Bit-Interleaved Coded Modulation for Delay-Constrained Mobile Communication Channels

Hugo M. Tullberg, Paul H. Siegel, *IEEE Fellow*

Center for Wireless Communications
UCSD, 9500 Gilman Drive, La Jolla CA 92093-0407, USA
{htullber,psiegel} @cwc.ucsd.edu

Abstract – The role of the interleaver in a Bit Interleaved Coded Modulation (BICM) system is investigated. Square block interleavers and convolutional interleavers are compared to the random interleaver originally used by Zehavi [1]. It is shown that for short latencies (20 ms) the square block interleaver performs better than the random interleaver. However, when the side of the square block interleaver, $\sqrt{N}$, is a multiple of n, the coded bits are grouped in such a way that the diversity, and hence performance, is reduced.For short delays, the convolutional interleaver outperforms both the random and square block interleaver as the vehicle speed varies from pedestrian to freeway speeds.

I. Introduction

In wireline modems, Trellis Coded Modulation (TCM) techniques achieve high spectral efficiency by generating coded sequences with large Euclidean distance. When designing codes for a wireless communications channel subject to fading, we encounter two major differences. First, the performance now depends on Hamming distance instead of Euclidean distance, and second, the error bursts generated by the fades must be broken up. The new performance criteria are addressed by using codes designed for maximum Hamming distance in conjunction with Gray-labeled signal constellations. To break up the fades, different interleaving methods are used: symbol-by-symbol interleaving, I-Q-interleaving [2], coordinate interleaving [3], and Bit Interleaved Coded Modulation (BICM) [1], [4]. The longer the interleaver, the better, in the sense that the channel samples tend towards independent random samples from some distribution, usually Rayleigh. In voice communications, however, the size of the interleaver is limited by a delay constraint. The acceptable interleaver delay is usually taken to be 20 ms. In this work, we investigate the performance of a BICM system when such a delay constraint is applied. We compare square block interleavers and convolutional interleavers to the random interleavers originally used by Zehavi.

The channel model used in this work is a correlated Rayleigh fading channel. Our choice of signaling parameters ensures that the transmitted signal experiences a slowly varying, flat fading channel.

This work was performed with the support of the Swedish Defence Research Establishment (FOA) and the National Science Foundation under Grant NCR-9612802.

II. System Description

In a BICM system, the encoded bits are permuted before they are passed to the signal constellation mapper. This increases the diversity order with the smallest possible reduction in free Euclidean distance.

The information sequence $\boldsymbol{i}$ is fed into a rate $R = k/n$ feedforward convolutional encoder designed for maximum Hamming distance. The output sequence $\boldsymbol{c}$ of n-tuples from the convolutional encoder is fed into an interleaver, spreading the n bits in time to break up fades. The interleaver may operate on the n bitstreams individually, as in Zehavi's original system, or multiplex the bitstreams into a single bitstream before permuting the bits. After the interleaver, the bits are grouped into sequence $\boldsymbol{c}'$ of permuted n-tuples. The permuted sequence $\boldsymbol{c}'$ is mapped onto a Gray-labeled signal constellation with 2^n signal points by a memoryless mapper, $\boldsymbol{x} = \mu(\boldsymbol{c}')$.

At the receiver, we have a faded sequence corrupted by additive white Gaussian noise, $\boldsymbol{y} = \boldsymbol{\rho x} + \boldsymbol{n}$. Each of the n bits that make up a channel symbol, y_k, partitions the signal constellation into two subsets, $S_i^c, i = 1 \ldots n, c \in \{0, 1\}$, where S_i^c is the subset of constellation points where the i-th bit in the label takes on the value c. For each of the n bits the decoder computes two suboptimal metrics, one for each value of the bit c^i,

$$m_i\left(y_k, S_i^c; \rho_k^i\right) = \min_{x \in S_i^c} ||y_k - \rho_k x||^2, \quad c = \{0, 1\},\ i = 1, \ldots, n \tag{1}$$

The metrics are deinterleaved and combined into branch metrics for the possible transitions in the code trellis,

$$\begin{aligned} &m\left(y_t, c_t; \rho_t\right) \\ &= \sum_{i=1}^{n} (1 - c_t^i) m_i(y_t^i, S_i^0; \rho_t^i) + c_t^i m_i(y_t^i, S_i^1; \rho_t^i). \end{aligned} \tag{2}$$

Finally, the convolutional code is decoded to the path that minimizes the accumulated metric,

$$m(\boldsymbol{y}, \boldsymbol{c}; \boldsymbol{\rho}) = \sum_{p=1}^{N} m\left(y_t, c_y; \rho_t\right), \tag{3}$$

by applying the Viterbi Algorithm.

The structure of a BICM system using a rate $R = 2/3$ convolutional code is shown in Fig. 1.

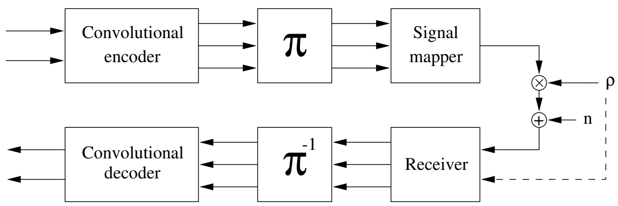

Fig. 1. Structure of a BICM system.

III. Performance

Following the analysis in [5], an upper bound for the bit error probability for a coded system over a Rayleigh fading channel at high signal-to-noise ratios (SNR), is given by

$$P_b \leq \sum_{\boldsymbol{x},\hat{\boldsymbol{x}}\in\mathcal{C}} A(\boldsymbol{x},\hat{\boldsymbol{x}})p(\boldsymbol{x}) \prod_{n\in\eta} \frac{4}{\left(\frac{\bar{E}_s}{N_0}\right)\|\hat{x}_n - x_n\|^2} \quad (4)$$

where $A(\boldsymbol{x},\hat{\boldsymbol{x}})$ is the number of bit errors that results when the receiver decodes to the sequence $\hat{\boldsymbol{x}} \neq \boldsymbol{x}$ instead of the transmitted sequence $\boldsymbol{x}$, $p(\boldsymbol{x})$ is the a priori probability of transmitting the sequence $\boldsymbol{x}$, $\mathcal{C}$ is the set of possible sequences, $\bar{E}_s/N_0$ is the average signal-to-noise ratio, and η is the index set of non-zero distances between symbols in the sequences $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$. The cardinality of η is the number of non-zeros distances between the symbols along the correct path and the symbols along an error event. By using a Gray-labeled signal constellation we assure that the normalized squared Euclidean distance is lower-bounded by

$$\frac{d_E^2(\mu(\hat{\boldsymbol{c}}),\mu(\hat{\boldsymbol{c}'}))}{E_s} \geq d_H(\hat{\boldsymbol{c}},\hat{\boldsymbol{c}}') \cdot \Delta_0 \quad (5)$$

$$\geq d_{\text{free}} \cdot \Delta_0 \quad (6)$$

where d_{free} is the free binary Hamming distance of the convolutional code and Δ_0 is the minimum squared Euclidean distance of the signal constellation. Hence, a code with good Hamming distance gives good squared Euclidean distance.

In his original paper, Zehavi used codes with maximal d_{free} from [6, p. 331]. By using codes with Optimum Distance Profile, i.e., codes whose distance profile is equal or superior to that of any other code with the same memory [7, p. 112], performance improvements can be achieved. Simulations comparing the feedforward and recursive systematic form of the rate $R = 2/3$, memory-3 code in [6, p. 331] with the ODP code having the same parameters in [7, p. 360] show a performance gain of 1 dB at bit-error-rate (BER) 10^{-5}; see Fig. 2. Both codes have a free Hamming distance of 4 but their weight spectra are slightly different. The number of low weight codewords for the two codes in shown in Table I.

The Hamming weight-4 error event is of length two, with weight 2 on the diverging branch and weight 2 on the remerging branch for both the ODP and non-ODP codes. For weight-5 error events, all error events begin with a diverging branch with weight 2 and end with a weight-2 remerging branch. There is one branch with Hamming weight 1 and in some cases one or more branches with Hamming weight zero. Our interpretation is that the performance differences between the feedforward version of the ODP and non-ODP codes can be attributed to the difference in multiplicity of the weight-5 error events.

		i			
		0	1	2	3
non-ODP	[6, p. 331]	1	8	24	73
ODP	[7, p. 360]	1	5	24	71

Table I Number of output codewords of weight $d_{\text{free}} + i$ for two different codes.

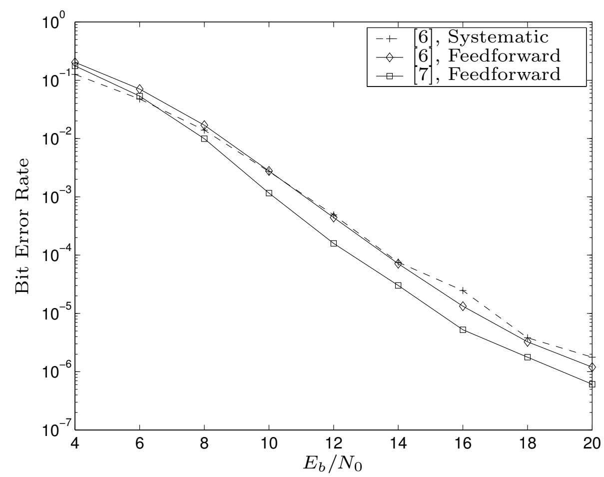

Fig. 2. Comparison of three different codes.

IV. Investigated Interleavers

The role of the interleaver in a BICM system is to break up the fades on the correlated channel. In this study, three kinds of interleavers have been compared: random block interleavers, square block interleavers, and convolutional interleavers.

A. Random Interleavers

The random block interleaver permutes each of the n bit streams from the convolutional encoder separately. For large block sizes, the random interleaver mimics an infinite interleaver well and the channel tends towards an uncorrelated fading channel. The size of the random interleaver is chosen to be as large as the delay constraint allows. The random interleaver is used as a baseline for comparisons.

B. Square Block Interleavers

In the square block interleaver, the n output bit streams are interlaced into a single bitstream and stored in a square matrix of size N. The bits are written row-wise and read out column-wise. For this reason, this kind of interleaver is sometimes referred to as a transposition interleaver. This interleaver can break up fades of length up to $\sqrt{N}/n$ symbols. Block interleavers can also be rectangular, but we have not investigated them in this work.

C. Convolutional Interleavers

A convolutional interleaver consists of a shift register and a commutator to either insert symbols into or read symbols from the shift register. Convolutional interleavers are naturally stream-oriented and therefore well-suited to use with convolutional codes. In recent research, convolutional interleavers have been used to streamline turbo codes [8].

Ramsey [9] introduced a class of convolutional interleavers described by two parameters (n_2, n_1) such that no contiguous sequence of n_2 symbols in the output sequence from the interleaver contains any symbols that were separated by less than n_1 symbols in the input sequence. Depending on how the convolutional interleaver is implemented, certain criteria on relative primeness of n_1 and n_2 must be met. By adding some shift logic to the shift register and thereby only storing symbols yet to be read out, the memory can be reduced roughly by a factor of two compared to a square block interleaver able to break up fades of comparable size.

To determine the parameters n_2 and n_1 we need to know the average fade duration $\bar{\tau}$. Let $\bar{S}$ be the time average of the fading amplitude. Depending on the additive noise, this corresponds to an average SNR. If the instantaneous amplitude s of the fading process falls below a system-dependent threshold S_{err}, we will make an error with high probability. The average duration of a fade $s \leq S_{\text{err}}$ is then given by [10, p. 36].

$$\bar{\tau} = \frac{e^{\gamma^2} - 1}{\sqrt{2\pi} f_d \gamma}, \tag{7}$$

where $\gamma = S_{\text{err}}/\bar{S}$ and $f_d = v/\lambda_c$ is the maximum Doppler frequency.

On average $\bar{\tau} \cdot R_S$ transmitted symbols will be affected by a fade, where R_S is the signaling rate. We want these symbols to be spread by the deinterleaver so that no two of them will end up closer than $\alpha \cdot \nu$ in the deinterleaved sequence, where ν is the constraint length of the convolutional code and α is some constant. If we interpret α as the truncation depth of the Viterbi decoder, α should be 4 or 5. The delay introduced by the interleaver depicted in Fig. 3 is $D = (n_1 - 1)(n_2 + 1)$. Subject to the overall delay constraint, we let $n_2 \geq \bar{\tau} \cdot R_S$ and $n_1 \geq \alpha \cdot \nu$.

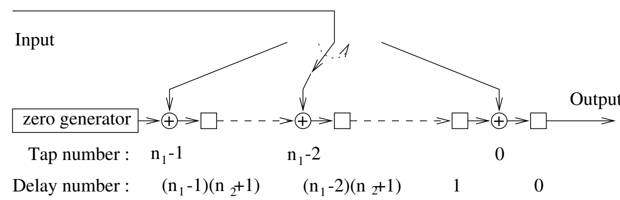

Fig. 3. Ramsey Type IV convolutional interleaver.

We would like $n_2 \geq \bar{\tau} \cdot R_S$ and $n_1 \geq \alpha \cdot \nu$ with $\alpha = 5$. Due to the overall delay constraint, this is not always possible. We simulated a system with $\gamma = 0.54$ and an overall delay constraint of 20 ms. With a signaling rate of 20 k symbols per second, this gives an average fade duration $\bar{\tau} = 2.98$ ms, corresponding to 60 symbols and an interleaver delay of 400 symbols. We used a memory-3 code and considered $n_1 = 3, 6, 9, 12, 15$ and $n_2 = 200, 79, 49, 35, 28$. The resulting performance is shown in Fig. 4.

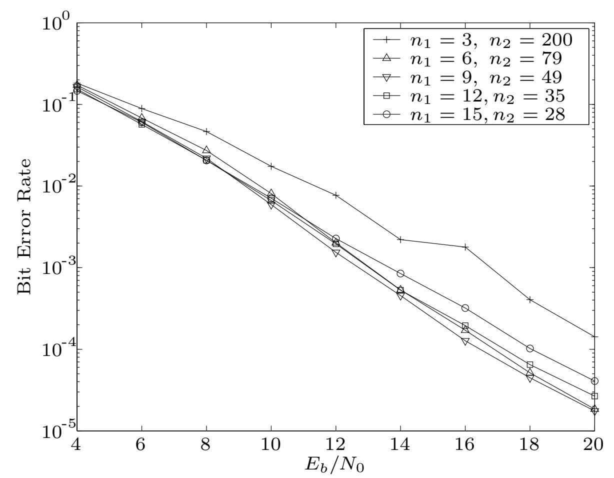

Fig. 4. Performance as a function of the parameters n_1 and n_2.

For small values of n_2, we can break up long fades, but the deinterleaved channel samples end up too close to each other for the code to be able to correct the errors, i.e., the resulting n_1 is too small. As n_1 grows larger, the performance improves up to a point where n_2 is too short to break up the fades. The actual values to be used depend on the system used.

The convolutional interleaver performs a subsampling of the fading channel with a factor of n_2 and the effective fade duration is reduced. The fading process is divided into n_1 subsequences and interlaced such that the (n_2, n_1) constraint is met. This effectively gives a fading process with a higher Doppler frequency, $f_d' = n_2 f_d$. An example of a fading process and the deinterleaved fading process is shown in Fig. 5. The envelope of one of the n_1 subprocesses is indicated with a dashed line.

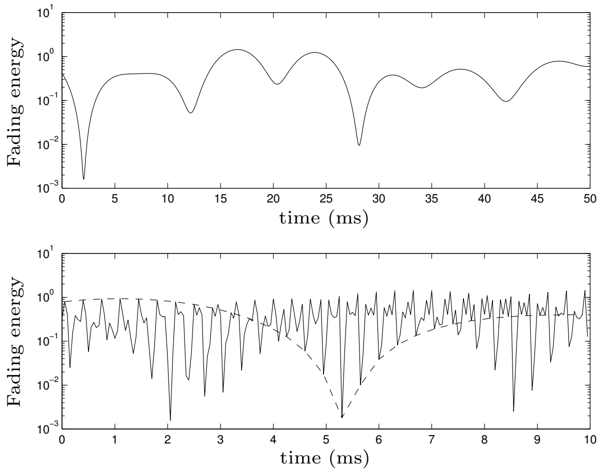

Fig. 5. Original and deinterleaved fading process. Note the different time scales.

V. Simulation Results

In the following simulations we have used a rate $R = 2/3$ convolutional code and a Gray-labeled 8-PSK signal constellation. The signaling rate is 20 k symbols per second. The channel is modeled as a correlated, slowly varying , flat Rayleigh fading channel.

A. Latency Constraint

In voice communications, large interleaver size results in unacceptable latency and thus shorter block lengths are required. An often-used number for acceptable interleaver latency is 20 ms. Although increasing the signaling rate would allow more symbols per block, the performance would not in general improve, because more symbols would be affected by the fades.

We first compare the performance of a random interleaver and square interleaver for different interleaver delays, shown in Fig. 6. In this case the vehicle speed is 100 km/h, corresponding to $f_d = 83$ Hz. For short interleaver delays, the size of the interleaver is so small that the randomly interleaved channel does not mimic an uncorrelated fading channel particularly well and the square block interleaver actually performs slightly better than the random interleaver.

Note the "peaks" in the curve for the square block interleaver at 25 ms and 48 ms. For the interleavers corresponding to these delays, the side of the interleaver, $\sqrt{N}$, is divisible by n, and all bits that make up a channel symbol come from the same bitstream from the convolutional encoder. This reduces the diversity in the system and results in a degradation in the performance. This effect can be avoided by choosing the size of the interleaver such that the side of the interleaver, $\sqrt{N}$, is relatively prime to n. In particular, if we choose $\sqrt{N} = (c \cdot n) + 1$, where c is some constant, we get an interleaver of size $N = (c^2 n + 2c)n + 1$. In this case, there will always be one unused memory element in the lower right corner but this will not affect the function of the interleaver since that element is the last element when both writing and reading data.

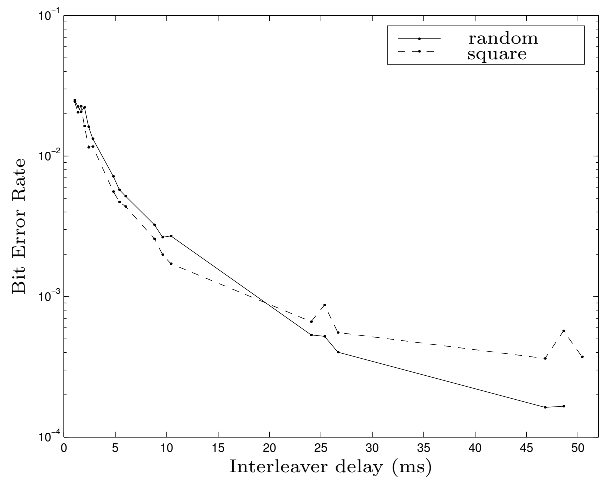

Fig. 6. Performance as a function of interleaver delay for random and square interleavers.

B. Mobile Speed

When the speed of the mobile varies, so does the characteristic of the channel. For high speeds, the average fade duration is comparably short, a few milliseconds at 100 km/h. When the speed of the mobile is reduced, the average duration of a fade increases. At some point the channel variations are so slow that the interleaver no longer can break up the fades.

We now compare the performance of three different interleavers as a function of the speed of the mobile. The random interleaver permutes the three bitstreams independently, 385 symbols at a time. The square interleaver interlaces the 385 symbols into a block of 1155 bits, corresponding to a side length $\sqrt{N} = 34$ bits. The convolutional interleaver permutes the three bitstreams separately, using the parameters $n_1^1 = 7, n_1^2 = 8, n_1^3 = 9$ and $n_2^1 = 65, n_2^2 = 55, n_2^3 = 49$, respectively. The convolutional interleavers have a maximum delay of 20 ms. The simulation results are show in Fig. 7.

Besides the general degradation of performance with decreasing speed, there is an additional "periodic" variation of the performance, in particular for the convolutional interleavers. The convolutional interleaver is designed for a particular average fade duration, corresponding to a particular speed. When the speed changes, so does the average fade duration and the convolutional interleaver is no longer suitable. However, for most mobile speeds the convolutional interleavers outperform both the square block and random interleavers. This suggests that the convolutional interleaver better breaks up the fades when the permissible interleaver delay is short.

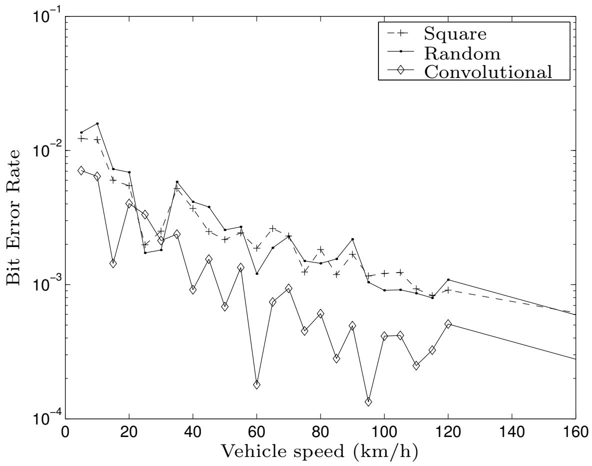

Fig. 7. Performance for three different interleavers as a function of the mobile speed.

VI. Conclusions

In this paper we have shown that the distance profile of the convolutional code used in a BICM system has a significant effect on the system performance. In particular, simulations show that codes having an optimum distance profile outperforms non-ODP codes.

For very short delays, the square block interleaver performs slightly better than the random interleaver, but they are comparable at an interleaver delay of 20 ms.

For an interleaver delay of 20 ms, the convolutional interleaver outperforms both the square block and random interleaver over a wide range of mobile speeds.

References

[1] E. Zehavi, "8-PSK trellis codes for a Rayleigh channel," *IEEE Transactions on Communications*, vol. 40, pp. 873–884, May 1992.

[2] S. A. Al-Semari and T. E. Fuja, "I-Q TCM: Reliable communication over the Rayleigh fading channel close to the cutoff rate," *IEEE Transactions on Information Theory*, vol. 43, pp. 250–262, January 1997.

[3] B. D. Jeličić and S. Roy, "Cutoff rates for coordinate interleaved QAM over Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 44, pp. 1231–1233, October 1996.

[4] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, pp. 927–946, May 1998.

[5] D. Divsalar and M. K. Simon, "The design of trellis coded MPSK for fading channels: Performance criteria," *IEEE Transactions on Communications*, vol. 36, pp. 1004–1012, September 1988.

[6] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, New Jersey: Prentice-Hall, 1983. ISBN 0-13-283796-X.

[7] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding*. New York, NY, USA: IEEE Press, 1999. ISBN 0-7803-3483-3.

[8] E. K. Hell and S. G. Wilson, "Stream-oriented turbo codes," in *Proceedings IEEE Vehicular Technology Conference*, vol. 3, (Ottawa, Ont., Canada), pp. 71–75, IEEE, May 1998.

[9] J. L. Ramsey, "Realization of optimum interleavers," *IEEE Transactions on Information Theory*, vol. 16, pp. 338–345, May 1970.

[10] W. C. Jakes, Jr., *Microwave Mobile Communications*. New York: Wiley, 1974. ISBN 0-471-43720-4.