Bit Interleaved Space-Time Codes

Tarik Muharemovic, Alan Gatherer, Will Ebel, Sri Hosur, Dale Hocevar
and Everest Huang

This research was done at Texas Instruments.

Abstract

We develop a new construction criterion for BSPK space-time codes which simultaneously reflects both "coding" and "diversity" advantage. Furthermore, we propose a new method of improving the performance of space-time codes, namely bit interleaving. Subsequently we provide a computationally effective, iterative way of decoding bit interleaved space-time codes. At the end we note the 2dB gain due to the new approach. We note the sixteen state code which operates close to 3dB from the outage probability.

Keywords

Transmit diversity, natural space-time codes, bit interleaving, MAP, iterative decoder.

I. INTRODUCTION

Transmit and Receive diversity are the only source of diversity in frequency non-selective, slowly fading channels. Each receive antenna receives a superposition of transmitted constellation symbols from each of the transmit antennas together with AWGN. Every symbol is randomly faded, as in Fig. 1. Here, E_S represents power transmitted on each antenna and j is the time index. Initially the idea was to generate an "artificial" multipath using multiple transmit antennas. To achieve this, the second antenna would track the first, with one or possibly more symbols delay. This is referred to as delay diversity [1]. Clearly, in order to have increased levels of diversity some type of correlation needs to be introduced among transmit antennas. Following these discoveries, a great deal of research was done in order to find out the best way of introducing this correlation ([2] [3] and others). This is popularly known as space-time coding. One of the underlying principles of space-time coding is that code matrices rather than codewords are transmitted due to the increase in dimension. Note that the code matrices reduce to codewords when there is only one transmit antenna available. If the transmitter has channel state information (CSI) then different types of orthogonalisation can be done and this problem is reduced to the problem of "one dimensional" coding. However, if the transmitter does not have CSI then pairwise error probability is the standard approach [2], [4] in providing guidelines for constructing good codes. Most research so far has been done for two-transmit, single receive antenna systems and we will proceed in the same manner.

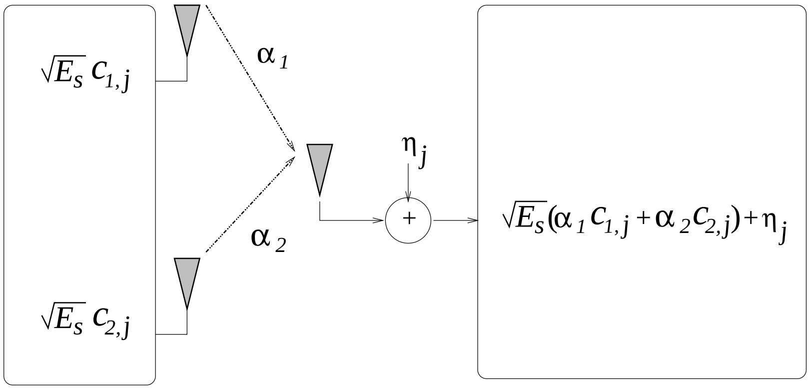

Fig. 1. Channel Model

II. Preliminaries and Previous Results

Assume that the bit stream is encoded with a rate 1/2 coder into a matrix b where rows b_1 and b_2 represents outputs of two coder branches. Let l be the transmission frame length and hence b is a binary $2 \times l$ matrix. b is further mapped onto constellation symbol matrix $c = (-1)^b$ which is sent over the channel as in Fig. 2. We refer to this transmission strategy as a "natural" space-time code [3]. If we normalize the constellation on each antenna to a unit energy BPSK, then the receiver at time j observes

$$y_j = \sqrt{E_s}(\alpha_1 c_{1,j} + \alpha_2 c_{2,j}) + \eta_j \tag{1}$$

where E_s is the transmission power on each of the antennas, α_1 and α_2 are Rayleigh faded coefficients which are constant throughout the frame transmission and η_j is Complex AWGN with variance $\frac{N_0}{2}$ per dimension. This is apparent from the channel model in Fig. 1. We examine pairwise error probability and for that purpose we denote $\tilde{b}$ as the matrix of encoded bits that the decoder incorrectly chooses, and $\tilde{c} = (-1)^{\tilde{b}}$. Let $\hat{b} = b \oplus \tilde{b}$ and $\hat{c} = c - \tilde{c}$. Note that the entries of $\hat{c}$ are 0 or ± 2 and that $\hat{b}_{i,j} = 0 \Leftrightarrow \hat{c}_{i,j} = 0$. We also require the following, known, results:

Performance Criterion [2]: Pairwise error probability is upper bounded as

$$\mathbb{P}[b \to \tilde{b}] = \mathbb{P}[c \to \tilde{c}] \leq \frac{1}{\Lambda}(\frac{E_s}{4N_0})^{-r} \tag{2}$$

where coding advantage Λ is the product of non-zero eigenvalues of $\hat{c}\hat{c}^\star$, and $\star$ denotes conjugate transposed. Diversity advantage r is the rank of $\hat{c}$.

Diversity Criterion [3] : $\text{rank}(\hat{c}) = \text{rank}(\hat{b})$ in F_2, the prime field of order 2.

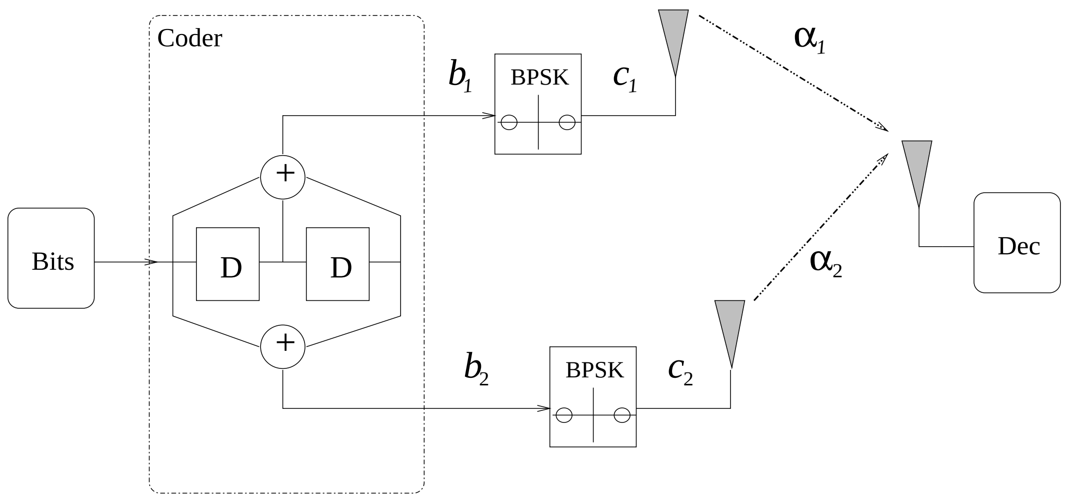

Fig. 2. A (7,5) natural Space-Time Code

III. Theory

Here we seek a function on $(b, \tilde{b})$ which will simultaneously reflect both coding and diversity advantage. For that purpose define $d_1 = d_1(b, \tilde{b}) = \sum_{j=1}^{l} \mathbb{I}[\hat{b}_{1,j} \neq 0]$ to be the hamming weight of the 1st row of $\hat{b}$. Here $\mathbb{I}[\cdot]$ denotes the indicator function. Similary $d_2 = d_2(b, \tilde{b}) = \sum_{j=1}^{l} \mathbb{I}[\hat{b}_{2,j} \neq 0]$ is the hamming weight of the 2nd row of $\hat{b}$. We also define $d_3 = d_3(b, \tilde{b}) = \sum_{j=1}^{l} \mathbb{I}[\hat{b}_{1,j} \oplus \hat{b}_{2,j} \neq 0]$ to be the hamming weight of $\hat{b}_1 \oplus \hat{b}_2$. Let $d^2 = (d_1 + d_2 + d_3)^2 - 2(d_1^2 + d_2^2 + d_3^2)$, where we suggestively write d^2 because we will show it to be a non-negative quantity.

Proposition 1: $d^2 \geq 0$

Proof: Since d_3 is the Hamming weight of $\hat{b}_1 \oplus \hat{b}_2$ and d_1 and d_2 are Hamming weights of $\hat{b}_1$ and $\hat{b}_2$ respectively, it follows that $0 \leq d_3 \leq d_1 + d_2$. Also because $\hat{b}_2 = (\hat{b}_1 \oplus \hat{b}_2) \oplus \hat{b}_1$ it follows that $0 \leq d_2 \leq d_3 + d_1$, and similary $0 \leq d_1 \leq d_2 + d_3$. We multiply these three inequalities by d_3, d_2 and d_1 respectively and add them up to get $d_1^2 + d_2^2 + d_3^2 \leq 2(d_1 d_2 + d_1 d_3 + d_3 d_2)$. From here the result follows. ■

Proposition 2: $d^2 = 0 \Leftrightarrow \text{rank } (\hat{b}) \leq 1$. Hence if the code is full diversity then $d^2 > 0$. Also, if the code is not full diversity then $d^2 = 0$.

Proof: ($\Rightarrow$): We examine when the equality holds in Proposition 1. If we suppose that d_1, d_2 and d_3 are all positive, then from the three inequalities in the proof of Proposition 1, the equalty holds only if $d_3 = d_1 + d_2$, $d_2 = d_1 + d_3$ and $d_1 = d_3 + d_2$. This implies that

$d_1 = d_2 = d_3 = 0$, and thus we have reached a contradiction. If any one of d_1, d_2 and d_3 is zero, then both $d^2 = 0$ and the rank $(\hat{b}) \leq 1$.

($\Leftarrow$): If the rank is one then either $\hat{b}_1 = 0$ or $\hat{b}_2 = 0$, or $\hat{b}_1 = \hat{b}_2$. Hence, either $d_1 = 0$ or $d_2 = 0$ or $d_3 = 0$. In every case it follows that $d^2 = 0$ by simple computation. The case of zero rank is even more trivial. ■

Proposition 3: $\Lambda \geq 4d^2$

Proof: If the code is not full diversity then $d^2 = 0$ and there is nothing to prove. If $\hat{b}$ is full rank so is $\hat{c}$, by the Diversity Criterion. The product of non-zero eigenvalues is then just the determinant of $\hat{c}\hat{c}^\star$:

$$\Lambda = \det(\begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \end{bmatrix} \begin{bmatrix} \hat{c}_1^\star & \hat{c}_2^\star \end{bmatrix}) = \|\hat{c}_1\|^2 \|\hat{c}_2\|^2 - |\hat{c}_1 \hat{c}_2^\star|^2. \tag{3}$$

Note that $\|\hat{c}_1\|^2 = \sum_{j=1}^{l} |\hat{c}_{1,j}|^2 = 4 \sum_{j=1}^{l} \mathbb{I}[\hat{c}_{1,j} \neq 0] = 4 \sum_{j=1}^{l} \mathbb{I}[\hat{b}_{1,j} \neq 0] = 4d_1$. Similarly we show that $\|\hat{c}_2\|^2 = 4d_2$. Also, we can upper bound the term $|\hat{c}_1 \hat{c}_2^\star|$ as

$$\begin{aligned} |\hat{c}_1 \hat{c}_2^\star| = |\sum_{j=1}^{l} \hat{c}_{1,j} \hat{c}_{2,j}^\star| \leq \sum_{j=1}^{l} |\hat{c}_{1,j}||\hat{c}_{2,j}| = 4 \sum_{j=1}^{l} \mathbb{I}[\hat{c}_{1,j} \hat{c}_{2,j} \neq 0] = 4 \sum_{j=1}^{l} \mathbb{I}[\hat{b}_{1,j} \hat{b}_{2,j} \neq 0] \\ = 4 \sum_{j=1}^{l} \frac{1}{2} (\mathbb{I}[\hat{b}_{1,j} \neq 0] + \mathbb{I}[\hat{b}_{2,j} \neq 0] - \mathbb{I}[\hat{b}_{1,j} \oplus \hat{b}_{2,j} \neq 0]). \end{aligned} \tag{4}$$

Here, the last equality is easily verified "term by term," by examining possibilities for $\hat{b}_{1,j}$ and $\hat{b}_{2,j}$. Hence, $|\hat{c}_1 \hat{c}_2^\star| \leq 2(d_1 + d_2 - d_3)$. It now follows that the determinant $\Lambda \geq 16 d_1 d_2 - 4(d_1 + d_2 - d_3)^2 = 4d^2$. The factor of 4 is just a nuisance coming from the unit energy constellation normalization, distance squared between the two BPSK points being 4. ■

So, what we have obtained is slightly looser upper bound on the pairwise error probabiliy.

$$\mathbb{P}[b \rightarrow \tilde{b}] \leq \frac{1}{4d^2} (\frac{E_s}{4N_0})^{-2} \tag{5}$$

For non-full diversity codes this bound is infinite, thereby indicating the weakness of the code. For a BPSK constellation, achieving full diversity is not generally difficult [3], and we can assume that almost any code we consider has full diversity. However, this bound will reveal interesting facts about bit interleaving. Furthermore note that

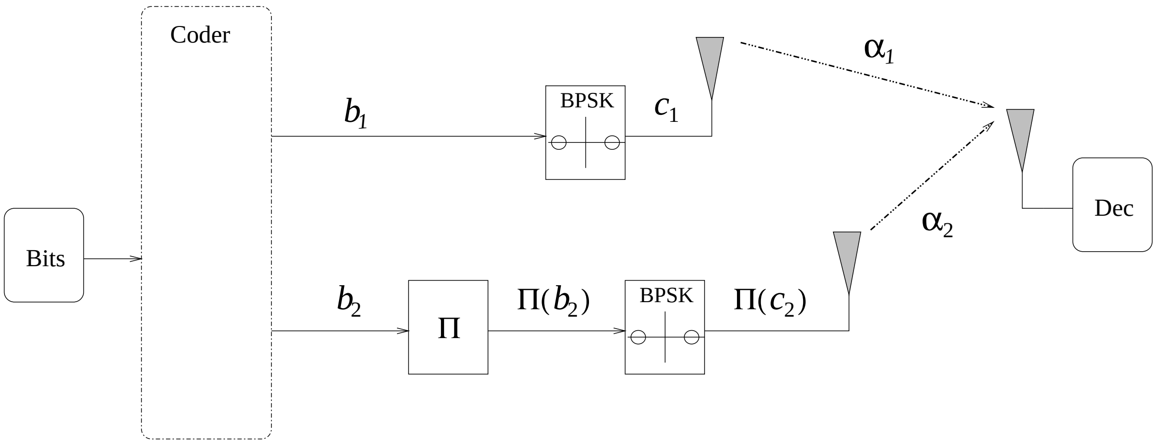

Fig. 3. System Description

$d^2(b,\tilde{b}) = d^2(b \oplus \tilde{b}, 0) = d^2(\hat{b}, 0)$ and thus we have "linearized" the code, that is to say we only need to consider codewords and not all codeword pairs when analyzing the code. From now on we reference everything with respect to the all zero code-matrix.

Proposition 5: As either d_1, d_2 or d_3 increases so does d^2 thereby decreasing the bound on the error probability, for a full diversity code.

Proof: We can do this by induction on either d_1, d_2 or d_3 because of the symmetry of the problem. First note that if one of the d_1, d_2 or d_3 increases by 1 so does exactly one of the two remaining terms. This is due to the $\oplus$ dependency. Because of the symmetry of the problem we assume that d_1 and d_2 increase by 1, and that it results in the total distance of d_{new}. Then:

$$d_{new}^2 = ((d_1+1)+(d_2+1)+d_3)^2 - 2((d_1+1)^2+(d_2+1)^2+d_3^2)$$

$$= (d_1+d_2+d_3)^2 + 4(d_1+d_2+d_3) + 4 - 2(d_1^2+d_2^2+d_3^2) - 4(d_1+d_2) - 4$$

$$= d^2 + 4d_3. \tag{6}$$

The result now follows because $d_3 > 0$. ■

IV. ENCODING WITH BIT INTERLEAVING

Coders with bigger memory will result in larger d_1, d_2, d_3 and therefore will give better performance. We will construct a simple way of devising a coder which maintains the same memory - or constraint length - but yields a larger d_3. Such a coder is simply given

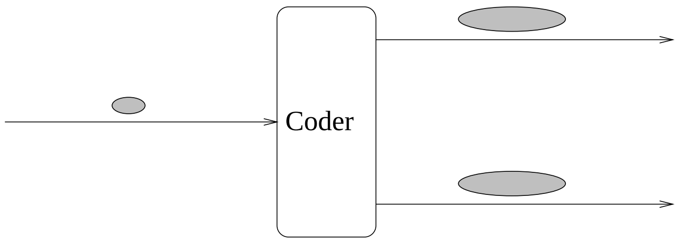

Fig. 4. Non-interleaved space-time code results in low values of d_3

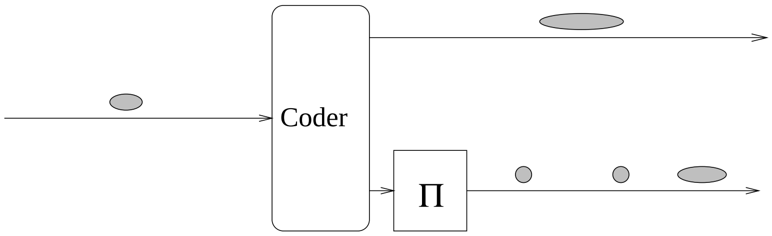

Fig. 5. Interleaved space-time code results in high values of d_3

by bit interleaving on one of the antennas as shown in Fig. 3. Bits are first encoded with a convolutional coder, then bits on one antenna are interleaved while on the another antenna they are not. Consecutively, bits are mapped to a BPSK constellation with a standard map $c = (-1)^b$ and transmitted over the channel. If we use maximum free distance non-recursive convolutional coders then low (high) input weights correspond to low (high) output weights. Using proposition 5 we see that as the output weights increase so does d^2, therefore yielding better performance. It follows that error events are dominated by low input weight codewords. In particular, a single input one yields the smallest d^2, naturally, with respect to the all zero codeword. In addition to that, such an event has extremely high multiplicity due to the shift invariance; it is equal to the length of the frame minus the memory of the coder. As for the non-interleaved code, observe that both branches of the coder produce ones in "clumps" and that there is a significant overlap amongst ones from the first branch and ones from the second branch. This results in low values of d_3. This becomes apparent from looking at Fig. 4. For example, it is easily verified that for (7,5) coder and the impulse response, $d_3 = 1$. Bit interleaving is done in order to decrease overlap amongst ones from the first branch and ones from the second branch of the coder, similar to Fig. 5. For the purpose of iterative decoding we propose a random permutation interleaver. This will in turn significantly increase d_3 and will give a better metric d^2. Note that due to the symmetry of d^2, d_3 is as important as d_1 and d_2. Of course, the maximum possible value of d_3 is $d_1 + d_2$. We do not exclude the possibility that the interleaver may produce some overlap between ones from the first and the second branch; however the probability of such overlap for low input weight is low and the multiplicity of such an event (if it happens) is also very low. For high input weight codewords overlap doesn't matter because d_1 and d_2 are already high enough. Note that bit interleaving is not done in order to increase diversity but rather to increase the coding advantage. Of course, there is always an issue whether bit interleaving would violate the full diversity criterion; however the probability of such an event is also low. If it happens, we pick another interleaver. It is interesting to note that simple rotation of the bottom row of the codeword does not improve the codeword as much as a random interleave of the bottom row. The reason for this is that the fundamental "clumped" nature of the error event remains and a two error event scenario can be proposed in which the bottom row from one error event overlaps with the top row of the other and vice versa. Random interleaving of the bottom row changes the fundamental nature of an error event from a clump to a scattered pattern. As the top row error event is still constrained into a clumped pattern this difference in nature between the two rows assures low levels of overlap.

V. ITERATIVE DECODING

Unfortunately, bit interleaving destroyed the trellis structure and the markovity of the received vector. Any trellis based decoder requires likelihoods $\mathbb{P}[y|b_{1,j}, b_{2,j}]$, where $b_{1,j}$ and $b_{2,j}$ are bits produced by the natural space-time encoder at the same time instance j, and y is the received vector. What is measured at the receiver is $\mathbb{P}[y|b_{1,j}, \Pi(b_{2,j})]$ because after interleaving, $\Pi(b_{2,j})$ is transmitted in place of $b_{2,j}$. Therefore it is necessary to "decouple" likelihoods $\mathbb{P}[y|b_{1,j}, \Pi(b_{2,j})]$ into $\mathbb{P}[y|b_{1,j}]$ and $\mathbb{P}[y|\Pi(b_{2,j})]$, for each j. These are further deinterleaved and recombined to produce likelihood $\mathbb{P}[y|b_{1,j}, b_{2,j}]$, which is required by the MAP block. Referring to Fig. 6, the Bit Prob Gen block first generates $\mathbb{P}[y|b_{1,j}, \Pi(b_{2,j})]$ from channel observations simply by measuring Euclidean distances between every posssible constellation point $\alpha_1 c_{1,j} + \alpha_2 \Pi(c_{2,j})$ and the cooresonding channel observations y_j. Furthermore, decoupling is done in Bit Prob Gen block using the formula:

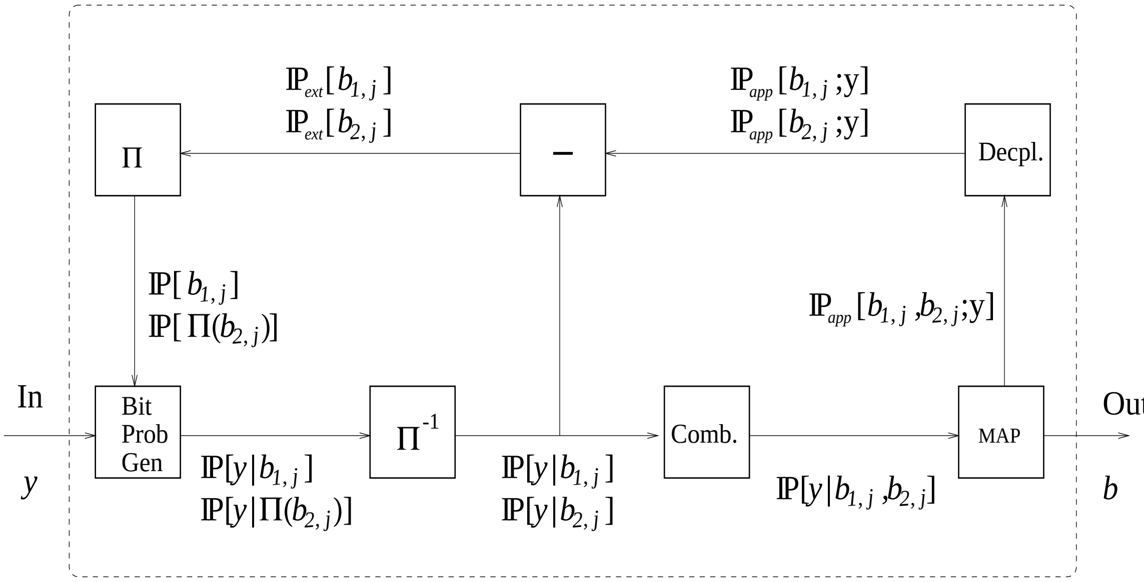

Fig. 6. Iterative Decoder

$$
\begin{aligned}
\mathbb{P}[y|b_{1,j}] &\overset{(a)}{=} \mathbb{P}[y,\ \Pi(b_{2,j})=0|b_{1,j}] + \mathbb{P}[y,\Pi(b_{2,j})=1|b_{1,j}] \\
&\overset{(b)}{=} \mathbb{P}[y|b_{1,j},\Pi(b_{2,j})=0]\mathbb{P}[\Pi(b_{2,j})=0|b_{1,j}] + \mathbb{P}[y|b_{1,j},\Pi(b_{2,j})=1]\mathbb{P}[\Pi(b_{2,j})=1|b_{1,j}] \\
&\overset{(c)}{=} \mathbb{P}[y|b_{1,j},\Pi(b_{2,j})=0]\mathbb{P}[\Pi(b_{2,j})=0] + \mathbb{P}[y|b_{1,j},\Pi(b_{2,j})=1]\mathbb{P}[\Pi(b_{2,j})=1].
\end{aligned} \tag{7}
$$

Here, (a) demonstrates how we obtain marginal probability from the joint by summing out possibilities for one random variable; (b) follows from straightforward application of Bayes' rule, and (c) is due to the assumed independence between $b_{1,j}$ and $\Pi(b_{2,j})$. Actual computation is done using the (c) part of the last equation. $\mathbb{P}[\Pi(b_{2,j})]$ is the *a-priori* information on bit $\Pi(b_{2,j})$. Initially, $\mathbb{P}[\Pi(b_{2,j})=0]$ and $\mathbb{P}[\Pi(b_{2,j})=1]$ are set to $1/2$ and they are updated later to improve estimates in subsequent iterations, as in Fig. 6. Similary, Bit Prob Gen computes $\mathbb{P}[y|\Pi(b_{2,j})]$. $\mathbb{P}[y|b_{1,j}]$ and $\mathbb{P}[y|\Pi(b_{2,j})]$ are further deinterleaved into $\mathbb{P}[y|b_{1,j}]$ and $\mathbb{P}[y|b_{2,j}]$ in preparation for the combining block. Note that $b_{1,j}$ and $b_{2,j}$ are independent i.e., we can write $\mathbb{P}[b_{1,j},b_{2,j}] = \mathbb{P}[b_{1,j}]\mathbb{P}[b_{2,j}]$. Also note that they are *not* independent when conditioned upon y. In general, it is not true that $\mathbb{P}[b_{1,j},b_{2,j}|y] = \mathbb{P}[b_{1,j}|y]\mathbb{P}[b_{2,j}|y]$. We still do make this assumption based on the pseudo-random interleaver properties - transmissons of $b_{1,j}$ and $b_{2,j}$ are separated enough in time.

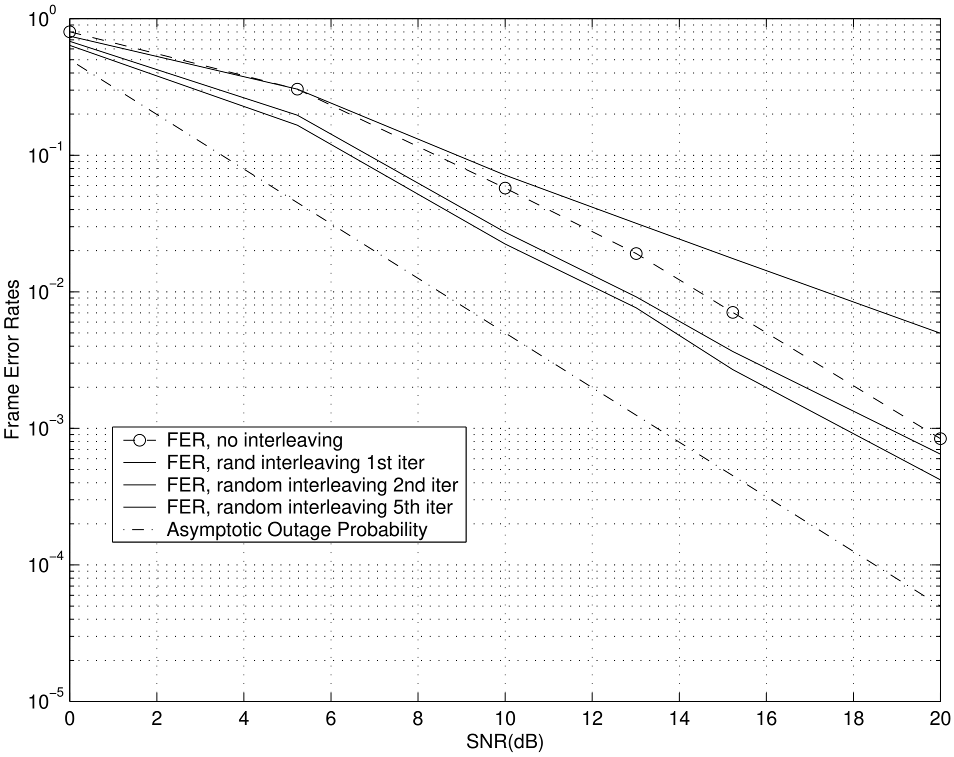

Fig. 7. Frame Error Rates for 4-state code, with and without interleaving

From here it follows that

$$\mathbb{P}[y|b_{1,j}, b_{2,j}] = \frac{\mathbb{P}[y|b_{1,j}]\mathbb{P}[y|b_{2,j}]}{\mathbb{P}[y]} \tag{8}$$

This combining is done in the Comb. block. Also, $\mathbb{P}[y]$ will not influence the MAP decision because it is independent of bits and hence we can drop it in the above computation. Likelihoods $\mathbb{P}[y|b_{1,j}, b_{2,j}]$ are further fed into the MAP block which generates $\mathbb{P}_{app}[b_{1,j}, b_{2,j}; y]$ as a trellis based estimate on $b_{1,j}$ and $b_{2,j}$ - the soft output. Here *app* stands for *a-posteriori* probability. Based on $\mathbb{P}_{app}[b_{1,j}, b_{2,j}; y]$ we compute marginals $\mathbb{P}_{app}[b_{1,j}; y]$ and $\mathbb{P}_{app}[b_{2,j}; y]$ in the Decpl. block. In the next block, with the $-$ sign, we take away the effect of the each individual bit $b_{1,j}$ or $b_{2,j}$ from *before* entering the MAP. Considering the MAP algorithm this turns out to be division if we operate with likelihoods or mere probabilities and subtraction if we operate with log-likelihoods. What we have left is the true bit extrinsic value - the bit probability based on all other bits in the trellis. This is important because we do not want the likelihood $\mathbb{P}[y|b_{1,j}]$ to influence the extrinsic value on the bit $b_{1,j}$ in the next iteration. The result is further re-interleaved and fed into the Bit Prob Gen block

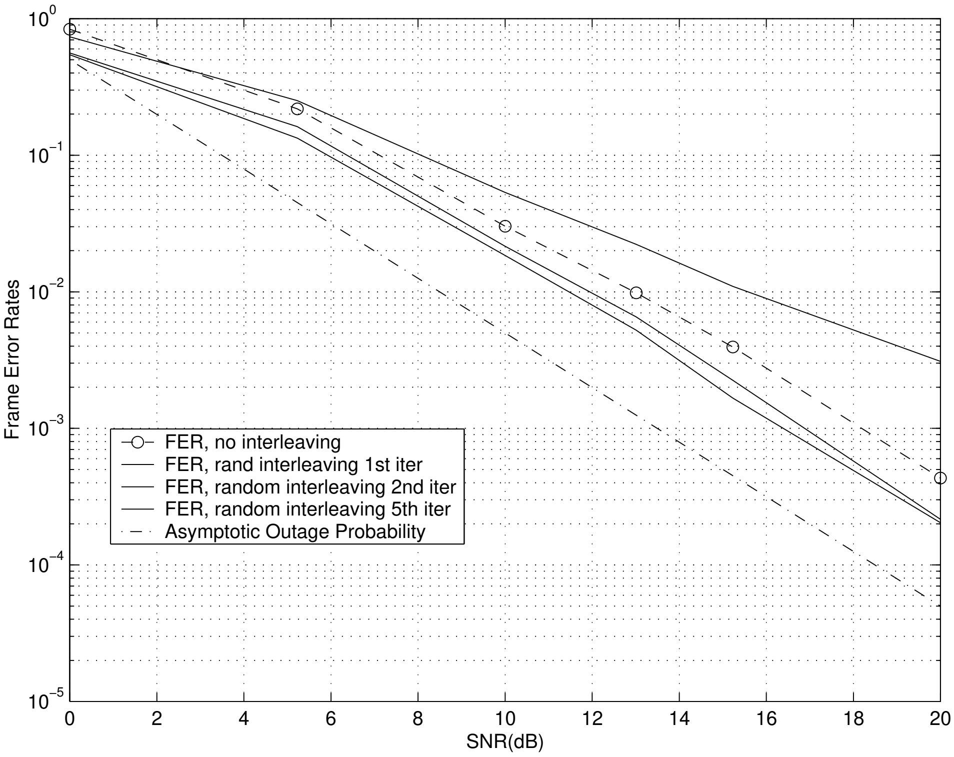

Fig. 8. Frame Error Rates for 16-state code, with and without interleaving

as an *a-priori* information to be used in equation (7) for the next iteration. This is a somewhat standard way of decomposing detection and decoding; decoders using similar principles can be found in [5], [6].

VI. RESULTS

The results are presented for 4-state, 16-state and 64-state in Fig. 7, Fig. 8, and Fig. 9 respectively. Simulations were ran for the frame length of 600 information bits. Non-interleaved codes, as proposed in [3] are compared with bit interleaved codes with iterative decoding. Because there is only one coder, iterations converge rapidly, and saturation occurs after about three iterations. For lower complexity coders, namely 4 and 16 state, we observe 2dB gain as compared to the non-interleaved system. Note that the bit interleaved 16-state coder operates 3dB away from the outage probability. For higher complexity coders (64 state) we see that interleaving does not yield as significant gains, namely about 1dB. This stems from the fact that for coders with larger constraint memory, d_1 and d_2 are already large enough so that an increase in d_3 obtained by interleaving does not

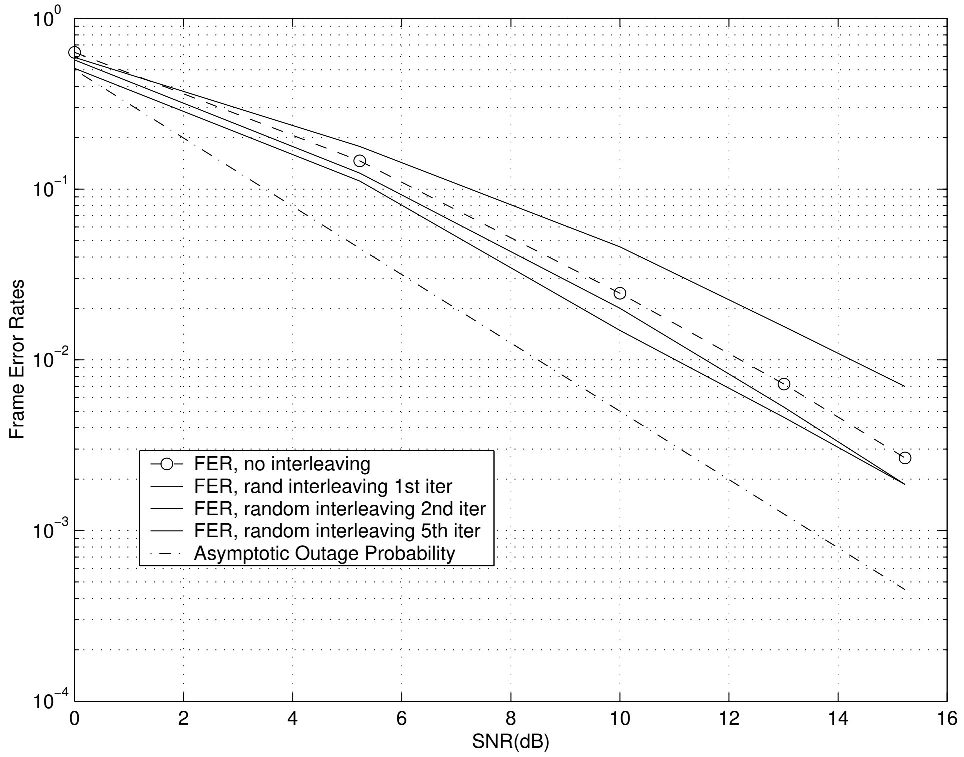

Fig. 9. Frame Error Rates for 64-state code, with and without interleaving

significantly impact the metric d^2. A major part of the decoder complexity lies in the MAP block, and hence the complexity of the decoder is approximately the number of iterations times the complexity of the MAP block. In comparison to non-interleaved case, for three times the decoder complexity we gain about 2dB in performance.

VII. CONCLUSION

We have demonstrated how bit interleaving can significantly improve the performance of space-time codes. However, this pseudo-randomization of one antenna with respect to another destroys the trellis structure of observed symbols at the receiver. Therefore, we resort to iterative decoding. The decoder converges in about three iterations and there are varying gains as compared to non interleaved encoder. The best gain, 2dB, is observed for simpler encoders.

VIII. ACKNOWLEDGEMENT

The authors thank Behnaam Aazhang for extremely valuable comments and discussions.

REFERENCES

[1] A. Wittenben "A new bandwidth efficient transmit antenna modulation diversity scheme for linear digital modulation," *Proc IEEE ICC,* 1993 pp. 1630-1634

[2] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-Time Codes for High Data Rate Wireless Communication: Performance Criterion and Code Construction," *IEEE Transactions on Information Theory*, vol. 44, No. 2, pp. 744–765, March 1998.

[3] A. R. Hammons and H. E. Gamal "On the Theory of Space-Time Codes for PSK Modulation," *IEEE Transactions on Information Theory*, vol. 2, No. 2, pp. 524–542, March 2000.

[4] D. Divsalar and M. K. Simon, "The design of trellis coded MPSK for fading channels: Performance criteria" *IEEE Transactions on Information Theory*, vol. 45, pp.1456-1467, July 1989.

[5] Y. Liu, M. P. Fitz, and O. Y. Takeshita, "QPSK space-time turbo codes," in *IEEE ICC*, June 2000.

[6] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding using soft feedback," *Electronic Letters*, vol. 34, pp. 942–943, 4 March 1998.

[7] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding," *Proc IEEE ICC*, vol. 2, pp. 858–863, June 1999.

[8] X. Li and J. A. Ritcey, "Trellis-coded modulation with bit interleaving and iterative decoding," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 715–724, April 1999.

[9] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding," *IEEE Communications Letters*, vol. 1, pp. 169–171, November 1997.

[10] E. Telatar, "Capacity of multi-antenna Gaussian channels,", AT&T-Bell Laboratories Internal Tech. Memo., June 1995.

[11] S. M. Alamouti, "A simple diversity scheme for wireless communications," *IEEE J. Select. Areas Commun.,* vol. 16, pp.1451-1458, Oct. 1998.