Upper Bounds on the Number of Errors Corrected by the Koetter–Vardy Algorithm

Justesen, Jørn

Published in:
I E E E Transactions on Information Theory

Link to article, DOI:
10.1109/TIT.2007.901169

Publication date:
2007

Document Version
Publisher's PDF, also known as Version of record

exists $\mathbf{v} \in B_1(\mathbf{y}) \subset B_2(\mathbf{x})$ with $\mathbf{v} \neq \mathbf{u}$ and $E(\mathbf{v})$ odd. Now, since $E(B_2(\mathbf{x}))$ is odd, there must exist $\mathbf{w} \in B_2(\mathbf{x})$ with $E(\mathbf{w})$ odd, $\mathbf{w} \neq \mathbf{u}$ and $\mathbf{w} \neq \mathbf{v}$. Since $E(\mathbf{u})$, $E(\mathbf{v})$ and $E(\mathbf{w})$ are odd, nonnegative integers adding to at most 5, at least two of them must equal one. □

We now may complete the proof of Theorem 1. As stated in [2] or [1] we have $B \subset \bigcup_{\mathbf{x}\in Z_2} B(\mathbf{x})$. Thus by (4) we get

$$B \subset \bigcup_{\mathbf{x}\in Z_2} B(\mathbf{x}) = \bigcup_{\mathbf{x}\in Z_2^{(1)}} B(\mathbf{x}) \cup \bigcup_{\mathbf{y}\in M} \bigcup_{\mathbf{x}\in H(\mathbf{y})} B(\mathbf{x}) \cup \bigcup_{\mathbf{x}\in Z_2^{(3)}\setminus H} B(\mathbf{x})$$

which by (2), Lemma 2, and Lemma 3 imply

$$\begin{aligned} |B| &\leq \sum_{\mathbf{x}\in Z_2^{(1)}} |B(\mathbf{x})| + \sum_{\mathbf{y}\in M} \sum_{\mathbf{x}\in H(\mathbf{y})} |B(\mathbf{x})| + \sum_{\mathbf{x}\in Z_2^{(3)}\setminus H} |B(\mathbf{x})| \\ &\leq \binom{n-2}{2} \left|Z_2^{(1)}\right| + 4|M| \binom{n-3}{2} \\ &\quad + \binom{n-3}{2} \left|\left(Z_2^{(2)} \cup Z_2^{(3)}\right) \setminus H\right| \\ &= \binom{n-2}{2} \left|Z_2^{(1)}\right| + \binom{n-3}{2} \left|Z_2^{(2)} \cup Z_2^{(3)}\right| \\ &= \binom{n-3}{2} |Z_2| + (n-3) \left|Z_2^{(1)}\right|. \end{aligned} \tag{5}$$

Moreover, $E := E(F^n) = \sum_{i\geq 0} i|Z_i| \geq 2|Z_2| + |Z_1|$. Thus

$$2|Z_2| \leq E - |Z_1|. \tag{6}$$

By Lemma 1 a), we have

$$\begin{aligned} \left(1 + n + \binom{n}{2}\right) E &= \sum_{\mathbf{x}\in F^n} E(B_2(\mathbf{x})) \\ &\geq 2|B| + 5|L| + 8(2^n - |B| - |L|) \\ &= 8 \cdot 2^n - 6|B| - 3|L|. \end{aligned}$$

Substituting (5), (3) and (6) on the right-hand side yields

$$\begin{aligned} &\left(1 + n + \binom{n}{2} + 3\binom{n-3}{2}\right) E \\ &\qquad \geq 8 \cdot 2^n + \left(3\binom{n-3}{2} - 2(n-3)\right) |Z_1| - 3|L|. \end{aligned} \tag{7}$$

By Lemma 4 for $n \geq 17$ we have

$$\begin{aligned} &\left(3\binom{n-3}{2} - 2(n-3)\right) |Z_1| \\ &\geq \frac{3}{2}\left(1 + n + \binom{n}{2}\right) |Z_1| \\ &= \frac{3}{2} \sum_{\mathbf{y}\in Z_1} \sum_{\mathbf{x}\in B_2(\mathbf{y})} 1 \geq \frac{3}{2} \sum_{\mathbf{x}\in L} \sum_{\mathbf{y}\in B_2(\mathbf{x})\cap Z_1} 1 \geq 3|L|. \end{aligned}$$

This inserted in (7) yields

$$E \geq \frac{8 \cdot 2^n}{1 + n + \binom{n}{2} + 3\binom{n-3}{2}}$$

and Theorem 1 follows by $E = (n+1)|C| - 2^n$.

ACKNOWLEDGMENT

The author wishes like to thank U. Blass and S. Litsyn for their fine paper [2].

REFERENCES

[1] G. Cohen, I. S. Honkala, S. Litsyn, and A. Lobstein, *Covering Codes*. Amsterdam, The Netherlands: North Holland Mathematical Library, 1997, vol. 54.

[2] U. Blass and S. Litsyn, "Several new lower bounds on the size of codes with covering radius one," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 1998–2002, Sep. 1998.

[3] G. J. M. van Wee, "Improved sphere bounds for the covering radius of codes," *IEEE Trans. Inf. Theory*, vol. 34, no. 2, pp. 237–245, Mar. 1988.

Upper Bounds on the Number of Errors Corrected by the Koetter–Vardy Algorithm

Jørn Justesen, *Member, IEEE*

***Abstract*—By introducing a few simplifying assumptions we derive a simple condition for successful decoding using the Koetter–Vardy algorithm for soft-decision decoding of Reed–Solomon codes. We show that the algorithm has a significant advantage over hard decision decoding when the code rate is low, when two or more sets of received symbols have substantially different reliabilities, or when the number of alternative transmitted symbols is very small.**

***Index Terms*—Reed-Solomon (RS) codes, soft-decision decoding.**

I. INTRODUCTION

In [1] Koetter and Vardy studied an extension of the Sudan–Guruswami algorithm [2] for decoding Reed–Solomon (RS) codes. The interpolating polynomial is required to have certain multiplicities of zeros for several likely symbols in each position. In particular, they analyzed the case where the multiplicities are chosen to be approximately proportional to the conditional probabilities of the symbol values. In general, it is difficult to interpret the condition for successful decoding. In Section III, we derive a much simpler condition for decoding by considering a typical distribution of received symbols and errors. However, in order to describe the decoding problem in this way we have to assume that for each symbol in the code alphabet there is a received symbol that can be considered 'correct', and that the errors can be divided into in a relatively small set of equivalence classes. In Section IV, we obtain bounds on the performance of the Koetter–Vardy (KV) algorithm using these assumptions. In Section V, we consider a few special cases, which lead to very simple bounds. Many channels of interest can be reduced to one of these special cases or to a combination of such cases. Finally, we consider decoding of concatenated codes as a particularly important application of the KV algorithm. We show that the algorithm has a significant advantage over hard-decision decoding when the code rate is low, when two or more sets of received symbols have substantially different reliabilities, or when the correct symbol is on a small list of possible transmitted symbols. However, the performance is far from maximum likelihood.

Manuscript received October 11, 2005; revised September 27, 2006. The material in this correspondence was presented in part at IEEE International Symposium on Information TheoryAdelaide, Australia, September 2005.

The author is with the Department of Communications, Technical University of Denmark, DK 2800 Lyngby, Denmark (e-mail: jju@com.dtu.dk).

Communicated by T. J. Richardson, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2007.901169

II. The Condition for Successful List Decoding

We consider decoding of an (N, K) RS code over the field $\mathrm{F}(q)$. The list decoding algorithm developed in [2] is based on a two-variable interpolating polynomial with the received symbol values and positions as roots of a given multiplicity. In the KV algorithm, the concept is extended by allowing a list of input symbols corresponding to roots of variable multiplicity. When the sum of the multiplicities for each position is upper bounded by a constant, the result of the KV algorithm is a list which is polynomial in N. Decoding is considered to be successful if the transmitted codeword is on the list, and thus all codewords satisfying the condition for successful decoding are found on the list.

In [1], the input to the decoder is a q by N matrix, Π, called the reliability matrix, which has entries

$$\pi_{ij} = P[a_i|y_j]$$

where a_i is a symbol from the code alphabet, A, and y_j is the received symbol, which may belong to a bigger alphabet, B. The multiplicities corresponding to the input lists are entries in the multiplicity matrix, a q by N matrix, $M = [m_{ij}]$. As discussed in [1], the entries in M should be chosen to approximate Π after a suitable normalization.

If the transmitted word is $c = [c(j)]$, where we let $c(j) = i$ indicate that the transmitted symbol is a_i, the condition for successful decoding is

$$\frac{\sum_j m_{c(j),j}}{\sqrt{\sum_{i,j} m_{ij}(m_{ij}+1)}} > \sqrt{K-1} \tag{1}$$

Koetter and Vardy refer to the numerator of (1) as the *score*, and the denominator as the *cost*. They argue that a suitable choice of the multiplicities can be obtained as a matrix of integers which are approximately proportional to the conditional probabilities for the q symbols given the received value. We shall simplify this condition by assuming that the integer entries of M are large enough to allow an accurate approximation to these probabilities, and also neglect smaller terms. Thus, if the sum of the multiplicities is the same in all positions, we can normalize them to obtain a set of weights, w_{ij}, and the condition can be expressed in terms of these weights as

$$\frac{\sum_i w_{c(j),j}}{\sqrt{\sum_{i,j} w_{ij}^2}} > \sqrt{K} \tag{2}$$

In general, the set of weights can be different for each position, and it is difficult to describe the set of received vectors that satisfy (2).

We are interested in the situation where there are few different weights, and the conditional probabilities may be approximated to reduce this number. In particular we assign a weight of 0 to all symbols with low conditional probability, and for this reason the normalized sum of the multiplicities may be less than one. The use of such approximated values is justified by the need to keep the sum of the multiplicities limited, but also by the observation that the performance of the algorithm is rather insensitive to small changes in the weights.

III. Typical Error Patterns and Symmetric Alphabets

In this correspondence, we analyze the situation where the received symbols and the errors can be placed in a small number of equivalence classes, and each type of error is expected to occur several times in a codeword.

On the average each symbol appears $NP[b_j]$ times in a received block, and of these cases the transmitted symbols was a_i in $NP[b_j]P[a_i|b_j]$ instances. Following a standard terminology in information theory we refer to such an error pattern as typical. With this assumption (2) becomes

$$\frac{\sum_{i,j} NP[b_j]w_{ij}^2}{\sqrt{\sum_{i,j} NP[b_j]w_{ij}^2}} > \sqrt{K} \tag{3}$$

or

$$\sum_{i,j} P[b_j]w_{ij}^2 > K/N \tag{4}$$

We use this simplified condition for decoding as a basis for deriving bounds on the performance of the KV algorithm. However, since each symbol on the average appears less than once in a received vector, we have to make some assumptions about the alphabets and the channel.

In the simplest case, the received alphabet equals the code alphabet and all transmitted symbols are equivalent in the sense that they have the same set of distances to other symbols. A set of orthogonal signals on a Gaussian channel (a q-ary symmetric channel) would satisfy this assumption, but in this case the improvement of the KV algorithm disappears with the approximations introduced above. However, other modulation formats like MPSK could be of interest, since transitions to neighboring symbols are much more likely than other types of errors. For modulation formats like QAM, certain symbols have fewer neighbors, but for large alphabets it may be an acceptable approximation to consider only the more common symbols and transitions to the nearest symbols in each direction. In this case the denominator in (2) is a constant, D, and we can collect the terms in the numerator that correspond to errors of the same magnitude. Let r_i indicate the number of errors of type i in a received block and w_i the corresponding weight. The condition (2) then becomes

$$\sum_i r_i w_i > D\sqrt{K} \tag{5}$$

If we move the term corresponding to correctly received symbol to the right side, this relation indicates that a received vector is decoded whenever a linear function of the number of errors of various types does not exceed a given limit. Thus (5) is a generalization of the square-root bound in [2] for the fraction of errors, τ, corrected by the Sudan–Guruswami algorithm

$$\tau < 1 - \sqrt{K/N}$$

Example 1: QAM can serve as an example of a channel with a large alphabet, but only a few types of likely errors. The alphabet is assumed so large that we can neglect the influence of extreme symbols with fewer neighbors, and the receiver uses hard decisions. If p is the probability of error in one dimension, the probabilities of the eight closest neighbors are

$$p - 2p^2 \quad \text{and} \quad p^2$$

We neglect other errors. From (5) we get

$$(N - r_1 - r_2)(1 - 4p + 4p^2) + r_1(p - 2p^2) + r_2p^2 > \sqrt{KN(1 - 8p + 28p^2 - 48p^3 + 36p^4)}.$$

For a typical error pattern we get from (4)

$$1 - 8p + 28p^2 - 48p^3 + 36p^4 > K/N$$

whereas the standard decoding algorithm gives

$$1 - 8p + 8p^2 > K/N$$

Thus for p=1/16 the rate is improved from $17/32$ to $39/64$.

In soft-decision decoding, the receiver alphabet is usually larger than the transmitted alphabet, and thus there are always several types of received symbols. Before proceeding with the more general case, we illustrate the concepts by a simple example:

Example 2: Consider a symmetric channel with the standard concept of errors and erasures. If a symbol is erased, all weights are chosen to be zero. In the remaining positions, the received symbol is assigned a weight of $1-p$, where p is the error probability on the channel, all other symbols have weight zero. Let the number of erasures in a block be e with average value E. For a block with t errors, (2) becomes

$$\frac{(N-e-t)(1-p)}{\sqrt{(N-e)(1-p)^2}} > \sqrt{K}$$

For a typical error pattern, we have from (4)

$$(1-E/N)(1-p)^2 > K/N$$

and we choose the rate to satisfy this condition. We can then find an upper bound on t as a function of e

$$t < N - e - (1-p)\sqrt{(N-E)(N-e)}$$

But we can get a more convenient linear bound by taking the tangent at the point corresponding to typical error patterns

$$t + e(1+p)/2 < Np + E/2 - Ep/2$$

For the same rate, standard errors and erasures correction would give

$$t + e/2 < Np + E/2 - Ep - (N-E)p^2/2$$

For $E/N = 1/4$ and $p = 1/8$ the two relations become

$$\begin{aligned} t \quad &+ \frac{9}{16}e < \frac{15}{64} \\ t \quad &+ e/2 < \frac{109}{512} \end{aligned}$$

indicating that more errors are corrected by the KV algorithm, and the number of errors increases a little faster when there are fewer erasures, as one would expect from the square-root bound.

IV. Bounds on Rate and Correctable Errors

The bound (1) refers to the classical channel model where the transition probabilities are given, the rate of the code is bounded, and the reliability is improved by selecting a somewhat lower rate. However, more realistic channels are described by one or more parameters, such as the S/N ratio, the code is designed for a set of worst-case parameters, and it is assumed that the code will have an improved reliability on a cleaner channel. In this section we derive a bound on the performance of the KV algorithm by converting (4) to a bound on the number of errors that can be corrected. Some additional assumptions are needed for such a bound to apply. The advantage of this approach is that the performance can be described in more intuitive terms and is more easily evaluated.

First we need a concept of a correctly received symbol. For each transmitted symbol we assume that there is a preferred received symbol $a_i' \in B$, such that when $b_j = a_i'$ is received, $P[a_i|b_j]$ is the maximum weight. If a different symbol is received, we refer to that event as an error.

In evaluating the condition (2) it is sufficient to know the weight of the symbol that was transmitted, the set of weights associated with each received symbol, and the number of times each value occurs in the column of M. Thus we may divide the alphabet B into equivalence classes by placing two symbols in the same class whenever the weights, sorted in decreasing order, is the same.

Definition: An error of type (i,j) is the event that the received symbol is of type j and the transmitted symbol appears in position i on the sorted list of weights. If there are several symbols of the same weight on the list, we assume that the transmitted symbol corresponds to the first of these weights.

Let r_j be the number of received symbols in a block that belong to class j. We can then collect the terms in (2) to obtain the square of the denominator

$$\sum_{i,j} r_j w_{ij}^2$$

Let $t_{ij}, i > 0$, be the number of errors of type (i,j), while t_{0j} is the number of correctly received symbols of type j. Note that some received symbols may never be correct, and there is no error or weight of type $(0,j)$. The number of received symbols of each type is

$$r_j = \sum_i t_{ij}$$

and we can write the numerator as

$$\sum_{i,j} t_{ij} w_{ij}$$

For the worst-case channel under consideration, let r_j' and t_{ij}' indicate the expected values of the number of received symbols and errors of each type. We can write (4) as

$$\sum_{i,j} r_j' w_{ij}^2 = \sum_{i,j} t_{ij}' w_{ij} > K$$

and we assume that the rate of the code is chosen close to this limit.

Lemma 1: If (2) is satisfied, the block is also correctly decoded if the r_j are unchanged, but the number of errors of type (i,j), $i > 0$, is less than or equal to t_{ij}

Proof: The denominator is independent of which symbols were actually transmitted, and the numerator is at least as large as in (2).

In general, if a received vector can be decoded, so can any vector where the received symbol belongs to the same class, but the transmitted symbol is changed to the one on top of the list of weights for that class. However, to get a satisfying bound we need to assume that when (2) is satisfied, it will always remain satisfied if a received symbol is replaced by the corresponding "correct" symbol.

For simplicity, assume that all correct symbols are of the same type. Rewriting (2), we get

$$\left(N - \sum_{i,j>0} t_{ij}\right) w_{01} + \sum_{i,j>0} t_{ij} w_{ij} - \sqrt{K}\sqrt{\sum_{ij} r_j w_{ij}^2} > 0$$

To replace this nonlinear expression by a linear bound, we take partial derivatives, $-d_{ij}(r_1, r_2, \cdots)$, with respect to the t_{ij}

$$-w_{01} + w_{ij} - \frac{1}{2}\sqrt{K}\left(\sum_{ij} r_j w_{ij}^2\right)^{-1/2}\left(-\sum_i w_{i1}^2 + \sum_i w_{ij}^2\right)$$

where again the last term disappears when the received symbol is of the same type as the correct symbols. In that case, the expression is negative since the weight of the correct symbol is larger than other weights. However, for a general set of weights there is not necessarily a best symbol since the derivative can change sign.

Example 3: Consider a particular pair of received symbols and a vector which contains several instances of each. If the list of conditional probabilities is $(2/3, 1/3, 0, 0, \cdots)$ for the 'correct' symbol, and $(1-\sqrt{2}/3, 0, 0, \cdots)$ for the other symbol, the derivative is zero for the expected distribution of received symbols.

The assumption that the right side of (2) is monotone when one type of symbol is replaced by the "correct" symbol is usually satisfied for channels of interest. The counter examples appear to require that there are two competing symbols, both of which have rather low reliability, and in such cases the function is extremely flat.

Let the set of numbers r'_j be chosen such that

$$\sum_j r'_j = N$$
$$\sum_{i,j} r'_j w_{ij}^2 > K/N$$

For simplicity let all "correct" symbols belong to the same class, $j = 1$

We then have the following condition for correct decoding of the received block:

Theorem 1: If the number of received symbols of each type is $r_j \leq r'_j$ for $j > 1$, and the number of errors of each type is at most

$$r'_j P[a_i|b_j]$$

for $b_j \neq a'_i$, list decoding by the KV algorithm succeeds.

Proof: Under these assumptions, the left side of (2) is lower bounded by the fraction where r_j is replaced by r'_j. We can then apply Lemma 1.

Under the same assumptions, we have the following theorem.

Theorem 2: For a given channel, (4) is an upper bound on the rate of a code that can be successfully decoded by the KV algorithm.

For long codes the bound is tight, since the distribution of errors is close to the typical distribution. However, for specific channels we might obtain a tighter bound as

$$\sum_{i,j>0} d_{ij}(r'_1, r'_2 \cdots) t_{ij} < D$$

replacing the nonlinear function by a tangent plane in the point corresponding to the typical distribution of received symbols. Example 2 was a simple case of this approach. For such a linearized bound to be a strict lower bound on the number of errors corrected, the second derivatives have to be nonpositive. This is true for the second derivatives with respect to any one of the variables, but as discussed in the case of the first derivative, there could be unusual channels where some mixed derivatives are positive.

We proceed to some cases, which provide more qualitative insight.

V. Some Important Special Cases

If r symbols are erased, we simply assign a weight of 0 to all of them. Clearly one value is correct, but for a large alphabet $1/q$ is too small to make a difference. Thus, from (4), we find

$$N - r > K \quad \text{or} \quad r < N - K$$

which just serves as a check on this approach. If a list of n possibilities is given, $n \ll q$, and each is assigned a probability of $1/n$, we find

$$N - r + \frac{r}{n} > K \quad \text{or} \quad r\frac{n-1}{n} < N - K. \tag{6}$$

We may interpret this result as saying that a list of two values counts as half of an erasure, a list of three as $2/3$, etc. Thus, very small lists offer an advantage compared to erasures, whereas longer lists are of negligible value. From an information theory point of view we would expect the cost of a binary list to be one bit, but the algorithm is far from this limit. The result can be easily extended to include unequal probabilities for the alternatives.

Consider the case where for a set of received symbols the probability of error is known, but other values each have probabilities that are too small to give a significant contribution. If all symbols have error probability p, we get the square-root bound for the Sudan-Guruswami algorithm. If several sets of symbols have different error probabilities, we get from (4)

$$\sum_j r_j (1 - p_j)^2 > K \tag{7}$$

Thus if we find the rates on the square root bound for each of the error probabilities, the rate for a code correcting a mixture of the probabilities is the average of the corresponding rates. Clearly, this means that more errors are corrected than in the case where the average error probability applies to all positions. However, there is only a significant difference if the higher error probabilities are large.

Example 4: If r symbols have a low reliability while the rest are more reliable we have

$$(1 - r/N)(1 - p_1)^2 + r(1 - p_2)^2/N > K/N$$

Let $p_1 = 1/4$, and let $p_2 = 1/2$. With $r = 1/3$ as the design point of the code, we find the rate as $11/21$. Thus there are three types of errors, t reliable positions in error, u unreliable positions with the correct symbol, and v unreliable positions in error. Taking partial derivatives in the design point, we get the linearized bound on the number of correctable errors

$$t + u/8 + 19v/24 < 23/72$$

It may be readily checked that this approximation is actually a lower bound.

Finally we consider the case where errors occur with probability p, but the correct symbol can always be assumed to belong to a set of n equally likely values. This adds a term to the bound

$$(1 - p)^2 + n(p/n)^2 > K/N \tag{8}$$

Clearly this bound approaches the square-root bound for increasing n. Expanding it in negative powers of n, we get the first-order approximation

$$p < 1 - \sqrt{K/N} + (1 - \sqrt{K/N})^2/(2n\sqrt{K/N}).$$

Thus for $K/N = 1/4$ we get $p < \frac{1}{2} + 1/(4n)$. Since there is only one type of received symbol, any number of errors less than pN is corrected.

In [3] Jiang and Narayanan considered RS codes over a field of size 2^m used on a binary erasure or binary symmetric channel. In the first case, there is usually a single erasure in a symbol, and the result can be obtained from (6) with $n = 2$. If a symbol contains a single bit error, there are m possible correct symbols, and we can again find the number of correctable errors from (8).

VI. Concatenated Codes

Concatenated codes with binary inner codes represent an important application where information about the conditional probabilities of the various symbols is available to the RS decoder. We assume throughout that the underlying binary channel is the binary symmetric channel and that the parameters of the inner code are (n, k, d)

As a first case consider the single parity check code $(n, n-1, 2)$. When a symbol has odd parity, it is considered to be an erasure with

n possible values. If the parity check is satisfied, the error probability is closely approximated by the probability of two bit errors, and there are $n(n-1/2)$equally likely error values. The improvement when the KV algorithm is used comes almost entirely from the possibility of correcting $n/(n-1)$ times as many erasures.

If the inner code is a $(n,k,3)$ Hamming code, an error-free symbol has a very low probability of error. Symbols with a corrected bit, on the other hand, are much more likely to be in error. Even though the number of possible error values could be counted, the improvement in the number of errors corrected is negligible. Thus the performance is calculated from (7).

For an inner code of lower rate, codewords with few bit errors have high reliability. When the number of errors approaches $d/2$, it is necessary to distinguish between cases where a second codeword is fairly close and the more common case that there is only a single likely transmitted symbol. When the number of errors exceeds $d/2$, some vectors are close to a codeword different from the one transmitted, while other vectors are far from all codewords.

Example 5: Consider the projective geometry code $(21,12,6)$ for which the necessary details can readily be worked out. Let the average number of bit errors in an inner codeword be 2. It follows from the binomial distribution that the probability of 0 or 1 error in a block is 0.39, and in this case the decision has a high reliability. Two errors are corrected, but the probability of decoding error (if four errors actually occur) is 0.12. The 280 weight 3 error patterns are uniquely decoded, and for simplicity, we merge this set with the double errors. The remaining 1120 weight 3 error patterns are in 380 cosets which gives a list of four possibilities. The remaining errors of weight 4 are treated as erasures, and in our estimate we neglect the contributions from weight 5 errors. In this way, we can apply (4) to get

$$K/N < 0.39 + 0.32(1-0.12)^2 + 0.16/4 = 0.68.$$

This can be compared to standard errors-and-erasures decoding of the RS code where the bound on the rate is 0.63. There is a gain from the small list size of weight 3 errors, and a small gain associated with distinguishing the different reliabilities.

VII. Conclusion

The aim of this correspondence has been to give a more accessible version of the bound on list decoding. Using the simpler expressions it is possible to characterize the errors patterns that are typically decoded by the Koetter–Vardy algorithm.

The cases discussed cover most of the situations that are important for applications. As demonstrated in specific cases, the improvements are significant only for fairly low rates, sets of symbols with large error probability, or a small set of alternatives with high probabilities.

In all cases the performance is still far from maximum likelihood decoding.

References

[1] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed-Solomon codes," *IEEE Trans. Inf. Theory*, vol. IT-49, pp. 2809–2825, Nov. 2003.

[2] V. Guruswami and M. Sudan, "Improved decoding of Reed–Solomon and algebraic-geometric codes," *IEEE Trans. Inf. Theory*, vol. 45, pp. 1757–1767, Sep. 1999.

[3] J. Jiang and K. R. Narayanan, "Performance analysis of algebraic soft decoding of Reed-Solomon codes over binary symmetric and erasure channels," in *Proc. IEEE Int. Symp. Information Theory*, Adelaide, Australia, Sep. 2005, pp. 1186–1190.

Quasi-Cyclic Low-Density Parity-Check Codes With Girth Larger Than 12

Sunghwan Kim, Jong-Seon No, *Member, IEEE*, Habong Chung, *Member, IEEE*, and Dong-Joon Shin, *Member, IEEE*

***Abstract*—A quasi--cyclic (QC) low-density parity-check (LDPC) code can be viewed as the protograph code with circulant permutation matrices (or circulants). In this correspondence, we find all the subgraph patterns of protographs of QC LDPC codes having inevitable cycles of length $2i, i = 6,7,8,9,10$, i.e., the cycles that always exist regardless of the shift values of circulants. It is also derived that if the girth of the protograph is $2g, g \geq 2$, its protograph code cannot have the inevitable cycles of length smaller than $6g$. Based on these subgraph patterns, we propose new combinatorial construction methods of the protographs, whose protograph codes can have girth larger than or equal to 14 or 18. We also propose a couple of shift value assigning rules for circulants of a QC LDPC code guaranteeing the girth 14.**

***Index Terms*—Girth, low-density parity-check (LDPC) codes, protograph, protograph codes, quasi--cyclic (QC) codes.**

I. Introduction

Since the low-density parity-check (LDPC) code exhibits the capacity-approaching performance for many channels such as binary erasure channel (BEC), binary symmetric channel (BSC), and additive white Gaussian noise (AWGN) channel, it has been one of the major research topics for many coding theorists at least for the last decade. It is known that the message-passing decoder of LDPC codes is relatively easy to implement due to the sparseness of the parity-check matrix, but the encoding complexity of LDPC codes is quite high. Thus, many researchers have been working on designing efficiently encodable LDPC codes.

Although the random construction shows good asymptotic performance, its randomness hinders the ease of analysis and implementation. In an effort toward the algebraic constructions of LDPC codes, a quasi--cyclic (QC) LDPC code is getting more attention due to its linear-time encodability and small size of required memory.

A (J,L) regular LDPC code is defined in terms of a parity-check matrix H in which each column contains J 1's and each row contains L 1's. Originally, a QC LDPC code is defined as a (J,L) regular LDPC code of length Lp whose parity-check matrix H is a $J \times L$ array of $p \times p$ circulant permutation matrices (shortly, circulants) [1]. Fossorier derived the necessary and sufficient condition for the existence of cycles of given length in QC LDPC codes. Fossorier [1] and Tanner [2] also showed that these QC LDPC codes have a girth at most 12.

Manuscript received May 12, 2005; revised December 2, 2006. This work was supported by the Information Technology Research Center (ITRC) Program and the Information and Telecommunication National Scholarship Program of the Korean Ministry of Information and Communications. The material in this correspondence was presented in part at the IEEE International Symposium on Information Theory, Nice, France, June 2007.

S. Kim and J.-S. No are with the School of Electrical Engineering and Computer Science, Seoul National University, Seoul 151-744, Korea (e-mail: nodoubt@ccl.snu.ac.kr; jsno@snu.ac.kr)).

H. Chung is with the School of Electronics and Electrical Engineering, Hong-Ik University, Seoul 121-791, Korea (e-mail: habchung@wow.hongik.ac.kr).

D.-J. Shin is with the Division of Electrical and Computer Engineering, Hanyang University, Seoul 133-791, Korea (e-mail: djshin@hanyang.ac.kr).

Communicated by T. J. Richardson, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2007.901193