Upper Bound on the Minimum Distance of Turbo Codes Using a Combinatorial Approach

Marco Breiling and Johannes Huber

Lehrstuhl für Nachrichtentechnik II, Cauerstr. 7, D-91058 Erlangen, Germany
Tel.: +49-9131-85-27668, Fax: +49-9131-85-28919
E-mail: breiling@LNT.de, http://www.LNT.de/∼breiling

Abstract: *By using combinatorial considerations, we derive new upper bounds on the minimum Hamming distance, which Turbo codes can maximally attain with arbitrary — including the best — interleavers. The new bounds prove that by contrast to general linear binary channel codes, the minimum Hamming distance of Turbo codes cannot asymptotically grow stronger than the third root of the codeword length.*

Keywords: Turbo code, minimum Hamming distance bounds, interleaver design

1. INTRODUCTION

Since Turbo codes were developped by a heuristic approach, very little is still known about their code structure, e.g. which limits exist for their minimum Hamming distance $\delta_{\min}$. Soon after the invention of Turbo codes, it was discovered that for practically every *randomly* chosen interleaver $\delta_{\min}$ is very low and does not even grow with the interleaver length K, such that an *error–floor* of the bit error rate curve occurs. To solve this problem, several algorithms were proposed (e.g. [1]) for increasing $\delta_{\min}$ by *designing* the interleaver. The so–attained $\delta_{\min}$ depends on K and the employed algorithm, but nothing is known about the *maximum* attainable $\delta_{\min}$ for a given K. The authors know only few papers giving bounds on $\delta_{\min}$ of Turbo codes: E.g. in [2], it is shown that for *random* interleavers, $\delta_{\min}$ does not grow with K with asymptotically probability 1. The paper [3] states an upper bound on $\delta_{\min}$ for a *given* interleaver. To derive this bound, only Turbo encoder input words of weight 2 are taken into account.

In the present paper, new upper bounds on the maximum attainable $\delta_{\min}$ for *all* (including the *best*) interleavers of a given length K are derived by making combinatorial considerations for input words of weight 2 as well as 4. Thus, the new upper bounds can be used as a benchmark for assessing the performance of interleaver design algorithms, and to point out the limits of Turbo codes. The paper is organized as follows: In Section 2, the considered Turbo encoder model is presented. The derivation of the new bounds is performed in Section 3, and we conclude by discussing their implications in Section 4.

2. SYSTEM MODEL

The Turbo codes considered here are binary parallel concatenated *convolutional* codes as shown in Fig. 1: The Turbo encoder input word $\boldsymbol{u} = (u_0; ..; u_{K-1})$ of length K is propagated to three parallel branches. The upper branch composes the systematic part of the Turbo codeword $\boldsymbol{c}$. The middle and the lower branches produce the two parity words $\boldsymbol{c}^{(1)}$ and $\boldsymbol{c}^{(2)}$. We refer to these branches as the 1st and 2nd *component*, respectively, and all quantities $\bullet$ associated with the i–th component ($i \in \{1; 2\}$) will be represented by $\bullet^{(i)}$. The Turbo codeword consists hence of the three parts $\boldsymbol{c} = (\boldsymbol{u}; \boldsymbol{c}^{(1)}; \boldsymbol{c}^{(2)})$.

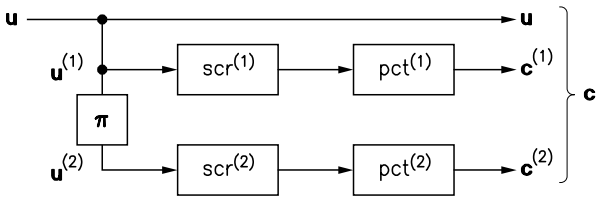

Figure 1: System model of the Turbo encoder

In the 1st component, the input word $\boldsymbol{u}^{(1)} = \boldsymbol{u}$ is fed into a scrambler ("scr" in the Fig.), which is a binary recursive convolutional encoder of rate 1. The output of the scrambler is then possibly punctured ("pct") in order to increase the rate of the Turbo code adequatly. The proceeding in the 2nd component is similar, however not $\boldsymbol{u}$ but an *interleaved* input word $\boldsymbol{u}^{(2)}$ is scrambled and punctured. The interleaver, which is described by a transposition vector $\boldsymbol{\pi} = (\pi_0; ..; \pi_{K-1})$, performs a permutation of the elements of $\boldsymbol{u}$, such that $u^{(2)}_{\pi_i} = u_i,\ i = 0; ..; K-1$. Note that for our encoder model, the interleaver length is identical to the input word length K. We will use the notations $\boldsymbol{u} = (u_0; ..; u_{K-1})$ and the D–transform $u(D) = \bigoplus\!\!\!\!\!\!\sum_{i=0}^{K-1} u_i \cdot D^i$ for vectors. Here $\oplus$ and $\bigoplus\!\!\!\!\!\!\sum$ represent the addition in GF(2).

In this paper we restrict ourselves without loss of generality to the case of identical scramblers and puncturers in both components. A generalization of the derived bounds to differing scramblers and/or puncturers is however straightforward. Since we do

not rule out the case that the input word $\boldsymbol{u}$ contains (besides information symbols) appropriate redundancy symbols in order to terminate one (or both) component scrambler(s) in the zero–state at the end of every input word, we use the term *input* word rather than *information* word. However, the bounds established and the conclusions drawn in this paper are valid for the cases that no, only one or both scramblers are terminated in the zero-state by $\boldsymbol{u}$.

3. DERIVATION OF THE BOUNDS

As Turbo codes are linear, we can find $\delta_{\min}$ by determining the minimum weight of all non–zero codewords. Since the Turbo codeword $\boldsymbol{c}$ is composed of the three parts $(\boldsymbol{u}; \boldsymbol{c}^{(1)}; \boldsymbol{c}^{(2)})$, its weight can be split into three terms:

$$w(\boldsymbol{c}) = w(\boldsymbol{u}) + w(\boldsymbol{c}^{(1)}) + w(\boldsymbol{c}^{(2)}), \tag{1}$$

In this paper, we are going to consider only input words $\boldsymbol{u}$ of weight 2 or 4. To start with the simple, let us first examine the parity weight produced by a *single* component.

If the component scrambler's period is denoted by p, then any input sequence "1 $0^{\kappa \cdot p-1}$ 1", $\kappa \in \mathbb{N}^+$, i.e. two "1"s spaced a multiple of p positions apart from each other, leads to an *error event*, i.e. a deviation from the zero–path in the scrambler trellis, of length $\kappa \cdot p+1$ (cf. [4]). This leads to the following consequence:

Lemma 1 *For a j–th component ($j \in \{1;2\}$) input word $u^{(j)}(D) = D^{i_1} \oplus D^{i_2}$ with $i_1; i_2 \in \{0; ..; K-1\}$ and $|i_2 - i_1| = \kappa \cdot p$, $\kappa \in \mathbb{N}^+$, the weight of the generated component parity word $\boldsymbol{c}^{(j)}$ is upper bounded by*

$$w(\boldsymbol{c}^{(j)}) \leq \alpha \cdot \kappa + \beta \tag{2}$$

for appropriately chosen constants $\alpha; \beta \in \mathbb{N} \cup \{0\}$, which depend only on the employed component scrambler and puncturer. The proof is straightforward: Since the error event is of length $\kappa \cdot p + 1$, the associated parity weight cannot exceed $\kappa \cdot p + 1$, and we obtain the upper bound $w(\boldsymbol{c}^{(j)}) \leq \kappa \cdot p + 1$ by setting $\alpha = p$ and $\beta = 1$. ■

An example for the smallest possible α, β for the case of no puncturing are $\alpha = 2$, $\beta = 2$ for a memory $\nu = 2$ scrambler with generator polynomial (in octal) 5/7 (period $p = 3$). Further examples are $\alpha = 4$, $\beta = 2$ for the $\nu = 3$ scrambler 17/13 ($p = 7$), and $\alpha = 8$, $\beta = 2$ for the scrambler with memory $\nu = 4$ and generator polynomial 35/23 ($p = 15$). We see from Lemma 1 that the component parity weight is low, if the component input word contains *two* "1" elements, whose indices $i_1; i_2$ are in the same equivalence class with respect to the modulo p–operation: $i_1 \bmod p = i_2 \bmod p$, and whose index difference $|i_2 - i_1|$ is small.

In order to exploit these two observations, we partition the set of indices $\{0; ..; K-1\}$ of the j–th component input word $\boldsymbol{u}^{(j)}$, $j \in \{1;2\}$ into $\mu^{(j)}$ disjoint subsets $\mathcal{I}_l^{(j)}$, $l = 0; ..; \mu^{(j)} - 1$, where $\mu^{(j)}$ must be a multiple of p. We choose the partition such that the first $\eta^{(j)} \triangleq K \bmod \mu^{(j)}$ subsets $\mathcal{I}_l^{(j)}$, $l = 0; ..; \eta^{(j)} - 1$ have cardinality $\|\mathcal{I}_l^{(j)}\| = \lceil K/\mu^{(j)} \rceil$, where $\lceil x \rceil$ is the smallest integer $\geq x$. The remaining $\mu^{(j)} - \eta^{(j)}$ subsets have cardinality $\lambda^{(j)} \triangleq \lfloor K/\mu^{(j)} \rfloor$, where $\lfloor x \rfloor$ represents the largest integer $\leq x$. Each subset is composed of *consecutive* indices belonging to an *identical* equivalence class ($\bmod p$) as follows:

$$\mathcal{I}_l^{(j)} = \begin{cases} \Big\{\iota_l^{(j)}; \iota_l^{(j)} + p; \iota_l^{(j)} + 2p; .. \\ \qquad ..; \iota_l^{(j)} + (\lambda^{(j)} - 1) \cdot p; \iota_l^{(j)} + \lambda^{(j)} \cdot p\Big\} \\ \qquad \text{for } 0 \leq l < \eta^{(j)} \\ \Big\{\iota_l^{(j)}; \iota_l^{(j)} + p; \iota_l^{(j)} + 2p; ..; \iota_l^{(j)} + (\lambda^{(j)} - 1) \cdot p\Big\} \\ \qquad \text{for } \eta^{(j)} \leq l < \mu^{(j)}, \end{cases} \tag{3}$$

where the minimum element $\iota_l^{(j)} = \min \mathcal{I}_l^{(j)}$ of any subset is recursively defined as follows:

$$\iota_l^{(j)} \triangleq \begin{cases} l & \text{for } 0 \leq l < p \\ \iota_{l-p}^{(j)} + p \cdot (\lambda^{(j)} + 1) & \text{for } 0 \leq l - p < \eta^{(j)} \\ \iota_{l-p}^{(j)} + p \cdot \lambda^{(j)} & \text{for } \eta^{(j)} \leq l - p < \mu^{(j)} - p, \end{cases} \tag{4}$$

From Eq. (3), we can calculate the maximum difference between any two indices belonging to an identical subset

$$\max_{l=0;..;\mu^{(j)}-1} \Big\{ \max_{\substack{i_1, i_2 \in \mathcal{I}_l^{(j)} \\ i_1 \neq i_2}} \{|i_1 - i_2|\} \Big\} = \left(\left\lceil \frac{K}{\mu^{(j)}} \right\rceil - 1 \right) \cdot p. \tag{5}$$

We must impose $\lambda^{(j)} \geq 2$, which is equivalent to $\mu^{(j)} \leq K/2$, to ensure that each subset contains at least two indices and that Eq. (5) can be applied. Combining Eq. (5) with Lemma 1, we find that for a component input word $u^{(j)}(D) = D^{i_1} \oplus D^{i_2}$, $j \in \{1;2\}$, whose two "1" elements belong to an (arbitrary) identical index subset $i_1, i_2 \in \mathcal{I}_l^{(j)}$, the associated component parity weight is upper bounded as follows:

$$w(\boldsymbol{c}^{(j)}) \leq \alpha \cdot \left(\left\lceil \frac{K}{\mu^{(j)}} \right\rceil - 1 \right) + \beta. \tag{6}$$

For deriving our bounds, we will show that there exists an input word $\boldsymbol{u}$ for *any* given interleaver $\boldsymbol{\pi}$, for which we can use Eq. (6) for $w(\boldsymbol{c}^{(1)})$ *and* for $w(\boldsymbol{c}^{(2)})$. Our first upper bound can now be derived by examining the 1st and the 2nd component simultanously:

Theorem 1 *The minimum distance* $\delta_{\min}(K;\alpha;\beta;p)$ *of a parallel concatenated convolutional code with parameters* α, β *and* p *describing the component scramblers/puncturers as stated in Lemma 1, is for* any *interleaver of length* K *upper bounded by*

$$\delta_{\min} \leq 4+2\alpha\cdot\left(\min_{(\mu^{(1)};\mu^{(2)})\in\mathcal{M}}\left\{\left\lceil\frac{K}{\mu^{(1)}}\right\rceil+\left\lceil\frac{K}{\mu^{(2)}}\right\rceil\right\}-2\right)+4\beta, \tag{7}$$

where the minimization is performed over the set $\mathcal{M}$ *of pairs* $\{(\mu^{(1)};\mu^{(2)})\}\subset\{1;..;K/2\}^2$, *for which both of the following conditions are true*

(1) $\mu^{(1)} \bmod p = \mu^{(2)} \bmod p = 0$

(2) $\mu^{(1)}\binom{\lambda^{(1)}}{2} + \lambda^{(1)}\cdot\eta^{(1)} > \binom{\mu^{(2)}}{2}$,

where $\lambda^{(1)} = \lfloor K/\mu^{(1)}\rfloor$ *and* $\eta^{(1)} = K \bmod \mu^{(1)}$.

Proof: Let us choose an arbitrary pair $(\mu^{(1)};\mu^{(2)})\in\mathcal{M}$ and partition the indices of $\boldsymbol{u}^{(1)}$ and $\boldsymbol{u}^{(2)}$ into $\mu^{(1)}$ and $\mu^{(2)}$ subsets according to the above partition strategy, respectively. Condition *(1)* and the condition $\mu^{(1)},\mu^{(2)} < K/2$ ensure that there actually exists such a partitioning. We will use $\mathcal{I}^{(2)}(j)$, $j\in\{0;..;K-1\}$ to represent the subset $\mathcal{I}^{(2)}_m$ containing the index j of the 2nd component.

Let us consider an *arbitrary* given interleaver $\boldsymbol{\pi}$ of length K. For this $\boldsymbol{\pi}$ and the considered $(\mu^{(1)};\mu^{(2)})$, two cases are possible:

Case 1: There exists a subset $\mathcal{I}^{(1)}_{l_0}$ of the 1st component, of which at least two indices $i_1,i_2\in\mathcal{I}^{(1)}_{l_0}$ are mapped by $\boldsymbol{\pi}$ to an *identical* associated subset $\mathcal{I}^{(2)}(\pi_{i_1}) = \mathcal{I}^{(2)}(\pi_{i_2})$ of the 2nd component.

We can hence use Eq. (6) to upper bound the component parity weight associated with $u^{(1)}(D) = D^{i_1}\oplus D^{i_2}$ as well as $u^{(2)}(D) = D^{\pi_{i_1}}\oplus D^{\pi_{i_2}}$. There exists hence a Turbo encoder input word $\boldsymbol{u}=\boldsymbol{u}^{(1)}$ of weight 2, whose codeword weight can be upper bounded by combining Eq. (6) with Eq. (1) such that:

$$\delta_{\min} \leq 2+\alpha\cdot\left(\left\lceil\frac{K}{\mu^{(1)}}\right\rceil-1\right)+\beta+\alpha\cdot\left(\left\lceil\frac{K}{\mu^{(2)}}\right\rceil-1\right)+\beta. \tag{8}$$

End of Case 1.

Case 2: For *every* subset $\mathcal{I}^{(1)}_l, l\in\{0;..;\mu^{(1)}-1\}$ of the 1st component, *all* indices of the subset are mapped by $\boldsymbol{\pi}$ to $\|\mathcal{I}^{(1)}_l\|$ *distinct* subsets $\mathcal{I}^{(2)}_m$ of the 2nd component.

Consider an arbitrary $\mathcal{I}^{(1)}_l$, $l\in\{0;..;\mu^{(1)}-1\}$ and a set $\{i_1;i_2\}\subset\mathcal{I}^{(1)}_l$ of *two distinct* indices $i_1\neq i_2$ taken from $\mathcal{I}^{(1)}_l$. For the present Case 2, these two indices are associated with two distinct subsets $\mathcal{I}^{(2)}(\pi_{i_1})\neq\mathcal{I}^{(2)}(\pi_{i_2})$. Let us construct a set composed of these two subsets: $\mathcal{E}_l=\{\mathcal{I}^{(2)}(\pi_{i_1});\mathcal{I}^{(2)}(\pi_{i_2})\}$ for $\{i_1;i_2\}\subset\mathcal{I}^{(1)}_l$. As there are $\binom{\|\mathcal{I}^{(1)}_l\|}{2}$ distinct sets $\{i_1;i_2\}$ of two elements taken from $\mathcal{I}^{(1)}_l$, we can construct $\binom{\|\mathcal{I}^{(1)}_l\|}{2}$ *distinct* sets $\mathcal{E}_{l;j}$, $j\in\{1;..;\binom{\|\mathcal{I}^{(1)}_l\|}{2}\}$ of two associated (2nd component) index subsets for *any* $\mathcal{I}^{(1)}_l$.

Taking into account *all* $l\in\{0;..;\mu^{(1)}-1\}$, we can construct a total of $\sum_{l=0}^{\mu^{(1)}-1}\binom{\|\mathcal{I}^{(1)}_l\|}{2} = \eta^{(1)}\cdot\binom{\lambda^{(1)}+1}{2} + (\mu^{(1)}-\eta^{(1)})\cdot\binom{\lambda^{(1)}}{2}$ sets $\mathcal{E}_{l;j}$. However, since every $\mathcal{E}_{l;j}\subset\{\mathcal{I}^{(2)}_0;..;\mathcal{I}^{(2)}_{\mu^{(2)}-1}\}$ and $\|\mathcal{E}_{l;j}\|=2$, there exist at most $\binom{\mu^{(2)}}{2}$ *distinct* sets $\mathcal{E}_{l;j}$. Condition *(2)* implies thus that among all the constructed $\mathcal{E}_{l;j}$, there exist two *identical* sets $\mathcal{E}_{l_1;j_1}$ and $\mathcal{E}_{l_2;j_2}=\mathcal{E}_{l_1;j_1}$ with $l_1\neq l_2$. This statement is equivalent to the following: There exists two subsets $\mathcal{I}^{(1)}_{l_1}$ and $\mathcal{I}^{(1)}_{l_2}\neq\mathcal{I}^{(1)}_{l_1}$, for which there exists an index pair $(i_{1;1};i_{1;2})$ with $i_{1;1},i_{1;2}\in\mathcal{I}^{(1)}_{l_1}$, $i_{1;1}\neq i_{1;2}$ and a pair $(i_{2;1};i_{2;2})$ with $i_{2;1},i_{2;2}\in\mathcal{I}^{(1)}_{l_2}$, $i_{2;1}\neq i_{2;2}$ with the following properties: $\mathcal{I}^{(2)}(\pi_{i_{1;1}}) = \mathcal{I}^{(2)}(\pi_{i_{2;1}})$ and $\mathcal{I}^{(2)}(\pi_{i_{1;2}}) = \mathcal{I}^{(2)}(\pi_{i_{2;2}})$.

Let us consider a 1st component input word $u^{(1)}(D) = D^{i_{1;1}}\oplus D^{i_{1;2}}\oplus D^{i_{2;1}}\oplus D^{i_{2;2}}$ of weight 4. Since $i_{1;1},i_{1;2}$ belong to an identical subset $\mathcal{I}^{(1)}_{l_1}$ and $i_{2;1},i_{2;2}$ belong to an identical subset $\mathcal{I}^{(1)}_{l_2}$, the associated 1st component parity weight can be upper bounded by (cf. Eq. (6))

$$w(\boldsymbol{c}^{(1)}) \leq 2\alpha\cdot\left(\left\lceil\frac{K}{\mu^{(1)}}\right\rceil-1\right)+2\beta. \tag{9}$$

$w(\boldsymbol{c}^{(2)})$ can be upper bounded in the same way, since the indices of the four "1" elements in $\boldsymbol{u}^{(2)}$, which corresponds to $\boldsymbol{u}^{(1)}$, form two pairs, where both indices of a pair belong to an identical subset, respectively. Together with Eq. (1), we obtain the following upper bound (cf. Case 1):

$$\delta_{\min} \leq 4+2\alpha\cdot\left(\left\lceil\frac{K}{\mu^{(1)}}\right\rceil-1\right)+2\beta+2\alpha\cdot\left(\left\lceil\frac{K}{\mu^{(2)}}\right\rceil-1\right)+2\beta. \tag{10}$$

End of Case 2.

Eq. (10) is valid for Case 1, too, since it upper bounds Eq. (8). Observe furthermore that Eq. (10) does *not* depend on the specific considered interleaver $\boldsymbol{\pi}$ and is hence valid for *all* interleavers of the given length K. Minimizing the upper bound of Eq. (10) over all $(\mu^{(1)};\mu^{(2)})\in\mathcal{M}$, we obtain Eq. (7). ■

By separating the minimization over $\mu^{(1)}$ from that over $\mu^{(2)}$ in Theorem 1, we can derive the following *equivalent*

Theorem 2 *An upper bound on* $\delta_{\min}(K;\alpha;\beta;p)$ *is obtained by minimizing in the following Eq. (11) over those* $\mu^{(1)}\in\{1;..;K/2\}$, *for which* $\mu^{(1)} \bmod p = 0$ *and* $\gamma \triangleq 1/2+\sqrt{1/4+2\left(\mu^{(1)}\binom{\lambda^{(1)}}{2}+\lambda^{(1)}\cdot\eta^{(1)}\right)} > p$ *is satisfied:*

$$\delta_{\min} \le 4 + 2\alpha \cdot \left(\min_{\mu^{(1)}} \left\{ \left\lceil \frac{K}{\mu^{(1)}} \right\rceil + \right.\right.$$
$$\left.\left. + \left\lceil \frac{K}{p \cdot \left(\left\lceil \frac{\frac{1}{2} + \sqrt{\frac{1}{4} + 2\left(\mu^{(1)}\binom{\lambda^{(1)}}{2} + \lambda^{(1)} \cdot \eta^{(1)}\right)}}{p} \right\rceil - 1 \right)} \right\rceil \right\} - 2 \right) + 4\beta. \quad (11)$$

It can be shown that there exists at least one valid $\mu^{(1)}$ satisfying $\gamma > p$, if $K \ge p \cdot (\lfloor 1/2 \cdot (1 + \sqrt{4p-3}) \rfloor + 1)$. By setting $\mu^{(1)} = \mu^{(2)} = p \cdot \lfloor ((K-1)^{2/3} - (K-1)^{1/3} + 1)/p \rfloor$ in Eq. (10) and using simple upper bounding techniques for the resulting Eq., we obtain a third bound:

Theorem 3 *$\delta_{\min}(K;\alpha;\beta;p)$ is upper bounded by*

$$\delta_{\min} < 4 + 4\alpha \cdot \frac{K}{(K-1)^{2/3} - (K-1)^{1/3} + 1 - p} + 4\beta. \quad (12)$$

4. DISCUSSION AND CONCLUSION

Fig. 2 shows existing and the new upper bounds $\delta_{\min,\mathrm{UB}}$ on the minimum distance $\delta_{\min}$ for Turbo codes of rate 1/3 and varying input word length K. The existing bounds are the Gilbert–Varshamov–bound ("GV" in the Fig.), the Hamming–bound and the (asymptotic) McEliece et al.–bound for general linear binary channel codes (GLBCC), which all grow approx. linearly with K (note that the codeword length is $3K$). Furthermore we have plotted the new bounds for Turbo codes employing component scramblers of memory $\nu = 2$ (generator polynomial in octal: 5/7, α, β–parameters cf. Section 3), $\nu = 3$ (17/13) and $\nu = 4$ (35/23). For each $\nu \in \{2;3;4\}$, Theorem 1/2 provides the lower, whereas Theorem 3 yields the upper of the two upper bounds, since the latter was derived by upper bounding Theorem 1. We see that the bounds of Theorem 3 are very loose for small K and become tighter for $K \to \infty$. The rapid fluctuations in the bounds of Theorem 1/2 are due to quantization effects caused by the discontinuous upwards rounding function in these bounds.

Obviously, the derived upper bounds grow when the parameter α is increased. The bounds reflect therefore the fact that component scramblers of larger memory and with a primitive feedback–polynomial (i.e. a maximum period) should be chosen in order to increase $\delta_{\min}$ and lower the error–floor. However, for any scrambler choice, the new upper bounds rapidly diverge from the existing ones for GLBCC (applying for e.g. Shannon's random codes), which grow approx. linearly with K. The reason is that according to Theorem 3, $\delta_{\min}$ of Turbo codes cannot asymptotically grow stronger than $4\alpha\sqrt[3]{K}$. We recognize that *all* Turbo codes are asymptotically bad, i.e. even if the best possible interleavers are employed. Turbo codes are hence not exactly the "Shannon codes" as what they are sometimes refered to, as regards to the *word error rate* (WER), although their *bit error rate* (BER) performance is very close to what we can expect from "Shannon codes". Battail introduced the term *weakly random–like* [5] for this type of codes exhibiting a bad WER performance but a good performance as regards to the BER. The new bounds prove therefore that Turbo codes are only weakly random–like — even for the best interleaver.

From the derivation of Theorem 1 we recognize furthermore that there are always input words of weight 2 *or* 4 generating low weight codewords. Most interleaver design algorithms try to avoid only the case that a weight 2–input word generates a low codeword weight. We see that weight 4–input words have to be considered when maximizing $\delta_{\min}$, too.

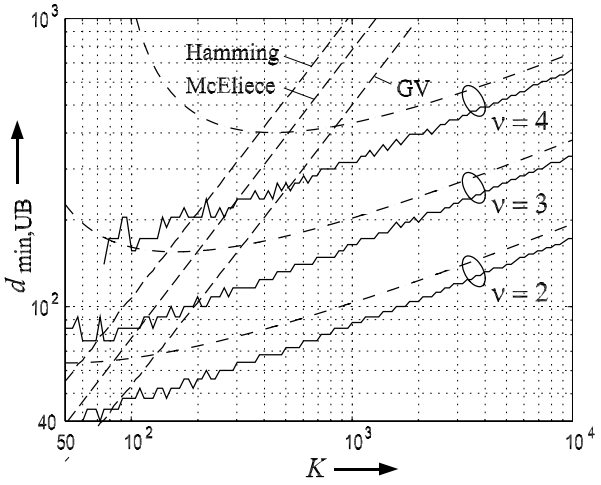

Figure 2: Bounds for (Turbo) codes of rate 1/3

REFERENCES

[1] Kenneth Andrews, Chris Heegard, and Dexter Kozen. Interleaver design methods for turbo codes. *Intern. Symp. on Inf. Th.*, page 420, 1998.

[2] Nabil Kahale and Rüdiger Urbanke. On the minimum distance of parallel and serially concatenated codes. `lcavwww.epfl.ch/~ruediger/publications.html`, 1997.

[3] W. Blackert, E. Hall, and S. Wilson. An upper bound on turbo code free distance. *Intern. Conf. on Comm.*, pages 957–961, 1996.

[4] Solomon Golomb. *Shift Register Sequences*. Aegean Park Press, revised edition, 1982.

[5] Gérard Battail. On random-like codes. *Canadian Workshop on Inf. Th.*, 1995.