SAMPLED CODING - DISPERSION OF RELIABILITY BY ITERATIVE DECODING

Jozef Hámorský and Johannes B. Huber
hamorsky@nt.e-technik.uni-erlangen.de huber@nt.e-technik.uni-erlangen.de
Lehrstuhl für Nachrichtentechnik
Universität Erlangen-Nürnberg
Cauerstraße 7, D-91058 Erlangen, Germany

ABSTRACT
A new approach for concatenation of codes is proposed. Only a part of data symbols is fed to an outer encoder whereas the remaining part is directly forwarded to an inner encoder. (For example the inner code can be intersymbol interference causing process.) Thus, only samples of data are encoded by the concatenated structure. During an iterative decoding procedure is improved reliability not only of the twice coded data, but also of the once coded data. An estimation of the performance of sampled coding scheme is derived and compared with simulation results. Several examples have been found where all data symbols achieve higher reliability in a sampled coding approach than in a conventional concatenated scheme with equal (average) rate. Additionally, there are two (or more) classes of symbols. The first one shows extreme low error rate for the concatenated encoded symbols. The second one shows error rates that are significantly lower compared to a scheme employing only the inner code. This effect can very efficiently be exploited by an appropriate combining of source and channel coding.

1. INTRODUCTION
Ensuring high reliability of transmitted data is one of the most important tasks of digital transmission systems. A very important tool to reach this goal is channel coding based on adding redundancy. There are a lot of important parameters of channel codes which characterize channels codes. Two of the most important are

- code rate \( R_c \), i.e. (for binary codes) the ratio of informations bits to the total number of transmitted bits
- complexity of the encoding and decoding procedure.

Codes should be used with redundancy as little as possible to guarantee necessary reliability of data. The more redundancy, the greater transmission speed and the more signal bandwidth are necessary, but also the better protection of data is achieved.

A popular method to achieve long powerful codes at a moderate complexity is concatenation of codes. Concatenation can be done in a serial and a parallel way (e.g. Turbo-codes). Here, we focus on serial concatenation. Each of the component codes of a concatenated scheme can be very simple, but together they provide a powerful protection of the data. At the receiver side the codes are decoded successively, hence the complexity grows only linear with the number of codes used. Interleaving between the encoders usually provide additional performance gain.

Up to now, the most powerful coding schemes known are concatenated codes, which are decoded iteratively\(^1\). Iterative decoding improves reliability of data due to repeated decoding using the results of previous decoding steps\(^1\). Such decoding procedure requires “soft-in/soft-out” symbol by symbol decoding algorithms, i.e. algorithms, which utilize reliability information of the input data and provide reliability information for each individual output symbol\(^2\). The information from previous decoding step, called extrinsic information (EI) is usually processed as a-priori probability of data in the subsequent decoding step. In the concatenated coding schemes EI is the a-posteriori probability of the decoded data without taking EI from the previous decoder into account \([3, 4, 14]\).

In many cases iterative decoding can achieve a reasonable gain in comparison to standard decoding methods. For example in [3] a bit error rate (BER) of \(10^{-5}\) is achieved at a signal-to-noise ratio (SNR) of 0.7 dB using binary signaling, which is more than 4 dB better when compared to noniterative decoding of comparable complexity. Iterative decoding was originally developed for systematic parallel concatenated convolutional codes [3]. Recently, iterative decoding was extended to serial concatenated and nonsystematic codes [4, 2]. For high effectiveness of iterative decoding interleaving is essential.

\( ^1 \)Decoding step denotes a single decoding of one component code. One complete decoding of all component codes is called decoding period (DP).

\( ^2 \)For linear codes, an optimal soft-in/soft-out algorithm is the so called BCJR algorithm, see [1].
In this paper we introduce a new coding scheme, in which only part of the source data are fed to the outer and the inner encoder of a serially concatenated coding scheme whereas the remaining part is only encoded by the inner encoder. Because only samples of data are encoded twice, we refer to this method as *sampled coding*. For a fixed overall code rate, the rate of the outer code is much lower than the ones of conventional concatenated coding schemes. Thus, those data which are encoded twice, achieve a higher reliability than in the conventional approach. The key point of sampled coding is to transfer this high reliability to the once coded data via the iterative decoding process. The twice coded data form points of support via strong a-priori probabilities during a second (and following) pass of the inner decoder. Thus, reliability of data not encoded twice is improved by the outer code in an indirect — dispersive way. In this paper we focus our attention on coded transmission over inter-symbol interference (ISI) channels. Thus, the channel itself form the inner encoder. After establishing the system model (Section 2), a simple estimation of error rate in sampled coded transmission (SCT) is derived in Section 3. The paper concludes in presenting simulation results of the proposed schemes (Section 4).

### 2. SYSTEM MODEL

Consider convolutionally coded binary data transmission over intersymbol interference producing discrete time channel with finite impulse response (FIR) and additive white Gaussian noise (AWGN), see Fig. 1. The discrete time channel model depicted in Fig. 2 is derived from the continuous time channel using Forney’s whitened-matched filter and T-spaced sampling [6]. It consists of a shift register containing $\nu$ most recent channel inputs $a_{\mu - 1}, a_{\mu - 2}, \ldots, a_{\mu - \nu}$. The channel output $y_\mu$ is formed as weighted sum of the shift register contents, the recent input $a_\mu$ and the white Gaussian noise $n_\mu$

$$y_\mu = \sum_{i=1}^{\nu} g_i a_{\mu-i} + n_\mu.$$  \hspace{1cm} (1)

It can be easily seen from Fig. 1 that regarding the channel as an inner (trellis) encoder the application of channel coding results in serial concatenation. The outer decoder then decodes the convolutional code and the inner one is a trellis equalizer. Therefore, in the following we use the term encoder/decoder for the outer code and ISI encoder/equalizer for the inner one. It is obvious that such structure can be decoded iteratively (see e.g. [4]).

The main idea of SCT is that only part of the source data is encoded. The source data are processed in the blocks $\langle u_{\mu} \rangle$ of length $k_T$. The data symbols of each block (in Fig. 3 denoted $\langle u^c_{\mu} \rangle$) are encoded by channel encoder which produces $n_c$ binary code symbols $c$. The $k_s = k_T - k_c$ remaining source symbols $\langle u^s_{\mu} \rangle$ are fed directly to the channel, sketched in Fig. 3. In the multiplexer the channel input block $\langle a \rangle$ of the length $n_T$ is formed from code symbols and uncoded source symbols. As a consequence, part of the channel symbols carry redundancy ("coded samples"), whereas the others directly represent information ("uncoded samples"). Denote $R_T = k_T/n_T$ the transmission rate, i.e. average amount of information bit per transmitted symbol, $R_s = k_s/n_s$ the sampling rate, i.e. the ratio of uncoded samples to all transmitted symbols and $R_c = k_c/n_c$ the code rate. For binary signaling $n_T = n_s$ $k_T = k_c + k_s$ $n_c = n_s - k_s$ hold. From eq. (2) we get

$$R_T = \frac{k_T}{n_T} = \frac{k_s}{n_T} + \frac{k_c}{n_c + k_s}$$

$$= R_s + \frac{(n_s - k_s)R_c}{n_s}$$  \hspace{1cm} (3)

$$= R_s - R_s R_c + R_c.$$  

It is obvious that $R_T$ grows monotonously in $R_c$ and $R_s$. So keeping $R_s$ constant in sampled coded transmission we are free to decide for lower transmission rate or lower code rate.

Iterative decoding is essential for sampled coded transmission. The structure of the iterative decoder of serial concatenated coding schemes was published in [2, 4]. Iterative decoder for sampled coding [5].

### Figure 3: Model of sampled coded digital transmission system

![Figure 3](image-url)

**Figure 3**: Model of sampled coded digital transmission system.
coded transmission is outlined in Fig. 4. The first decoding period in the iterative decoding is identical with the noniterative decoding. After deinterleaving the output of the equalizer is fed to the decoder. Extrinsic information has to be computed after decoding and in the following DP used as the additional input of the trellis equalizer. Therefore, the outer decoder has to be able to estimate not only data bits, but also the code bits together with their reliability. EI in the $\ell$-th decoding period is given as

$$I_{\ell,e}(c_{\mu}) = \frac{\Pr(c_{\mu} | \hat{c}_{\mu,e})}{\Pr(c_{\mu})},$$

where $c_{\mu}$ is $\mu$-th binary code symbol in the block, $I_{\ell,e}(c_{\mu})$ is EI of the code symbol $c_{\mu}$, $\hat{c}_{\mu,e}$ is the estimation of $c_{\mu}$ by the equalizer in the $\ell$-th DP and $\langle \cdot \rangle$ denotes whole block of symbols. The BCJR algorithm can be enhanced in a simple way for this task, see [9]. EI of these symbols can be computed as the quotient of the a-posteriori probability $\Pr(c_{\mu} | \hat{c}_{\mu,e})$ and the probability of the same symbols computed by the equalizer in the same decoding period $\Pr(\hat{c}_{\mu,e})$ [14]

$$I_{\ell,e}(c_{\mu}) = \frac{\Pr(c_{\mu} | \hat{c}_{\mu,e})}{\Pr(\hat{c}_{\mu,e})}. \quad (5)$$

Extrinsic information $I_{\ell,e}(c_{\mu})$ is used in the equalizer as a priori information of transmitted symbols during the following decoding period. The uncoded symbols will be equalized with a-priori probability 0.5 for each bit. The input of the outer decoder in the second and following iterations is computed from the a-posteriori probability of data and their a-priori probability according to

$$\Pr(\hat{c}_{\mu,e}) = \frac{\Pr(c_{\mu} | \hat{c}_{\mu,e})}{\Pr(\hat{c}_{\mu,e})}, \quad (6)$$

where $\langle y_\mu \rangle$ is the block of coded samples received from the channel (see also [14]).

Already after the first decoding period the coded data achieve high reliability. Hence, in the second iteration the a-priori information of the coded data is very strong and equalizer is forced to estimate coded samples according to this information. Because of the dispersive properties of the channel also the estimation of neighbouring samples is influenced and the error probability of iteratively decoded data will be lower than of noniteratively decoded ones.

We call the strategy, which data are encoded by the outer encoder and which are not the sampling pattern (SP). We characterize the sampling pattern by a sequence of zeros and ones, where "1" denotes encoding of the corresponding symbol and "0" not encoding. This pattern is applied periodically during transmission in a similar way, as puncturing matrix in punctured convolutional codes. $R_s$ can be calculated as a ratio of number of ones and all symbols in the sampling pattern. The sampling pattern has to be optimized for each individual channel. All uncoded data should have strong correlation with the coded ones. The stronger interference between coded and uncoded samples is, the greater improvement of error rate can be achieved due to iterative decoding. For example, a channel with impulse response $\langle g_i \rangle = \langle 10, 6, 3, 1 \rangle$ causes

---

4 A-posteriori probability of each transmitted symbol is computed by equalizer from channel output and EI computed by the outer decoder in the previous decoding period.
strong interferences to adjacent symbols. For this example possible sampling patterns may be as listed in Tab. 1 a and c. For an impulse response \( g(t) = (1, 0, 1, 0, 1) \) sampling pattern a is not suitable, because each uncoded sample would be superimposed only by uncoded ones. Using sampling pattern b each uncoded sample interferes with one, and using sampling pattern c with two coded symbols. So in this case one of them should be used, depending on the available bandwidth.

3. BOUNDS ON PERFORMANCE

The analysis of performance of the uncoded and coded digital transmission over ISI producing channels was done by many authors, for example see [6, 12, 13] for the uncoded transmission and [5, 10] for the coded one. Here we will analyse the error probability of the uncoded symbols in sampled coded transmission over dispersive channels.

The bit error rate \( P_e \) for digital transmission over ISI-producing channels, AWGN and maximum likelihood sequence estimation is upperbounded by (see [6])

\[
P_e \leq \sum_{d \in \mathcal{D}} C(d) A(d) Q \left( \frac{d}{\sqrt{2\sigma}} \right),
\]

where \( \mathcal{D} \) is the set of all possible normalized Euclidean distances between two different signals distorted by ISI, \( A(d) \) is the number of the error events with distance \( d \) — the distance profile, \( C(d) \) is the average number of erroneous bits in these error events, \( \sigma \) is the variance of the noise and \( Q(x) = \frac{1}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt \) denotes the Gaussian integral function. For high signal-to-noise ratio (SNR)

\[
P_e \approx C(d_{\text{min}}) A(d_{\text{min}}) Q \left( \frac{d_{\text{min}}}{\sqrt{2\sigma}} \right)
\]

well approximates the error rate. In [6] and [12] algorithms for determining \( \mathcal{D}, C(d) \) and \( A(d) \) are described. Because of their complexity we neglect in the first approach the coefficients in eq. (8) and approximate the error probability by

\[
P_e \approx Q \left( \frac{d_{\text{min}}}{\sqrt{2\sigma}} \right).
\]

Hence, only

\[
d_{\text{min}} = \min(d((a_j^{\mu}), (a_k^{\mu}))) \quad j \neq k,
\]

the minimum of the normalized Euclidean distances of all possible pairs of different paths \( a^{\mu}_j \) in the ISI trellis has to be calculated. \( \langle a_j^{\mu} \rangle \) denotes \( j \)-th noise free channel output sequence, \( \mu \) is again the time index. Operator \( d(\cdot, \cdot) \) produces Euclidean distance of two signals. Although the error probability estimation according eq. (9) is very rough, the asymptotic behaviour characterizes sufficiently.

The minimum distance \( d_{\text{min}} \) can be calculated by Dijkstra’s shortest path algorithm (DA) very efficiently, see [11, 8, 7]. We briefly sketch DA, for a detailed description see references listed above. Suppose two paths start at time \( \mu = 0 \) in the same state. For their normalized distance in the \( l \)-th step, the following expression holds

\[
d_i^l = d_i^1((a_j^{\mu}), (a_k^{\mu})),
\]

where \( d_i \) denotes the Euclidean distance of two paths in time interval \( 0 \leq \mu \leq l \), \( (a_j^{\mu}) \) is the finite length part for \( 0 \leq \mu \leq l \) of the sequence \( (a_j^{\mu}) \),

\[
D_i^l = \left[ \sum_{i=0}^\nu g_i a_{i-1} \right]^2, \quad a_{\mu} = a_j^{\mu} - a_k^{\mu},
\]

\( \nu \) number of memory elements of FIR channel model (see Fig. 2) and \( E_b \) is the energy per bit. As one can see from Eq. 11 the distance between trellis paths depends on the difference sequence \( (a_j^{\mu}) \). To get \( d_{\text{min}} \) it is sufficient to check the “weights” of nonzero paths in such a diagram, i.e. the sum of labels belonging to branches of individual paths (see eq. (11) and also [6]). One more simplification is possible: it was shown in [8] that for determining \( d_{\text{min}} \) states which differ only in a constant factor can be joined. This lead to a lower number of valid states in the difference state diagram. So only the paths started and ended in the zero difference state with at least one nonzero difference symbol \( a_{\mu} \) have to be evaluated. The minimum of the weights of these paths is the minimum Euclidean distance. The DA exploits the fact that weight of each path can not decrease with its length, see eq. (11). It starts searching from the zero difference state and checks all paths except the all-zero one. If some paths come in the same state, only that one with the lowest weight survives. In one step DA extends only the lowest weight path. A path returning to zero state determines a new upper bound on the minimum weight \( w_{\text{min}} \). Furthermore, only paths with lower weight than \( w_{\text{min}} \) are extended. If no such path remains,
the current upper bound $w_{min}$ corresponds to the minimum normalized Euclidean distance $d_{min}$.

For determining $d_{min}$ of sampled coded transmission, DA has to be modified. For this goal we suppose that extrinsic information, i.e. the feedback information is correct and provides much more reliable information than the channel output symbols. This assumption is legitimate at least for moderate and high SNR. In such case, trellis branches which are inconsistent with the extrinsic information (feedback information) are very unprobable. Therefore those branches can be removed from the ISI-trellis. Thus, for sampled coded transmission the trellis is time variant: sections for uncoded channel symbols remain the same as for uncoded transmission, in sections for coded channel symbols only branches compatible to the feedback information (i.e. to correct transmission) remain. The corresponding difference state diagram is time variant too and has less branches in the coded steps than original one — only those ones corresponding to zeros on positions of code symbols remains. Of course, there is also less number of valid states, because in the difference sequence $(a_{2-1}, a_{2-2}, ..., a_{2-v})$ positions corresponding to coded channel symbols are always zero. In the DA, the position in the sampling pattern has to be stored for each path. When two paths join in the same state, one of them can be removed only if both have the same position in the sampling pattern.

Due to the described time variance of the trellis in sampled coded transmission much less paths are valid than in original trellis. So

$$A_{SC}(d) \leq A(d) \quad \forall d \in \mathcal{D}$$

holds, where $\mathcal{D}$ is the set of all possible path’s distances in the original trellis. From equations (7) and (12) we therefore get

$$P_{e,SC} \leq P_{e}.$$  

If the sampling pattern is chosen suitable the paths with the lowest distances could be excluded, so $d_{min, SC} > d_{min}$ and BER of uncoded channel symbols in sampled coded transmission is always lower than for uncoded transmission over the same channel. We call the asymptotically difference of SNR, for which the same error rate is achieved by sampled coded transmission and uncoded transmission, respectively, the sampling gain. Sampling gain greater than zero is possible only if the error events causing $d_{min}$ correspond to difference sequences with more than one non zero element. Otherwise, $d_{min, SC} = d_{min}$ holds. Therefore, sampled coding is not suitable for all dispersive channels. For example BPSK signaling over a channel with impulse response $\langle g_i \rangle = \langle 4, 1, 1 \rangle$ has the minimum distance $72/2E_b$ and the corresponding difference sequence is $\langle 2, 0, 0 \rangle$ (see difference state diagram in Fig. 5, thick line). Sampled coding can not increase $d_{min}$ of such channel by any sampling pattern. In the contrary to this, for the channel with impulse response $\langle g_i \rangle = \langle 3, 2, 1 \rangle$ (difference state diagram in Fig. 6, the difference sequence achieving $d_{min}$ is $\langle 2, -2, 0, 0 \rangle$ — drawn thick) causes using sampling pattern from Tab. 1c increasing the minimum distance from $48/2E_b$ to $56/2E_b$ (dotted path in Fig. 6).

![Figure 5: Difference state transition diagram for BPSK transmission over the channel with impulse response $\langle g_i \rangle = \langle 4, 1, 1 \rangle$. Brunches are labeled by $D_i^j$ from eq. (11).](image)

![Figure 6: Difference state transition diagram for BPSK transmission over the channel with impulse response $\langle g_i \rangle = \langle 3, 2, 1 \rangle$.](image)

We see that for the error probability of uncoded bits (at least asymptotically) the performance of the outer code is unimportant. The only requirement on the code is to ensure higher reliability of the decoder output data than at the decoder input. This is fulfilled for almost all channel codes used in practice. Also the codes with extremely low redundancy satisfy this condition, so the sampled coded transmission is very suitable in cases, where very high information rate per transmitted symbol is required.

4. SIMULATION RESULTS

For simulation of the SCT system described in Section 2 the BCJR algorithm was used for both, equalizing and decoding. We have examined sampled coded transmission for the channels listed in Tab. 2 and baseband transmission. The sampling patterns applied are listed in Tab. 3. In all cases the outer code was a convolutional code with 16 states, which for rates higher than 0.5 was punctured.
Table 2: Impulse responses of channels used for simulations.

<table>
<thead>
<tr>
<th>Channel</th>
<th>A</th>
<th>7</th>
<th>10</th>
<th>5</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>B</td>
<td>6</td>
<td>9</td>
<td>10</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 3: Sampling patterns applied in simulations.

<table>
<thead>
<tr>
<th>Pattern</th>
<th>I</th>
<th>I</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>II</td>
<td>I</td>
<td>I</td>
</tr>
<tr>
<td></td>
<td>III</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>III</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>III</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The results of the BPSK transmission over channel A using sampling pattern I and outer code of rate 1/2 are plotted in Fig. 7. For the sake of comparison with non sampled coded transmission at the same $R_T$, the overall error rates are shown in this figure and in Fig. 8. $d_{\text{min}}$ of the channel A is 0.926. The $d_{\text{min,SC}}$ was determined with the algorithm introduced in Section 3. For this transmission is $d_{\text{min,SC}} = 2$. It can be seen that the iterative decoding consists of equalization, decoding and repeated equalization achieves (nearly) maximal gain while complexity increasing is less than 100%. From Fig. 7 it also can be seen, that sampled coded transmission over channel A using sampling pattern I and convolution code of rate $R_c = 1/2$ in certain interval of BER is more powerful than transmission with all coded symbols at the same information rate per transmitted symbol.

Similar results was achieved also for channel B, see Fig. 8. $d_{\text{min}}$ of the channel B is 0.364, for sampled coded transmission using sampling pattern I is $d_{\text{min,SC}} = 0.98$ and sampling gain more than 4.5 dB. The sampling gain is higher than for channel A, because channel B causes much stronger interference between symbols than channel A.

The next feature of the sampled coded transmission is that there are two classes of reliability of the received data. The coded bits are more reliable than the uncoded ones. The BERs of the both classes of data from simulation presented already in Fig. 7 are shown in Fig. 9. In such configuration of transmission system achieve coded data BER of $10^{-5}$ at SNR approximately 4 dB lower than the uncoded ones.

The influence of different sampling patterns on the BER of uncoded data is very strong. Simulation results of sampled coded transmission under the same conditions as simulation from Fig. 7, but with different sampling patterns are shown in Fig. 10. It is obvious that the choice of the suitable sampling pattern is crucial. If too little samples of data are coded, only very low or no sampling gain will be achieved. Contrary, if too many samples are coded, transmission rate will be lowered, but no additional gain will be achieved.

The influence of the outer code on the BER of uncoded bits is marginal, as it was already presupposed in Section 3. The BER curves of the sampled coded transmission over channel A using sampling pattern I and convolutional codes of rates 0.33, 0.5 and 0.9 are shown in Fig. 11. $BER = 10^{-5}$ is achieved at $E_b/N_0$ only 0.2 dB higher by using the highest rate code than by using the lowest rate one. By drawing the BER curves over $E_b/N_0$ (the overall rate of the sampled coded transmission with the highest rate code is 0.66 and with the lowest rate one 0.55) the highest rate sampled coded transmission achieves $BER = 10^{-5}$ at $E_b/N_0$ more than 1.5 dB lower than the lowest rate sampled coded transmission, see Fig. 12.

---

The overall error rate is an average error rate of coded and uncoded data. Using approximation $BER = \max(BER_c, BER_u)$, holds for moderate and low error rates $BER = BER_u$, where $BER$ is the overall bit error rate, $BER_c$ bit error rate of coded bits and $BER_u$ of uncoded ones.
5. CONCLUSION

In this paper a novel coding strategy for transmission over dispersive channels was introduced. It was shown that by using iterative decoding it is possible to improve reliability not only of the coded data, but also of the uncoded ones. By suitable positioning of coded symbols in the transmitted sequence and suitable choice of the code used SCT can outperform a transmission with all data coded at the same overall transmission rate. The main potential field of using the SCT is compatible improving of already established standards in which both, coded and uncoded transmission are realized simultaneously (for example GSM and G.723). The other useful implementation can be transmission with very high information rate per symbol.

In principle, the same results can be achieved by concatenation (parallel or serial) of an outer channel code with any other intersymbol dependencies causing processor as inner code, for example some channel code or some kind of modulation as CPM, DPCM and GMSK.

REFERENCES


Figure 11: Sampled coded transmission over channel A, sampling pattern I using as an outer code convolutional code of rate 0.33 - line marked with “+”, code of rate 0.5 - line marked with “o” and code of rate 0.9 - line marked with “*”. BER drawn versus $E_b/N_0$.


