence, this ratio, which is $1/2$ for the Chase-3 algorithm [3] and $1/4$ for the method presented in [2], can be made arbitrarily small while maintaining the BD decoding property.

To find the smallest test set among the $\mathcal{T}^1_{d,m}$ in case d is finite, the expression from (6) should be minimized over all possible m. For example, we consider the case $d = 95$ in Table I. Note that $m = 4$ and $m = 5$ give the smallest test sets. Hence, BD decoding of a code of Hamming distance 95 can be achieved by Chase-like decoding in 15 trials only. For comparison, also the considerably larger test sets from [3] (Chase-3, of size 48) and [2] (of size 25) have been included in Table I.

In general, it follows from (6) that

$$|\mathcal{T}^1_{d,m}| = \frac{d}{2m} + \frac{m^2}{2} + O(m). \tag{8}$$

TABLE I
TEST SETS IN CASE $d = 95$

algorithm	test set	size
Chase-3 [3]	$\{\mathbf{t}_{94}, \mathbf{t}_{92}, \mathbf{t}_{90}, \mathbf{t}_{88}, \mathbf{t}_{86}, \mathbf{t}_{84}, \mathbf{t}_{82}, \mathbf{t}_{80}, \mathbf{t}_{78}, \mathbf{t}_{76}, \mathbf{t}_{74}, \mathbf{t}_{72}, \mathbf{t}_{70}, \mathbf{t}_{68}, \mathbf{t}_{66}, \mathbf{t}_{64}, \mathbf{t}_{62}, \mathbf{t}_{60}, \ldots, \mathbf{t}_{14}, \mathbf{t}_{12}, \mathbf{t}_{10}, \mathbf{t}_{8}, \mathbf{t}_{6}, \mathbf{t}_{4}, \mathbf{t}_{2}, \mathbf{t}_{0}\}$	48
Arico/Weber [2]	$\{\mathbf{t}_{93}, \mathbf{t}_{89}, \mathbf{t}_{85}, \mathbf{t}_{81}, \mathbf{t}_{77}, \mathbf{t}_{73}, \mathbf{t}_{69}, \mathbf{t}_{65}, \mathbf{t}_{61}, \mathbf{t}_{57}, \mathbf{t}_{53}, \mathbf{t}_{49}, \mathbf{t}_{45}, \mathbf{t}_{41}, \mathbf{t}_{37}, \mathbf{t}_{33}, \mathbf{t}_{29}, \mathbf{t}_{25}, \mathbf{t}_{21}, \mathbf{t}_{17}, \mathbf{t}_{13}, \mathbf{t}_{9}, \mathbf{t}_{5}, \mathbf{t}_{1}, \mathbf{t}_{0}\}$	25
Weber [9]	$\{\mathbf{t}_{95}, \mathbf{t}_{89}, \mathbf{t}_{83}, \mathbf{t}_{77}, \mathbf{t}_{71}, \mathbf{t}_{65}, \mathbf{t}_{59}, \mathbf{t}_{53}, \mathbf{t}_{47}, \mathbf{t}_{41}, \mathbf{t}_{35}, \mathbf{t}_{29}, \mathbf{t}_{23}, \mathbf{t}_{17}, \mathbf{t}_{11}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	17
Class 1, $m = 3$	$\{\mathbf{t}_{95}, \mathbf{t}_{89}, \mathbf{t}_{83}, \mathbf{t}_{77}, \mathbf{t}_{71}, \mathbf{t}_{65}, \mathbf{t}_{59}, \mathbf{t}_{53}, \mathbf{t}_{47}, \mathbf{t}_{41}, \mathbf{t}_{35}, \mathbf{t}_{29}, \mathbf{t}_{23}, \mathbf{t}_{17}, \mathbf{t}_{11}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	17
Class 1, $m = 4$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{83}, \mathbf{t}_{75}, \mathbf{t}_{67}, \mathbf{t}_{59}, \mathbf{t}_{51}, \mathbf{t}_{43}, \mathbf{t}_{35}, \mathbf{t}_{27}, \mathbf{t}_{19}, \mathbf{t}_{11}, \mathbf{t}_{3}, \mathbf{t}_{0}\}$	15
Class 1, $m = 5$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{75}, \mathbf{t}_{65}, \mathbf{t}_{55}, \mathbf{t}_{45}, \mathbf{t}_{35}, \mathbf{t}_{25}, \mathbf{t}_{15}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	15
Class 1, $m = 6$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{79}, \mathbf{t}_{77}, \mathbf{t}_{65}, \mathbf{t}_{53}, \mathbf{t}_{41}, \mathbf{t}_{29}, \mathbf{t}_{17}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	17
Class 1, $m = 7$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{79}, \mathbf{t}_{77}, \mathbf{t}_{75}, \mathbf{t}_{73}, \mathbf{t}_{71}, \mathbf{t}_{69}, \mathbf{t}_{67}, \mathbf{t}_{53}, \mathbf{t}_{39}, \mathbf{t}_{25}, \mathbf{t}_{11}, \mathbf{t}_{0}\}$	20
Class 1, $m = 8$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{79}, \mathbf{t}_{77}, \mathbf{t}_{75}, \mathbf{t}_{73}, \mathbf{t}_{71}, \mathbf{t}_{69}, \mathbf{t}_{67}, \mathbf{t}_{65}, \mathbf{t}_{63}, \mathbf{t}_{61}, \mathbf{t}_{59}, \mathbf{t}_{57}, \mathbf{t}_{55}, \mathbf{t}_{39}, \mathbf{t}_{23}, \mathbf{t}_{7}, \mathbf{t}_{0}\}$	25
Class 1, $m = 9$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{79}, \mathbf{t}_{77}, \mathbf{t}_{75}, \mathbf{t}_{73}, \mathbf{t}_{71}, \mathbf{t}_{69}, \mathbf{t}_{67}, \ldots, \mathbf{t}_{51}, \mathbf{t}_{49}, \mathbf{t}_{47}, \mathbf{t}_{45}, \mathbf{t}_{43}, \mathbf{t}_{41}, \mathbf{t}_{23}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	31
Class 1, $m = 10$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{79}, \mathbf{t}_{77}, \mathbf{t}_{75}, \mathbf{t}_{73}, \mathbf{t}_{71}, \mathbf{t}_{69}, \mathbf{t}_{67}, \ldots, \mathbf{t}_{39}, \mathbf{t}_{37}, \mathbf{t}_{35}, \mathbf{t}_{33}, \mathbf{t}_{31}, \mathbf{t}_{29}, \mathbf{t}_{27}, \mathbf{t}_{25}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	38
Class 2, $m = 3$	$\{\mathbf{t}_{95}, \mathbf{t}_{89}, \mathbf{t}_{83}, \mathbf{t}_{77}, \mathbf{t}_{71}, \mathbf{t}_{65}, \mathbf{t}_{59}, \mathbf{t}_{53}, \mathbf{t}_{47}, \mathbf{t}_{41}, \mathbf{t}_{35}, \mathbf{t}_{29}, \mathbf{t}_{23}, \mathbf{t}_{17}, \mathbf{t}_{11}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	17
Class 2, $m = 4$	$\{\mathbf{t}_{95}, \mathbf{t}_{91}, \mathbf{t}_{83}, \mathbf{t}_{75}, \mathbf{t}_{67}, \mathbf{t}_{59}, \mathbf{t}_{51}, \mathbf{t}_{43}, \mathbf{t}_{35}, \mathbf{t}_{27}, \mathbf{t}_{19}, \mathbf{t}_{11}, \mathbf{t}_{3}, \mathbf{t}_{0}\}$	14
Class 2, $m = 5$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{89}, \mathbf{t}_{85}, \mathbf{t}_{75}, \mathbf{t}_{65}, \mathbf{t}_{55}, \mathbf{t}_{45}, \mathbf{t}_{35}, \mathbf{t}_{25}, \mathbf{t}_{15}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	13
Class 2, $m = 6$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{89}, \mathbf{t}_{83}, \mathbf{t}_{77}, \mathbf{t}_{65}, \mathbf{t}_{53}, \mathbf{t}_{41}, \mathbf{t}_{29}, \mathbf{t}_{17}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	12
Class 2, $m = 7$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{79}, \mathbf{t}_{73}, \mathbf{t}_{67}, \mathbf{t}_{53}, \mathbf{t}_{39}, \mathbf{t}_{25}, \mathbf{t}_{11}, \mathbf{t}_{0}\}$	13
Class 2, $m = 8$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{87}, \mathbf{t}_{85}, \mathbf{t}_{79}, \mathbf{t}_{71}, \mathbf{t}_{63}, \mathbf{t}_{55}, \mathbf{t}_{39}, \mathbf{t}_{23}, \mathbf{t}_{7}, \mathbf{t}_{0}\}$	13
Class 2, $m = 9$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{75}, \mathbf{t}_{73}, \mathbf{t}_{65}, \mathbf{t}_{57}, \mathbf{t}_{49}, \mathbf{t}_{41}, \mathbf{t}_{23}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	16
Class 2, $m = 10$	$\{\mathbf{t}_{95}, \mathbf{t}_{93}, \mathbf{t}_{91}, \mathbf{t}_{89}, \mathbf{t}_{85}, \mathbf{t}_{83}, \mathbf{t}_{81}, \mathbf{t}_{75}, \mathbf{t}_{73}, \mathbf{t}_{65}, \mathbf{t}_{55}, \mathbf{t}_{45}, \mathbf{t}_{35}, \mathbf{t}_{25}, \mathbf{t}_{5}, \mathbf{t}_{0}\}$	16

Since $d/(2m) + m^2/2$ achieves a minimum value of

$$3 \times 2^{-5/3} \times d^{2/3} \approx 0.94 \times d^{2/3} \tag{9}$$

for $m = (d/2)^{1/3}$, we have the following important result.

Corollary 1: Bounded-distance decoding can be achieved by a Chase-like decoder using $O(d^{2/3})$ trials.

B. Class 2

It may be possible to remove some of the test patterns from the Class 1 test set $\mathcal{T}^1_{d,m}$, while maintaining the BD decoding property. The removal strategy is inspired by the iterative process from [5] for computation of the error-correction radius, as explained in the Appendix . The resulting test sets in Class 2 are given by

$$\mathcal{T}^2_{d,m} = \{\boldsymbol{t}_d\} \cup (\cup_{h \in \mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3} \{\boldsymbol{t}_h\}) \cup \{\boldsymbol{t}_0\} \tag{10}$$

where (see (11)–(12) at the bottom of the page) and

$$\mathcal{I}_3 = \bigcup_{j=1}^{\left\lceil \frac{d-m^2+m}{2m} \right\rceil} \left\{ d - m^2 + 3m - 2jm \right\}. \tag{13}$$

Theorem 2: For any binary linear code of length n and Hamming distance $d < n$, and any integer m such that $m \geq 3$ and $m^2 - m + 1 \leq d$, the Chase-like decoder with test set $\mathcal{T}^2_{d,m}$ and reliability values set as $\alpha_i = |\rho_i|$ for all i achieves BD decoding.

Again, the proof is provided in the Appendix.
Note from (10)–(13) that

$$\begin{aligned} |\mathcal{T}^2_{d,m}| &= 1 + \sum_{i=1}^{\lfloor \frac{m}{2} \rfloor - 1} \left(\left\lceil \frac{m}{2} \right\rceil - i \right) + \left\lceil \frac{m}{2} \right\rceil - 2 \\ &\quad + \left\lceil \frac{d - m^2 + m}{2m} \right\rceil + 1 \\ &= \sum_{i=1}^{\lceil \frac{m}{2} \rceil - 1} i - \left(\left\lceil \frac{m}{2} \right\rceil - \left\lfloor \frac{m}{2} \right\rfloor \right) + \left\lceil \frac{m}{2} \right\rceil \\ &\quad + \left\lceil \frac{d - m^2 + m}{2m} \right\rceil \\ &= \frac{1}{2} \left\lceil \frac{m}{2} \right\rceil^2 - \frac{1}{2} \left\lceil \frac{m}{2} \right\rceil + \left\lfloor \frac{m}{2} \right\rfloor + \left\lceil \frac{d - m^2 + m}{2m} \right\rceil. \end{aligned} \tag{14}$$

$$\mathcal{I}_1 = \bigcup_{i=1}^{\lfloor \frac{m}{2} \rfloor - 1} \left(\left(\bigcup_{j=1}^{\lceil \frac{m}{2} \rceil - 1 - i} \left\{ d - 2(i-1) \left\lceil \frac{m}{2} \right\rceil - 2j \right\} \right) \bigcup \left\{ d - 2i \left\lceil \frac{m}{2} \right\rceil \right\} \right) \tag{11}$$

$$\mathcal{I}_2 = \bigcup_{j=1}^{\lceil \frac{m}{2} \rceil - 2} \left\{ d - 2 \left\lceil \frac{m}{2} \right\rceil \left(\left\lfloor \frac{m}{2} \right\rfloor - 1 \right) - 2j \left\lfloor \frac{m}{2} \right\rfloor \right\} \tag{12}$$

TABLE II
OVERVIEW OF LIMITED-TRIAL CHASE-LIKE ALGORITHMS ACHIEVING BD DECODING

algorithm	remarks	test set	size of test set l	l/d, $d \to \infty$
Chase-3 [3]	$d \geq 3$	$\{\mathbf{t}_0, \mathbf{t}_{d-1}, \mathbf{t}_{d-3}, \mathbf{t}_{d-5}, \dots\}$	$\lfloor d/2+1 \rfloor$	$1/2$
Arico/Weber [2]	$d \geq 3$	$\{\mathbf{t}_0, \mathbf{t}_{d-2}, \mathbf{t}_{d-6}, \mathbf{t}_{d-10}, \dots\}$	$\lceil (d+2)/4 \rceil$	$1/4$
Weber [9]	$d \geq 7$	$\{\mathbf{t}_0, \mathbf{t}_d, \mathbf{t}_{d-6}, \mathbf{t}_{d-12}, \dots\}$	$\lceil d/6+1 \rceil$	$1/6$
Class 1	$m \geq 3$, $d \geq m^2-m+1$	$\{\mathbf{t}_0, \mathbf{t}_d, \mathbf{t}_{d-2}, \dots, \mathbf{t}_{d-m^2+3m}, \mathbf{t}_{d-m^2+m}, \mathbf{t}_{d-m^2-m}, \dots\}$	$2+m(m-3)/2+\lceil (d-m^2+m)/(2m) \rceil = d/(2m)+m^2/2+O(m)$	$1/(2m)$
Class 2	$m \geq 3$, $d \geq m^2-m+1$	as defined in (10)-(13)	$\lceil m/2 \rceil^2/2 - \lceil m/2 \rceil/2 + \lfloor m/2 \rfloor + \lceil (d-m^2+m)/(2m) \rceil = d/(2m)+m^2/8+O(m)$	$1/(2m)$

Consequently, for any $m \geq 3$,

$$\lim_{d\to\infty} \frac{|\mathcal{T}^2_{d,m}|}{d} = \frac{1}{2m}. \tag{15}$$

From (7) and (15) we conclude that Class 2 offers no essential complexity reduction with respect to Class 1 in case m is finite and the Hamming distance d approaches infinity.

However, for finite values of d, Class 2 may offer substantial complexity reductions. Revisiting the $d = 95$ example, note from Table I that $|\mathcal{T}^2_{95,m}|$ is minimum for $m = 6$. Hence, BD decoding of a code of Hamming distance 95 can be achieved by Chase-like decoding in $|\mathcal{T}^2_{95,6}| = 12$ trials only, a reduction of three trials compared to the smallest Class 1 test set.

In general, it follows from (14) that

$$|\mathcal{T}^2_{d,m}| = \frac{d}{2m} + \frac{m^2}{8} + O(m). \tag{16}$$

Since $d/(2m) + m^2/8$ achieves a minimum value of

$$3 \times 2^{-7/3} \times d^{2/3} \approx 0.60 \times d^{2/3} \tag{17}$$

for $m = (2d)^{1/3}$, there is a clear reduction in the number of trials in comparison to Class 1 (see (9)). Still, the minimum number of trials is $O(d^{2/3})$ for Class 2 as well.

IV. DISCUSSION

In this correspondence, we have presented Chase-like decoders which enable BD decoding of binary linear block codes of Hamming distance d in only $O(d^{2/3})$ trials, whereas the least complex Chase(-like) algorithms known so far require $O(d)$ trials. An overview of limited-trial Chase-like algorithms achieving BD decoding is provided in Table II. The Class 1 and 2 algorithms proposed in this correspondence are similar to the Chase-3 algorithm, in the sense that all test patterns are of the $\boldsymbol{t}_i$ format, i.e., only the i least reliable received bits are inverted. A first improvement in comparison to the Chase-3 algorithm, as already proposed in [2], is to choose the parity of i equal to the parity of d, rather than its complement (with the possible exception of $i = 0$). Hence, the proposed test sets are subsets of $\mathcal{T}_d = \cup_j \{\boldsymbol{t}_{d\pm 2j}\} \cup \{\boldsymbol{t}_0\}$. The next improvement is the deliberate removal of test patterns from $\mathcal{T}_d$, while preserving the BD decoding property. Several strategies, represented by the resulting test sets $\mathcal{T}^1_{d,m}$ and $\mathcal{T}^2_{d,m}$ have been given. The smallest of these sets have sizes of $O(d^{2/3})$, which shows the claimed result. An interesting research challenge is to investigate whether or not even less complex Chase-like BD decoding algorithms do exist.

For the AWGN channel and BPSK signaling, the BD decoding property guarantees optimal error performance when the SNR approaches infinity. The results presented in this correspondence are mostly of theoretical importance, as they apply to high SNRs, extremely low error rates, and large values of d. For practical SNR values the impact is much smaller. Significant complexity savings are only obtained for codes with a large minimum Hamming distance. However, for such codes, the BD decoding criterion does not reflect well the error performance at practical error rates [7]. For Chase-like decoders, the number of test patterns influences the error performance more than the Euclidean error correction radius at practical error rates.

APPENDIX

In this Appendix, we prove Theorems 1 and 2, i.e., we show that the Chase-like decoders proposed in this correspondence achieve BD decoding in case we set $\alpha_i = |\rho_i|$ for all i. To do so, we apply the method from [5], which evaluates the error-correction radii of reliability-based soft-decision decoding algorithms. For a Chase-like decoding algorithm $\mathcal{A}$, this method can be described as follows.

1. Identify among all binary error vectors $\boldsymbol{z}$ the most likely vector $\boldsymbol{e}$ such that the transmitted codeword is not generated in any of the trials of $\mathcal{A}$ when $\boldsymbol{e}$ occurs. In general, the weight w of $\boldsymbol{e}$ is minimum among all valid error vectors, and the w ones in $\boldsymbol{e}$ are in positions which are as unreliable as possible. For $i = 1, 2, \dots, w$, define a_i as the number of zeroes directly following the i^{th} one in $\boldsymbol{e}$, i.e., $\boldsymbol{e} = 0^s 10^{a_1} 10^{a_2} \cdots 10^{a_w}$, with
$$s = n - w - \sum_{i=1}^{w} a_i.$$
2. Initially, set $h := w$, $N_i := 2a_i$, $D_i := a_i + 1$, and $A_i := N_i/D_i$ for $i = 1, 2, \dots, w$. Define $\boldsymbol{A}^{\text{INI}} = (A_1, A_2, \dots, A_w)$.
3. If there exists j, $1 \leq j \leq h-1$, such that $A_j < A_{j+1}$ and $A_i \geq A_{i+1}$ for $i = 1, 2, \dots, j-1$, then merge entries j and $j+1$ into one new entry, i.e., reset $h := h-1$, $N_j := N_j + N_{j+1}$, $D_j := D_j + D_{j+1}$, $A_j := N_j/D_j$, $N_i := N_{i+1}$, $D_i := D_{i+1}$, and $A_i := A_{i+1}$ for $i = j+1, j+2, \dots, h$. Repeat this until $A_j \geq A_{j+1}$ for all $j = 1, 2, \dots, h-1$. Then, define $\boldsymbol{A}^{\text{FIN}} = (A_1, A_2, \dots, A_h)$.
4. The squared error-correction radius of Algorithm $\mathcal{A}$ is
$$\Delta = \min\{d, \sum_{i=1}^{h} ((D_i - N_i/2)M_i^2 + (N_i/2)(2-M_i)^2)\} \tag{18}$$
where $M_i = \max\{A_i, 1\}$ for all i.

For the algorithms based on the test sets $\mathcal{T}^a_{d,m}$ ($a = 1, 2$) under consideration in this correspondence, the presence of test pattern $\boldsymbol{t}_0$ implies that $\boldsymbol{e}$ contains at least $\lceil d/2 \rceil$ ones in the first $n-b$ positions, where

$$b = \begin{cases} 0, & \text{if } d \text{ is odd} \\ 1, & \text{if } d \text{ is even.} \end{cases} \tag{19}$$

Furthermore, $\boldsymbol{e}$ contains at least $\lceil d/2 \rceil - (i-1)/2$ ones in the first $n-i-b$ positions, for any odd $i \geq 1$ such that $\boldsymbol{t}_{i+b}$ is in the test set.

For the case $a = 1$ (Class 1), it follows from (5) that the error vector with the minimum number of ones and with these ones in the least reliable positions, while satisfying the restrictions just mentioned, is

$$\boldsymbol{e} = 0^{n-d-1}(10)^{\frac{m^2-3m}{2}}(10^m 1^{m-1})^{\left\lceil \frac{d-m^2+m}{2m} \right\rceil} 10^v 1^{v-1} 0^b \quad (20)$$

where b is as defined in (19) and

$$v = \left\lceil \frac{d - m^2 + 3m - 2m\left\lceil \frac{d-m^2+m}{2m} \right\rceil}{2} \right\rceil. \quad (21)$$

Hence,

$$\boldsymbol{A}^{\mathrm{INI}} = \left(\frac{2}{2}\right)^{\frac{m^2-3m}{2}} \left(\left(\frac{2m}{m+1}\right)\left(\frac{0}{1}\right)^{m-1} \right)^{\left\lceil \frac{d-m^2+m}{2m} \right\rceil} \boldsymbol{v} \quad (22)$$

where

$$\boldsymbol{v} = \begin{cases} \left(\frac{2v}{v+1}\right)\left(\frac{0}{1}\right)^{v-1}, & \text{if } d \text{ is odd} \\ \left(\frac{4}{3}\right), & \text{if } d \text{ is even and } v = 1 \\ \left(\frac{2v}{v+1}\right)\left(\frac{0}{1}\right)^{v-2}\left(\frac{2}{2}\right), & \text{if } d \text{ is even and } v \geq 2 \end{cases} \quad (23)$$

with v as defined in (21). After performing the iterative process, the final solution is

$$\boldsymbol{A}^{\mathrm{FIN}} = \left(\frac{m^2-m}{m^2-2m+1}\right)\left(\frac{2m}{2m}\right)^{\left\lceil \frac{d-m^2-m}{2m} \right\rceil} \boldsymbol{v}^* \quad (24)$$

where

$$\boldsymbol{v}^* = \begin{cases} \left(\frac{2v}{v+m}\right)\left(\frac{0}{1}\right)^{v-1}, & \text{if } d \text{ is odd} \\ \left(\frac{2v+2}{2v+m}\right), & \text{if } d \text{ is even and } m > v^2 - v \\ \left(\frac{2v}{v+m}\right)\left(\frac{2}{v}\right), & \text{if } d \text{ is even and } m \leq v^2 - v \end{cases} \quad (25)$$

with v as defined in (21). Since $1 \leq v \leq m$, it follows that $M_1 = m/(m-1)$ and $M_i = 1$ for all $i \geq 2$, and thus that

$$\begin{aligned} \Delta &= \min\{d, ((m^2-3m+2)/2)(m/(m-1))^2 \\ &\quad +((m^2-m)/2)(2-m/(m-1))^2 + d + 1 - (m^2-2m+1)\} \\ &= \min\{d, d\} = d. \end{aligned} \quad (26)$$

Hence, BD decoding is achieved indeed for Class 1 algorithms.

For the case $a = 2$ (Class 2), the BD decoding property can be proved similarly. Actually, the removal of test patterns from $\mathcal{T}^1_{d,m}$ resulting in $\mathcal{T}^2_{d,m}$ has been done in such a way that the iterative evaluation process leads to a solution which is effectively the same as (24). This will be illustrated for the case $d = 95$ and $m = 4$, for which (20) and (22) are

$$\boldsymbol{e} = 0^{n-96}1010(10000111)^{11}1001 \quad (27)$$

and

$$\boldsymbol{A}^{\mathrm{INI}} = \left(\frac{2}{2}\right)\left(\frac{2}{2}\right)\left(\left(\frac{8}{5}\right)\left(\frac{0}{1}\right)^3\right)^{11}\left(\frac{4}{3}\right)\left(\frac{0}{1}\right) \quad (28)$$

respectively, leading to the final solution (24) reading

$$\begin{aligned} \boldsymbol{A}^{\mathrm{FIN}} &= \left(\frac{2+2+8}{2+2+5}\right)\left(\frac{0+0+0+8}{1+1+1+5}\right)^{10}\left(\frac{0+0+0+4}{1+1+1+3}\right)\left(\frac{0}{1}\right) \\ &= \left(\frac{12}{9}\right)\left(\frac{8}{8}\right)^{10}\left(\frac{4}{6}\right)\left(\frac{0}{1}\right). \end{aligned} \quad (29)$$

Removing the test pattern $\boldsymbol{t}_{93}$ from $\mathcal{T}^1_{95,4}$ (see Table I), the Class 2 equivalents of (27) and (28) are

$$\boldsymbol{e} = 0^{n-96}1001(10000111)^{11}1001 \quad (30)$$

and

$$\boldsymbol{A}^{\mathrm{INI}} = \left(\frac{4}{3}\right)\left(\frac{0}{1}\right)\left(\left(\frac{8}{5}\right)\left(\frac{0}{1}\right)^3\right)^{11}\left(\frac{4}{3}\right)\left(\frac{0}{1}\right) \quad (31)$$

respectively, leading to the final solution

$$\begin{aligned} \boldsymbol{A}^{\mathrm{FIN}} &= \left(\frac{4}{3}\right)\left(\frac{0+8}{1+5}\right)\left(\frac{0+0+0+8}{1+1+1+5}\right)^{10}\left(\frac{0+0+0+4}{1+1+1+3}\right)\left(\frac{0}{1}\right) \\ &= \left(\frac{4}{3}\right)\left(\frac{8}{6}\right)\left(\frac{8}{8}\right)^{10}\left(\frac{4}{6}\right)\left(\frac{0}{1}\right). \end{aligned} \quad (32)$$

Since consecutive entries in $\boldsymbol{A}^{\mathrm{FIN}}$ of equal value may be merged (e.g., $(4/3)(8/6)$ may be replaced by $(12/9)$) without affecting the final outcome of the evaluation algorithm, (29) and (32) are effectively the same, both leading to

$$\begin{aligned} \Delta &= \min\{95, 3 \times (4/3)^2 + 6 \times (2/3)^2 + 95 + 1 - 9\} \\ &= \min\{95, 95\} = 95. \end{aligned} \quad (33)$$

In general, it follows from (10) that the Class 2 equivalent of (20) (i.e., $\boldsymbol{e}$) is given by the concatenation of the binary strings

$$\begin{aligned} &0^{n-d-1} \\ &(10)^{\left\lceil \frac{m-2-2i}{2} \right\rceil} 10^{i+1}1^i, \qquad \text{for } i = 1, \ldots, \left\lfloor \frac{m-2}{2} \right\rfloor \\ &(10^{\left\lfloor \frac{m}{2} \right\rfloor} 1^{\left\lfloor \frac{m-2}{2} \right\rfloor})^{\left\lceil \frac{m-4}{2} \right\rceil} \\ &(10^m 1^{m-1})^{\left\lceil \frac{d-m^2+m}{2m} \right\rceil} \quad \text{and} \\ &10^v 1^{v-1} 0^b \end{aligned} \quad (34)$$

where b and v are as defined in (19) and (21), respectively. The Class 2 equivalent of (22) (i.e., $\boldsymbol{A}^{\mathrm{INI}}$) is the concatenation of the strings

$$\begin{aligned} &\left(\frac{2}{2}\right)^{\left\lceil \frac{m-2-2i}{2} \right\rceil}\left(\frac{2+2i}{2+i}\right)\left(\frac{0}{1}\right)^i, \qquad \text{for } i = 1, \ldots, \left\lfloor \frac{m-2}{2} \right\rfloor \\ &\left(\left(\frac{2\left\lfloor \frac{m}{2} \right\rfloor}{\left\lfloor \frac{m+2}{2} \right\rfloor}\right)\left(\frac{0}{1}\right)^{\left\lfloor \frac{m-2}{2} \right\rfloor}\right)^{\left\lceil \frac{m-4}{2} \right\rceil} \quad \text{and} \\ &\left(\left(\frac{2m}{m+1}\right)\left(\frac{0}{1}\right)^{m-1}\right)^{\left\lceil \frac{d-m^2+m}{2m} \right\rceil} \boldsymbol{v} \end{aligned} \quad (35)$$

where $\boldsymbol{v}$ is as defined in (23). After application of the iterative evaluation process on (35), the squared error-correction radius Δ is again given by (26). Hence, we can conclude that Class 2 algorithms achieve BD decoding as well.

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers and the Associate Editor for their thoughtful comments and valuable suggestions.

REFERENCES

[1] C. Argon, S. W. McLaughlin, and T. Souvignier, "Iterative application of the Chase algorithm on Reed-Solomon product codes," in *Proc. IEEE Int. Conf. Communications*, Helsinki, Finland, June 11–14, 2001, pp. 320–324.

[2] G. Arico and J. H. Weber, "Limited-trial Chase decoding," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2972–2975, Nov. 2003.

[3] D. Chase, "A class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 170–182, Jan. 1972.
[4] G. D. Forney, Jr., "Generalized minimum distance decoding," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 125–131, Apr. 1966.
[5] M. P. C. Fossorier and S. Lin, "A unified method for evaluating the error-correction radius of reliability-based soft-decision algorithms for linear block codes," *IEEE Trans. Inform. Theory*, vol. 44, pp. 691–700, Mar. 1998.
[6] ——, "Chase-type and GMD coset decodings," *IEEE Trans. Commun.*, vol. 48, pp. 345–350, Mar. 2000.
[7] ——, "Error performance analysis for reliability-based decoding algorithms," *IEEE Trans. Inform. Theory*, vol. 48, pp. 287–293, Jan. 2002.
[8] R. M. Pyndiah, "Near-optimum decoding of product codes: Block turbo codes," *IEEE Trans. Commun.*, vol. 46, pp. 1003–1010, Aug. 1998.
[9] J. H. Weber, "Low-complexity Chase-like bounded-distance decoding algorithms," in *Proc. IEEE GLOBECOM*, San Francisco, CA, Dec. 1–5, 2003, pp. 1608–1612.

A Finite Gilbert–Varshamov Bound for Pure Stabilizer Quantum Codes

Keqin Feng and Zhi Ma

***Abstract*—A finite Gilbert–Varshamov (GV) bound for pure stabilizer (binary and nonbinary) quantum error correcting codes is presented in analogy to the GV bound for classical codes by using several enumerative results in finite unitary geometry. From this quantum GV bound we obtain several new binary quantum codes in a nonconstructive way having better parameters than the known codes.**

***Index Terms*—Finite fields, finite unitary geometry, quantum codes, quantum Gilbert–Varshamov (GV) bound.**

I. Introduction

The theory of quantum error-correcting codes has been developed rapidly in recent years. Many good q-ary quantum codes have been constructed by using classical error-correcting codes over $\mathbb{F}_q$ or $\mathbb{F}_{q^2}$ with special orthogonal properties. Among these constructive methods, the following result we used in this paper is effective and typical. Let $\mathbb{F}_{q^2}^n$ be the vector space of dimension n over $\mathbb{F}_{q^2}$ with the following hermitian inner product $(,)$ defined by

$$(a,\ b) = \sum_{i=1}^{n} a_i^q b_i \in \mathbb{F}_{q^2} \tag{1}$$

for $a = (a_1, \ldots, a_n)$, $b = (b_1, \ldots, b_n) \in \mathbb{F}_{q^2}^n$. For a $\mathbb{F}_{q^2}$-linear subspace C of $\mathbb{F}_{q^2}^n$, the dual space of C is defined by

$$C^{\perp} = \left\{ a \in \mathbb{F}_{q^2}^n | (a, c) = 0 \text{ for all } c \in C \right\}.$$

Manuscript received January 9, 2002; revised May 19, 2003. The work was performed while K. Feng was visiting the Institute for Mathematical Sciences, National University of Singapore in 2001. The visit was supported by the National Fundamental Sciences Project of China under Grant G19990751, by the IMS, and by a Grant from DSTA of Singapore.

K. Feng is with the Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China (e-mail: kfeng@math.tsinghua.edu.cn).

Z. Ma is with the State Key Laboratory of Information Security, Graduate School of the Chinese Academy of Sciences, Beijing 100039, China (e-mail: mazhi@gscas.ac.cn).

Communicated by W. P. Shor, Associate Editor for Quantum Information Theory.

Digital Object Identifier 10.1109/TIT.2004.838088

Lemma 1.1: Suppose that there exists a $\mathbb{F}_{q^2}$-linear subspace C of $\mathbb{F}_{q^2}^n$ with $\dim_{\mathbb{F}_{q^2}} C = \frac{n-k}{2}$ (so that $2|n-k \geq 0$), and $C \subset C^{\perp}$ (i.e., C is self-orthogonal). Then there exists a quantum code $[[n,\ k,\ d]]_q$ where

$$d = \min\{w_H(c) | c \in C^{\perp} \backslash C\}$$

and $w_H(c)$ is the Hamming weight of c.

This result has been proved in [3] for binary case ($q = 2$) and generalized in [1] to the general case (q is a power of prime number). The quantum codes constructed in this way are called stabilizer quantum codes. If the minimum distance of $C^{\perp}$ is d, the quantum code is called pure. We refer [1], [3], [5] for basic concepts of quantum codes.

There are two bounds which have been established as necessary conditions for quantum codes.

Lemma 1.2 (Quantum Hamming Bound): For any pure stabilizer quantum code $[[n, k, d]]_q$

$$q^{n-k} \geq \sum_{i=0}^{[\frac{d-1}{2}]} (q^2-1)^i \binom{n}{i}.$$

Proof: See [3] for binary case. This can be easily extended to the general case. □

Lemma 1.3 ([5] Quantum Singleton Bound): For any quantum code $[[n, k, d]]_q$, $n \geq k + 2d - 2$.

In this correspondence we present the following bound which is a sufficient condition for the existence of pure stabilizer quantum codes in analogy to the classical Gilbert–Varshamov (GV) bound.

Theorem 1.4: Suppose that $n > k \geq 2$, $d \geq 2$ and $n \equiv k (\mathrm{mod} 2)$. Then there exists a pure stabilizer quantum code $[[n, k, d]]_q$ provided that

$$\frac{q^{n-k+2}-1}{q^2-1} > \sum_{i=1}^{d-1} (q^2-1)^{i-1} \binom{n}{i}. \tag{2}$$

We prove this theorem in Section II. Our proof is similar with the argument in [4, Theorem 1] for the classical case, but we need some enumerative results in finite geometry. In Section III, we make some remarks and, by using Theorem 1.4, present several binary quantum codes with better parameters than the known codes listed in [2].

II. Proof of Theorem 1.4

Let $V = \mathbb{F}_{q^2}^n$ be the vector space of dimension n over $\mathbb{F}_{q^2}$ with the hermitian inner product defined by (1). The unitary group

$$U_n(\mathbb{F}_{q^2}) = \{A = (a_{ij}) \in GL_n(\mathbb{F}_{q^2}) \mid AA^* = I_n\}$$

acts on V where $A^* = \left(a_{ji}^q\right)$. This action keeps the Hermitian inner product and has nice transitive properties (see [6, Ch. 5] for the exact statements on the transitive properties).

Before proving Theorem 1.4 we need two simple enumerative results.

Lemma 2.1: The number of nonzero self-orthogonal vectors in $\mathbb{F}_{q^2}^n$ $(n \geq 1)$ is

$$N_n = (q^n - (-1)^n)(q^{n-1} - (-1)^{n-1}). \tag{3}$$

Proof: For each vector $x = (x_1, \ldots, x_n) \in \mathbb{F}_{q^2}^n$ we know that $x_i^{q+1} \in \mathbb{F}_q$ so that $(x, x) = x_1^{q+1} + \cdots + x_n^{q+1} \in \mathbb{F}_q$. On the other