Wireless Pers Commun (2017) 97:4789–4811
DOI 10.1007/s11277-017-4751-9

Linear Prediction-Based Detection of Serially Concatenated DQPSK in SIMO-OFDM

Vineel K. Veludandi[1] · **K. Vasudevan**[1]

Published online: 10 August 2017

Abstract An approach for the linear prediction-based detection of serially concatenated turbo coded differential quaternary phase shift keyed signals, is presented for SIMO-OFDM systems. The proposed method exploits the high degree of correlation in the channel frequency response, when the channel impulse response is much smaller than the FFT length. A prediction filter is used to estimate the channel frequency response. At the receiver, the inner decoder operates on a supertrellis with just $S_{ST} = S_E \times 2^{P-1}$ states, where the complexity reduction is achieved by using the concept of isometry (here S_E denotes the number of states in the encoder trellis and P denotes the prediction order). A reduced complexity version based on *per-survivor processing* for the proposed receiver is also given. Simulation results for the proposed method is compared to the case where the channel is estimated using pilots in the frequency domain. Though the BER performance of the channel estimation approach using pilots is much better than the proposed method, its throughput is much lower, since the pilots have to be transmitted for every Orthogonal Frequency Division Multiplexing (OFDM) frame. However, in the proposed approach, the pilots are required only for the first OFDM frame for the purpose of estimating the statistical properties of the channel frequency response and noise. For the rest of the frames, pilots are not required since, the channel and noise statistics are assumed to be unchanged. Therefore, the proposed method is throughput efficient.

Keywords Linear prediction · Turbo principle · Supertrellis · RS-BJCR · Serially concatenated convolutional codes · Isometry · OFDM

✉ Vineel K. Veludandi
vineelkv@iitk.ac.in

K. Vasudevan
vasu@iitk.ac.in

[1] Department of EE, IIT Kanpur, Kanpur, India

1 Introduction

Communication systems aim for near capacity performance [6–9]. Orthogonal Frequency Division Multiplexing (OFDM) converts a frequency selective channel into a frequency flat channel, resulting in a simple detection rule at the receiver. Near capacity performance is achieved using coherent detection [6–9] and the turbo principle [4], and is efficiently implemented using soft-input, soft-output decoders. It is well known that for a given SNR, coherent detection performs better than non-coherent detection, in terms of the bit-error-rate (BER) [10]. However, coherent detection requires carrier and timing synchronization and also accurate channel estimation. In this work, we address the problem of data detection in serially concatenated turbo coded OFDM systems using the concept of channel estimation via linear prediction. The channel prediction filter exploits the high degree of correlation in the channel frequency response at the output of the FFT in the OFDM receiver when the channel impulse response is much smaller than the FFT length. Perfect timing and carrier synchronization is assumed.

In this work, the inner MAP decoder operates on a supertrellis [5, 10] resulting from the memory of the inner recursive systematic convolutional encoder and a prediction filter. This approach is based on a posteriori probability (APP) based channel estimation [1, 12]. However, the novelty of this work lies in the use of a supertrellis obtained by making use of isometry [11]. The inner decoder has both error correcting capability and also the ability to perform channel estimation (prediction). The proposed method is compared to the case where the channel is estimated using pilots in the frequency domain. Though the BER performance of the channel estimation approach using pilots is much better than the proposed method, its throughput is much lower, since the pilots have to be transmitted for every OFDM frame. However, in the proposed approach, the pilots are required only for the first OFDM frame for the purpose of estimating the statistical properties of the channel frequency response and noise. For the rest of the frames, pilots are not required since, the channel and noise statistics remain same. Therefore, the proposed method is throughput efficient.

This paper is organized as follows. The notation used throughout this paper is given in Sect. 2. The system model is given in Sect. 3. The proposed linear prediction-based receiver is discussed in Sect. 4. In Sect. 5, we present the pilot-based receiver. The simulation results, throughput and complexity comparision are given in Sect. 6. Finally Sect. 7 concludes the paper.

2 Notation

In this paper, all lower-case and upper-case letters without a tilde e.g. d_k represent real-valued scalar. Letters with a tilde e.g. $\tilde{h}_k$, denote complex quantities. However, complex symbols are denoted by S_k (without a tilde). Boldface letters represent vectors or matrices. All letters with a hat, e.g. $\hat{X}_k$ denote the statistical estimate of $\tilde{X}_k$ (or X_k, if it is real-valued). The $(\cdot)^*$ denotes complex conjugate, $(\cdot)^{\mathscr{H}}$ denotes conjugate transpose and $E[\cdot]$ denotes the expectation operation. We also assume that bit 0 maps to $+1$ and bit 1 maps to -1 [binary phase shift keying (BPSK)]. Recall that QPSK is just BPSK signalling in the in-phase and quadrature arms.

3 System Model

3.1 Transmitter

The frame structure is given in Fig. 1 and system model is given in Fig. 2. The binary input data g_k $(0 \le k \le L_d/2 - 1)$ from the source is encoded using an outer rate-1 / 2 recursive systematic convolutional (RSC) encoder. The encoded data b_k $(0 \le k \le L_d - 1)$ is passed through a random interleaver. The interleaved bits c_k $(0 \le k \le L_d - 1)$ is again encoded using an inner rate-1 / 2 recursive systematic convolutional encoder to get d_k $(0 \le k \le 2L_d - 1)$. The generator matrix for both the inner and outer encoder is:

$$G(D) = \left[1 \quad \frac{1 + D^2}{1 + D + D^2}\right]. \tag{1}$$

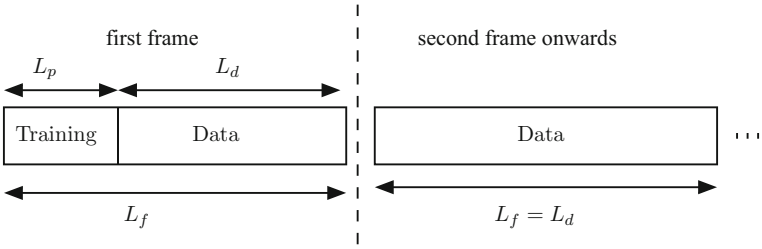


Fig. 1 OFDM frames in the frequency domain for the proposed linear prediction-based approach

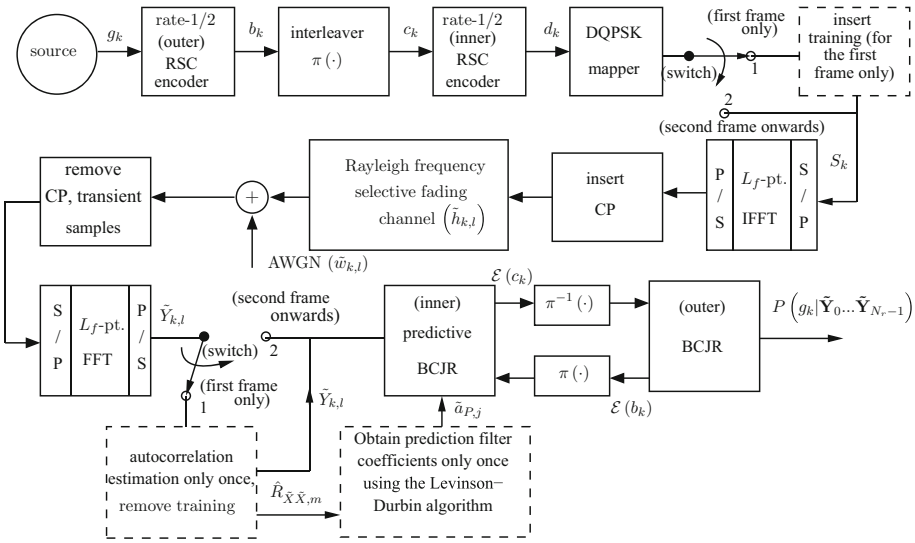


Fig. 2 Block diagram for SIMO-OFDM. Two consecutive bits of d_k are mapped to QPSK using the differential encoding rules in Table 1. Note that the switch will be in position 1 for the first OFDM frame only and moves to the position 2 from the second OFDM frame onwards

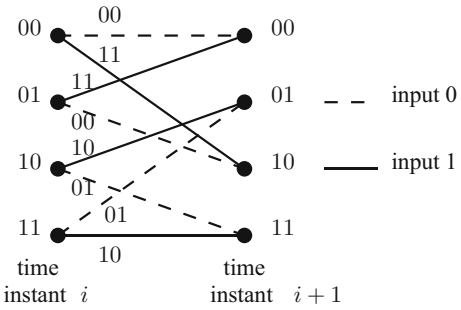


Fig. 3 Trellis for the encoder in (1)

The trellis diagram of the encoder is shown in Fig. 3. The inner encoded data is mapped to differential quadrature phase shift keying (DQPSK) according to the differential encoding rules [10] given in Table 1. A training sequence of length L_p is inserted in the DQPSK symbol stream as shown in Fig. 1 to get S_k $\left(0 \le k \le L_f - 1\right)$, where S_k $\left(L_p \le k \le L_f - 1\right)$ denotes the data. Note that this training sequence is inserted in the first OFDM frame only. The symbol stream S_k is fed to a serial to parallel converter (S/P) and modulated on to the OFDM sub-carriers by an L_f-point IFFT operation. The length of the cyclic prefix (CP) is equal to the length of the channel memory $L_{\mathrm{CP}} = L_h - 1$ [10], and is inserted into the OFDM frame. Note that the overall rate of the transmitter in Fig. 2 is 1 / 2.

3.2 Channel Model

We assume a Rayleigh frequency selective fading channel having a uniform power delay profile [6, 8]. The channel is assumed to be time-invariant over each OFDM frame and varies independently from frame to frame i.e. quasistatic. For the lth diversity arm $(0 \le l \le N_r - 1)$, the channel impulse response $\tilde{h}_{k,l}$ $(0 \le k \le L_h - 1)$ and AWGN noise $\tilde{w}_{k,l}$ $(0 \le k \le L_f - 1)$ are both wide-sense stationary (WSS) circularly symmetric complex Gaussian random variables with autocorrelation given by:

$$\begin{aligned} \frac{1}{2}E\left[\tilde{h}_{k,l}\tilde{h}^*_{k',l'}\right] &= \begin{cases} \sigma_f^2, & \text{if } k = k' \text{ and } l = l' \\ 0, & \text{otherwise} \end{cases} \\ \frac{1}{2}E\left[\tilde{w}_{k,l}\tilde{w}^*_{k',l'}\right] &= \begin{cases} \sigma_w^2, & \text{if } k = k' \text{ and } l = l' \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{2}$$

Table 1 Differential encoding rules

Dibit $(d_{k-1}d_k)$	Decimal equivalent of the dibit $(\mathscr{S}_{0,j})$	Phase change (in radians)
00	0	0
01	1	$\pi/2$
10	2	$3\pi/2$
11	3	π

Note that $L_h \ll L_f$, that is, the channel impulse response is much smaller than the FFT size. The parameters σ_f^2, σ_w^2 and L_h are not known at the receiver (L_h is not exactly known since there may be leading and trailing zeros).

4 Linear Prediction-Based Receiver

The output of the FFT operation at the lth diversity arm in the receiver is given by:

$$\tilde{Y}_{k,l} = \tilde{H}_{k,l} S_k + \tilde{W}_{k,l}, \quad 0 \le k \le L_f - 1, \ 0 \le l \le N_r - 1 \tag{3}$$

where

$$\begin{aligned} \tilde{H}_{k,l} &= \sum_{i=0}^{L_h-1} \tilde{h}_{i,l} e^{-\mathrm{j}2\pi ik/L_f} \\ \tilde{W}_{k,l} &= \sum_{i=0}^{L_f-1} \tilde{w}_{i,l} e^{-\mathrm{j}2\pi ik/L_f}. \end{aligned} \tag{4}$$

The 1-D variance of the channel frequency response samples is [6]

$$\frac{1}{2} E\left[\tilde{H}_{k,l} \tilde{H}_{k,l}^*\right] = L_h \sigma_f^2 \tag{5}$$

and the 1-D variance of the DFT of the AWGN samples is [6]

$$\frac{1}{2} E\left[\tilde{W}_{k,l} \tilde{W}_{k,l}^*\right] = L_f \sigma_w^2. \tag{6}$$

The autocorrelation of $\tilde{H}_{k,l}$ is given by [6]:

$$\tilde{R}_{\tilde{H}\tilde{H},m} \stackrel{\Delta}{=} \frac{1}{2} E\left[\tilde{H}_{k,l} \tilde{H}_{k-m,l}^*\right] = \sigma_f^2 \sum_{n=0}^{L_h-1} e^{-\mathrm{j}2\pi nm/L_f}. \tag{7}$$

Now consider

$$\tilde{X}_{k,l} = \tilde{Y}_{k,l} / S_k. \tag{8}$$

During training S_k is known and during data detection S_k is obtained from the supertrellis state which will be explained in Sect. 4.3.

The autocorrelation of $\tilde{X}_{k,l}$ is given by:

$$\tilde{R}_{\tilde{X}\tilde{X},m} \stackrel{\Delta}{=} \frac{1}{2} E\left[\tilde{X}_{k,l} \tilde{X}_{k-m,l}^*\right] = \tilde{R}_{\tilde{H}\tilde{H},m} + \frac{\sigma_w^2 L_f}{|S_k|^2} \delta_{K,m} \tag{9}$$

where $\delta_{K,m}$ is the Kronecker delta function defined as:

$$\delta_{K,m} = \begin{cases} 1, & \text{if } m = 0, \\ 0, & \text{if } m \neq 0. \end{cases} \tag{10}$$

The key idea behind this approach is to estimate (predict) $\tilde{X}_{k,l}$ assuming S_k is known, as follows [10]:

$$\hat{X}_{k,l} = -\sum_{j=1}^{P} \tilde{a}_{P,j} \tilde{X}_{k-j,l} \tag{11}$$

where $\tilde{a}_{P,j}$ denotes the jth coefficient of the optimum Pth-order predictor. Note that $\tilde{X}_{k,l} \approx \tilde{H}_{k,l}$ at high SNR.

The prediction error is [10]:

$$\tilde{z}_{k,l} \stackrel{\Delta}{=} \tilde{X}_{k,l} - \hat{X}_{k,l} = \sum_{j=0}^{P} \tilde{a}_{P,j} \tilde{X}_{k-j,l} \tag{12}$$

and the 1-D prediction error variance is given by [10]:

$$\sigma_P^2 \stackrel{\Delta}{=} \frac{1}{2} E\left[\left|\tilde{z}_{k,l}\right|^2\right] = \sum_{j=0}^{P} \tilde{a}_{P,j} \tilde{R}_{\tilde{X}\tilde{X},-j} \tag{13}$$

where $\tilde{a}_{P,0} = 1$. In Sect. 4.2 we give a formal derivation of the linear prediction-based receiver.

In practice of course, the autocorrelation $\tilde{R}_{\tilde{X}\tilde{X},m}$ required for generating the prediction filter coefficients is not known. Hence we estimate the autocorrelation $\tilde{R}_{\tilde{X}\tilde{X},m}$ using the training sequence $S_k \ \left(0 \le k \le L_p - 1\right)$ from the first OFDM frame. This training sequence is transmitted only once during the simulations.

4.1 Estimating the Autocorrelation $\tilde{R}_{\tilde{X}\tilde{X},m}$

Since the training sequence $S_k \ \left(0 \le k \le L_p - 1\right)$ from the first OFDM frame is known to the receiver, the autocorrelation of $\tilde{X}_{k,l}$ can be estimated as [2]:

$$\begin{aligned} \hat{R}_{\tilde{X}\tilde{X},m} &\approx \frac{1}{2N_r\left(L_p - m\right)} \sum_{l=0}^{N_r-1} \sum_{i=0}^{L_p-1-m} \frac{\tilde{Y}_{i+m,l}}{\tilde{S}_{i+m,l}} \left(\frac{\tilde{Y}_{i,l}}{\tilde{S}_{i,l}}\right)^* \\ &\approx \frac{1}{2N_r\left(L_p - m\right)} \sum_{l=0}^{N_r-1} \sum_{i=0}^{L_p-1-m} \tilde{X}_{i+m,l} \tilde{X}_{i,l}^*, \quad m \ge 0. \end{aligned} \tag{14}$$

After estimating the autocorrelation, the predictor coefficients $\tilde{a}_{P,j}$ can be obtained using the Levinson–Durbin algorithm [10].

4.2 The Suboptimal Predictive MAP Decoder [11]

We now derive the principle of operation of the inner MAP decoder in Fig. 2.

After discarding the training sequence, the received sequence at the output of the FFT in Fig. 2 at each diversity arm can be represented as:

$$\tilde{\mathbf{Y}}_l = \mathbf{S}^{(q)} \tilde{\mathbf{H}}_l + \tilde{\mathbf{W}}_l, \ 0 \le q \le 2^{L_d} - 1, \ 0 \le l \le N_r - 1 \tag{15}$$

where $\tilde{\mathbf{Y}}_l$ is an $L_d \times 1$ column vector of received samples, $\mathbf{S}^{(q)}$ is an $L_d \times L_d$ diagonal matrix with elements containing the qth possible QPSK symbol sequence, $\tilde{\mathbf{H}}_l$ is an $L_d \times 1$ column vector of the channel DFT and $\tilde{\mathbf{W}}_l$ is an $L_d \times 1$ column vector containing the DFT of the AWGN samples $\tilde{w}_{k,l}$ in Fig. 2.

The MAP detector decides in favour of $\mathbf{S}^{(q)}$ that maximizes the joint conditional pdf

$$\begin{aligned} &\max_q \; P\left(\mathbf{S}^{(q)}|\tilde{\mathbf{Y}}_0, \tilde{\mathbf{Y}}_1, \ldots, \tilde{\mathbf{Y}}_{N_r-1}\right) \\ \Rightarrow \; &\max_q \frac{p\left(\tilde{\mathbf{Y}}_0, \tilde{\mathbf{Y}}_1, \ldots, \tilde{\mathbf{Y}}_{N_r-1}|\mathbf{S}^{(q)}\right)P\left(\mathbf{S}^{(q)}\right)}{p\left(\tilde{\mathbf{Y}}_0, \tilde{\mathbf{Y}}_1, \ldots, \tilde{\mathbf{Y}}_{N_r-1}\right)} \\ \Rightarrow \; &\max_q \prod_{l=0}^{N_r-1} p\left(\tilde{\mathbf{Y}}_l|\mathbf{S}^{(q)}\right)P\left(\mathbf{S}^{(q)}\right) \end{aligned} \tag{16}$$

where $P(\cdot)$ denotes probability and $p(\cdot)$ denotes the probability density function, and we have assumed that $\tilde{\mathbf{H}}_l$ and $\tilde{\mathbf{W}}_l$ are independent over l.

Note that the denominator in the second equation of (16) is independent of q, and hence can be ignored. Substituting for the conditional pdfs in the last equation of (16), we get

$$\max_q \; \exp\left(-\frac{1}{2}\sum_{l=0}^{N_r-1} \tilde{\mathbf{Y}}_l^{\mathscr{H}} \left(\tilde{\mathbf{R}}^{(q)}\right)^{-1} \tilde{\mathbf{Y}}_l\right) P\left(\mathbf{S}^{(q)}\right) \tag{17}$$

where

$$\begin{aligned} \tilde{\mathbf{R}}^{(q)} \triangleq \frac{1}{2}E\left[\tilde{\mathbf{Y}}_l\tilde{\mathbf{Y}}_l^{\mathscr{H}}|\mathbf{S}^{(q)}\right] &= \frac{1}{2}\mathbf{S}^{(q)}E\left[\tilde{\mathbf{H}}_l\tilde{\mathbf{H}}_l^{\mathscr{H}}\right]\left(\mathbf{S}^{(q)}\right)^{\mathscr{H}} + \sigma_w^2 L_f \mathbf{I} \\ &\approx \frac{1}{2}\mathbf{S}^{(q)}E\left[\tilde{\mathbf{H}}_l\tilde{\mathbf{H}}_l^{\mathscr{H}}\right]\left(\mathbf{S}^{(q)}\right)^{\mathscr{H}} \text{ (at high SNR)} \\ &= \mathbf{S}^{(q)}\mathbf{\Phi}\left(\mathbf{S}^{(q)}\right)^{\mathscr{H}} \text{ (say)}. \end{aligned} \tag{18}$$

Now, by applying Cholesky decomposition of the autocovariance matrix $\mathbf{\Phi}$, it can be shown that [10]

$$\mathbf{\Phi}^{-1} = \tilde{\mathbf{B}}^{\mathscr{H}}\mathbf{D}^{-1}\tilde{\mathbf{B}} \tag{19}$$

where

$$\tilde{\mathbf{B}} \triangleq \begin{bmatrix} 1 & 0 & \ldots & 0 \\ \tilde{a}_{1,1} & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{a}_{L_d-1,L_d-1} & \tilde{a}_{L_d-1,L_d-2} & \ldots & 1 \end{bmatrix} \tag{20}$$

is the $(L_d \times L_d)$ matrix of predictor coefficients with $\tilde{a}_{i,\tau}$ being the τth coefficient of the optimum ith-order predictor and the $(L_d \times L_d)$ matrix

$$\mathbf{D} \triangleq \begin{bmatrix} \sigma_0^2 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \sigma_{L_d-1}^2 \end{bmatrix} \tag{21}$$

where σ_k^2 is the 1-D prediction error variance of the optimum kth-order predictor and $\sigma_0^2 = \tilde{R}_{\tilde{H}\tilde{H},0} \approx \tilde{R}_{\tilde{X}\tilde{X},0}$ at high SNR.

We emphasize once again that the predictor coefficients in (20) can be obtained during training, using the Levinson-Durbin algorithm and the estimated autocorrelation in (14). Similarly, the prediction error variances in (21) can be obtained from (13) and (14).

Now, the maximization rule in (17) can be expressed as

$$\max_q \exp\left(-\frac{1}{2}\sum_{l=0}^{N_r-1}\tilde{\mathbf{Y}}_l^{\mathscr{H}}\left(\left(\mathbf{S}^{(q)}\right)^{\mathscr{H}}\right)^{-1}\mathbf{B}^{\mathscr{H}}\mathbf{D}^{-1}\mathbf{B}\left(\mathbf{S}^{(q)}\right)^{-1}\tilde{\mathbf{Y}}_l\right)P\left(\mathbf{S}^{(q)}\right).$$

Since there is a one–one mapping between the sequence $c_k^{(q)}$ and $S_k^{(q)}$ in Fig. 2 for $(0 \le k \le L_d - 1)$, the maximization rule can be simplified to (note that $c_k^{(q)}$ is independent over k, whereas $S_k^{(q)}$ is not independent over k since it is coded)

$$\begin{aligned}&\max_q \exp\left(-\frac{1}{2}\sum_{l=0}^{N_r-1}\tilde{\mathbf{Y}}_l^{\mathscr{H}}\left(\left(\mathbf{S}^{(q)}\right)^{\mathscr{H}}\right)^{-1}\mathbf{B}^{\mathscr{H}}\mathbf{D}^{-1}\mathbf{B}\left(\mathbf{S}^{(q)}\right)^{-1}\tilde{\mathbf{Y}}_l\right)P\left(\mathbf{c}^{(q)}\right)\\ &\Rightarrow \max_q \exp\left(-\sum_{l=0}^{N_r-1}\sum_{k=0}^{L_d-1}\frac{\left|\tilde{z}_{k,l}^{(q)}\right|^2}{2\sigma_k^2}\right)\prod_{k=0}^{L_d-1}P\left(c_k^{(q)}\right)\end{aligned} \tag{22}$$

where the prediction error $\tilde{z}_{k,l}^{(q)}$ is an element of

$$\begin{bmatrix}\tilde{z}_{0,l}^{(q)}\\ \tilde{z}_{1,l}^{(q)}\\ \vdots\\ \tilde{z}_{L_d-1,l}^{(q)}\end{bmatrix} \triangleq \tilde{\mathbf{Z}}^{(q)} = \mathbf{B}\left(\mathbf{S}^{(q)}\right)^{-1}\tilde{\mathbf{Y}}_l. \tag{23}$$

Assuming that a Pth-order predictor completely decorrelates noise, (22) can be written as

$$\max_q \exp\left(-\sum_{l=0}^{N_r-1}\sum_{k=0}^{P-1}\frac{\left|\tilde{z}_{k,l}^{(q)}\right|^2}{2\sigma_k^2}-\sum_{l=0}^{N_r-1}\sum_{k=P}^{L_d-1}\frac{\left|\tilde{z}_{k,l}^{(q)}\right|^2}{2\sigma_P^2}\right)\prod_{k=0}^{L_d-1}P\left(c_k^{(q)}\right). \tag{24}$$

Note that the first double summation in (24) denotes the "transient" part and the second double summation denotes the "steady state" part. Observe also that the predictor coefficients in (20) correspond to the autocorrelation of the channel frequency response. In practice, the predictor coefficients are obtained from the autocorrelation of $\tilde{X}_{k,l}$, as given in (14). Finally we note that the complexity in (24) increases exponentially with L_d. In the Sect. 4.4, we present the predictive BCJR algorithm, whose complexity increases linearly with L_d.

4.3 Supertrellis Construction [10]

In Sect. 4 [just after (8)] we had mentioned that the symbols S_k are not known, and in practice they are obtained from a supertrellis. In this section we discuss the construction of the supertrellis.

Consider the inner decoder (predictive BCJR) in Fig. 2. The inner decoder trellis must be modified to a supertrellis which incorporates the memory of the prediction filter.

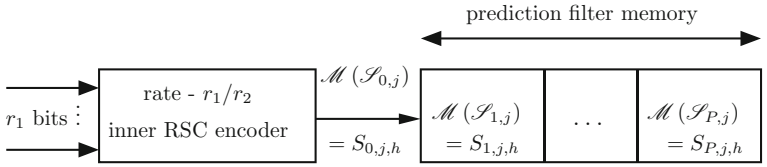


Fig. 4 Procedure for constructing the supertrellis [10], for a *P*th-order predictor

Assume that the r_2 coded bits from the inner rate-r_1/r_2 convolutional encoder (in this paper $r_1 = 1$, $r_2 = 2$) are mapped[1] to an M-ary ($M = 2^{r_2}$) constellation according to the set partitioning rule, e.g. $S_{0,j,h} = \mathscr{M}(\mathscr{S}_{0,j})$ [10]. Here, $\mathscr{S}_{0,j}$ $(0 \le \mathscr{S}_{0,j} \le 2^{r_2} - 1)$ is the decimal equivalent of the r_2 coded bits $d_k d_{k-1} \ldots d_{k-r_2+1}$ in Fig. 2, and is referred to as the input code digit. Note that since the supertrellis is a periodic structure, we have removed the time index k in S_k [see (8)] and replaced it with $S_{i,j,h}$ (see Fig. 4). The subscript "i" in $S_{i,j,h}$ refers to the ith memory element of the prediction filter (the 0th element is the input), the subscript "j" refers to the present supertrellis state and "h" denotes the next supertrellis state. Observe that the symbol sequence S_k in Fig. 2 corresponds to one of the paths through the supertrellis.

Now consider Fig. 4. Note that

$$\{\mathscr{M}(\mathscr{S}_{0,j}), \mathscr{M}(\mathscr{S}_{1,j}), \ldots, \mathscr{M}(\mathscr{S}_{P,j})\} \tag{25}$$

in Fig. 4 is a valid encoded symbol sequence. In (25), the subscript P refers to a Pth-order prediction filter and j refers to the jth supertrellis state, as will be explained later.

Let S_E denote the number of encoder states. For any given present convolutional encoder state $\mathscr{E}_i$ $(0 \le i \le S_E - 1)$, there are $2^{r_1} = N$ possible encoded symbols. Hence, starting from any particular encoder state, there are N^P ways in which a prediction filter of order P can be populated (see Fig. 4). Therefore, the total number of ways in which a Pth-order predictor can be populated is $S_E \times N^P$ which is also equal to the number of supertrellis states. Therefore

$$S_{ST} = S_E \times N^P. \tag{26}$$

Let $\mathscr{F}_m$ $(0 \le m \le N^P - 1)$ denote the prediction filter state. Supertrellis state is given by $\mathscr{S}_{ST,j}$, where

$$j = i \times N^P + m, \quad 0 \le j \le S_E \times N^P - 1. \tag{27}$$

Symbolically, the supertrellis state can be represented as:

$$\mathscr{S}_{ST,j} : \{\mathscr{E}_i; \mathscr{F}_m\}. \tag{28}$$

Let us represent the prediction filter state $\mathscr{F}_m$ by an N-ary P-tuple as follows:

$$\mathscr{F}_m : \{\mathscr{N}_{1,m} \ldots \mathscr{N}_{P,m}\} \tag{29}$$

where the input digits are denoted by

$$\mathscr{N}_{t,m} \in \{0, \ldots, N-1\} \quad \text{for } 1 \le t \le P \tag{30}$$

[1] In this section we assume that the DQPSK mapper in Fig. 2 is absent. The need to have a DQPSK mapper will be explained in Sect. 4.4.2.

such that

$$m = \sum_{t=1}^{P} \mathscr{N}_{P+1-t,m} N^{t-1} \tag{31}$$

is the decimal equivalent of the N-ary, P-tuple in (29).

$\mathscr{F}_m$ is actually the input sequence to the encoder in Fig. 4 with $\mathscr{N}_{P,m}$ being the initial input digit. Let $\mathscr{E}_s$ $(0 \le s \le S_{\mathrm{E}} - 1)$ be the encoder state corresponding to the input digit $\mathscr{N}_{P,m}$. The code digit sequence corresponding to the supertrellis state $\mathscr{S}_{\mathrm{ST},j}$ is generated as follows:

$$\begin{aligned} &\mathscr{E}_s, \mathscr{N}_{P,m} \rightarrow \mathscr{E}_a, \mathscr{S}_{P,j} \quad \text{for} \quad 0 \le a < S_{\mathrm{E}},\ 0 \le \mathscr{S}_{P,j} < M \\ &\mathscr{E}_a, \mathscr{N}_{P-1,m} \rightarrow \mathscr{E}_b, \mathscr{S}_{P-1,j} \quad \text{for} \quad 0 \le b < S_{\mathrm{E}},\ 0 \le \mathscr{S}_{P-1,j} < M \end{aligned} \tag{32}$$

which means: the encoder at (starting) state $\mathscr{E}_s$ with input digit $\mathscr{N}_{P,m}$ yields the code digit $\mathscr{S}_{P,j}$ and the next encoder state $\mathscr{E}_a$ and so on. We repeat this procedure till the last input digit, to get:

$$\begin{aligned} &\mathscr{E}_c, \mathscr{N}_{1,m} \rightarrow \mathscr{E}_i, \mathscr{S}_{1,j} \quad \text{for } 0 \le c < S_{\mathrm{E}},\ 0 \le \mathscr{S}_{1,j} < M \\ &\mathscr{E}_i, \mathscr{N}_{0,m} \rightarrow \mathscr{E}_f, \mathscr{S}_{0,j} \quad \text{for } 0 \le \mathscr{N}_{0,m} < N,\ 0 \le \mathscr{S}_{0,j} < M,\ 0 \le f < \mathrm{S_E}. \end{aligned} \tag{33}$$

Thus, the prediction filter is populated with a valid encoded symbol sequence as given in (25).

Now, given the supertrellis state $\mathscr{S}_{\mathrm{ST},j}$ and the input digit $\mathscr{N}_{0,m}$ in (33), the next supertrellis state $\mathscr{S}_{\mathrm{ST},h}$ can be obtained as follows:

$$\begin{aligned} &\mathscr{F}_l : \{\mathscr{N}_{0,m}\mathscr{N}_{1,m}\ldots\mathscr{N}_{P-1,m}\} \\ &h = f \times N^P + l \quad 0 \le f < S_{\mathrm{E}}, \\ &\quad 0 \le l \le N^P - 1,\ 0 \le h \le S_{\mathrm{E}} \times N^P - 1. \end{aligned} \tag{34}$$

To summarize

$$\mathscr{S}_{\mathrm{ST},j}, \mathscr{N}_{0,m} \rightarrow \mathscr{S}_{\mathrm{ST},h}, \mathscr{S}_{0,j} \tag{35}$$

which means: the supertrellis state $\mathscr{S}_{\mathrm{ST},j}$ with input digit $\mathscr{N}_{0,m}$ gives the code digit $\mathscr{S}_{0,j}$ and the next supertrellis state $\mathscr{S}_{\mathrm{ST},h}$. Note also that according to the notation in (28)

$$\mathscr{S}_{\mathrm{ST},h} : \{\mathscr{E}_f; \mathscr{F}_l\}. \tag{36}$$

The supertrellis for a 1st-order predictor ($P = 1$) and the rate-1/2 encoder in (1) is given in Table 2.

4.4 Predictive BCJR for the Inner Code [5]

The BCJR for the inner code operates on the supertrellis. Let $\mathscr{D}_n$ denote the supertrellis states that diverge from the supertrellis state n $(0 \le n \le S_{\mathrm{ST}} - 1)$ and $\mathscr{C}_n$ denote the supertrellis states that converge to the supertrellis state n. The forward sum-of-product (SOP) and the backward SOP at the time i and supertrellis state n are denoted by $\alpha_{i,n}$ $(0 \le i \le L_d - 1)$ and $\beta_{i,n}$ $(1 \le i \le L_d)$.

Table 2 Unnormalized supertrellis for the inner code, for a first-order ($P = 1$) prediction order and a rate $-1/2$ encoder given in (1)

Present supertrellis state j (time n)	Input $\mathscr{N}_{0,m}$	Next supertrellis state h(time $n+1$)
0	0	0
0	1	5
1	0	0
1	1	5
2	0	4
2	1	1
3	0	4
3	1	1
4	0	6
4	1	3
5	0	6
5	1	3
6	0	2
6	1	7
7	0	2
7	1	7

4.4.1 Forward Recursion

Assuming that the receiver knows the starting supertrellis state $\mathscr{S}_{\mathrm{ST},j}$ of the encoder, we first initialize $\alpha_{0,n}$ as:

$$\alpha_{0,n} = \begin{cases} 1, & \text{if } n = j\,, \\ 0, & \text{if } n \neq j \end{cases} \tag{37}$$

and then calculate $\alpha_{i+1,n}$ $(0 \leq i \leq L_d - 2)$ recursively as:

$$\begin{aligned} \alpha'_{i+1,n} &= \sum_{m \in \mathscr{C}_n} \alpha_{i,m} \gamma_{i,m,n} P\left(c_{i,m,n}\right) \\ \alpha_{i+1,n} &= \frac{\alpha'_{i+1,n}}{\sum_{n=0}^{S_{\mathrm{ST}}-1} \alpha'_{i+1,n}} \end{aligned} \tag{38}$$

where $P\left(c_{i,m,n}\right)$ is the a priori probability of the input bit corresponding to the transition from the supertrellis state m to n at time instant i and is set to 0.5 for the first iteration [4]. However, from the second iteration, it takes the values of the interleaved extrinsic information from the outer decoder. The normalization step in the last equation of (38) is required for numerical stability.

From (24), we have

$$\gamma_{i,m,n} = \exp\left(- \sum_{l=0}^{N_r-1} \frac{\left|\tilde{z}_{i,m,n,l}\right|^2}{2\hat{\sigma}_D^2} \right) \tag{39}$$

where

$$D = \min(i, P)$$
$$\tilde{z}_{i,m,n,l} = \sum_{j=0}^{D} \tilde{a}_{D,j} \frac{\tilde{Y}_{i-j,l}}{S_{j,m,n}} \tag{40}$$

where $S_{0,m,n}$ denotes the input symbol corresponding to the transition from supertrellis state m to n and the data $S_{j,m,n}$ are the contents of the prediction filter of supertrellis state m (see Fig. 4), and $\hat{\sigma}_D^2$ is the estimate of σ_D^2, and is obtained from (13) and (14).

4.4.2 *Complexity Reduction of the Supertrellis Using Isometry [11]*

Consider the error signal $\tilde{z}_{i,m,n,l}$ in (39):

$$\begin{aligned} \tilde{z}_{i,m,n,l} &= \tilde{X}_{i,l} - \hat{X}_{i,l} \\ &= \sum_{j=0}^{D} \tilde{a}_{D,j} \frac{\tilde{Y}_{i-j,l}}{S_{j,m,n}} \\ &= \frac{1}{S_{0,m,n}} \sum_{j=0}^{D} \tilde{a}_{D,j} \frac{\tilde{Y}_{i-j,l} \times S_{0,m,n}}{S_{j,m,n}}. \end{aligned} \tag{41}$$

Note that $\left|\tilde{z}_{i,m,n,l}\right|^2$ is independent of $S_{0,m,n}$ (due to isometry [11]) and is dependent only on the phase changes between $S_{j,m,n}$ and $S_{0,m,n}$. In particular, the all-zero and all-one-zero code sequence d_k (this corresponds to the all-zero state sequence and the all-one state sequence respectively in Fig. 3), yield the same magnitude squared error $\left|\tilde{z}_{i,m,n,l}\right|^2$, and are hence indistinguishable.

In general it is clear from (41) that two encoded symbol sequences

$$\mathbf{S}^{(v)} = \left\{ \ldots S_{k-1}^{(v)} S_k^{(v)} S_{k+1}^{(v)} \ldots \right\} \tag{42}$$

and

$$\mathbf{S}^{(\omega)} = \left\{ \ldots S_{k-1}^{(\omega)} S_k^{(\omega)} S_{k+1}^{(\omega)} \ldots \right\} \tag{43}$$

are isometric if

$$\mathbf{S}^{(\omega)} = e^{\mathrm{j}\phi} \mathbf{S}^{(v)} \tag{44}$$

where ϕ is a constant phase.

This implies that we need to differentially encode d_k at the transmitter. However, when differential encoding is done then $\mathscr{M}(\mathscr{S}_k) \neq S_k$ (see Fig. 4 and Sect. 4.3).

Now consider (41). Note that $S_{0,m,n}/S_{1,m,n}$ is a function of the input code digit $\mathscr{S}_{0,j}$ in Fig. 4 (see also Table 1). Mathematically, this can be stated as

$$\frac{S_{0,m,n}}{S_{1,m,n}} = f_1\left(\mathscr{S}_{0,j}\right). \tag{45}$$

Similarly in (41)

$$\frac{S_{0,m,n}}{S_{2,m,n}} = f_2\left(\mathscr{S}_{0,j}, \mathscr{S}_{1,j}\right) \tag{46}$$

where $f_2\left(\mathscr{S}_{0,j}, \mathscr{S}_{1,j}\right)$ is some function of $\mathscr{S}_{0,j}$ and $\mathscr{S}_{1,j}$ in Fig. 4, depending on the differential encoding rules in Table 1. Continuing in this manner we find that

$$\frac{S_{0,m,n}}{S_{P,m,n}} = f_P\left(\mathscr{S}_{0,j}, \mathscr{S}_{1,j}, \ldots, \mathscr{S}_{P-1,j}\right). \tag{47}$$

Thus, we find from (47) that the metric in (41) is a function of only $P-1$ code digits in the memory, with $\mathscr{S}_{0,j}$ being the input code digit. Thus the number of supertrellis states with differential encoding is only $S_{\mathrm{E}} \times 2^{P-1}$ instead of $S_{\mathrm{E}} \times 2^{P}$.

4.4.3 Backward Recursion

Assuming that the trellis is not terminated using tail bits, we first initialize $\beta_{L_d,n}$ as $\beta_{L_d,n} = 1$ for $(0 \le n \le S_{\mathrm{ST}} - 1)$ and then compute $\beta_{i,n}$ $(1 \le i \le L_d - 1)$ recursively as:

$$\begin{aligned} \beta'_{i,n} &= \sum_{m \in \mathscr{D}_n} \beta_{i+1,m} \gamma_{i,n,m} P\left(c_{i,n,m}\right) \\ \beta_{i,n} &= \frac{\beta'_{i,n}}{\sum_{n=0}^{S_{\mathrm{ST}}-1} \beta'_{i,n}}. \end{aligned} \tag{48}$$

The normalization step in the last equation of (48) is required for numerical stability.

Let $\rho^+(n)$ denote the next supertrellis state when the present supertrellis state is n and the input symbol c_k at time instant k is $+1$. Similarly, $\rho^-(n)$ denotes the next supertrellis state when the present supertrellis state is n and the input symbol c_k is -1.

Now, for $(0 \le k \le L_d - 1)$ define:

$$\begin{aligned} C_{k+} &= \sum_{n=0}^{S_{\mathrm{ST}}-1} \alpha_{k,n} \gamma_{k,n,\rho^+(n)} \beta_{k+1,\rho^+(n)} \\ C_{k-} &= \sum_{n=0}^{S_{\mathrm{ST}}-1} \alpha_{k,n} \gamma_{k,n,\rho^-(n)} \beta_{k+1,\rho^-(n)}. \end{aligned} \tag{49}$$

Finally, the extrinsic information that is fed to the outer code via the deinterleaver is given as:

$$\mathcal{E}(c_k = +1) = \frac{C_{k+}}{C_{k+} + C_{k-}}, \; \mathcal{E}(c_k = -1) = \frac{C_{k-}}{C_{k+} + C_{k-}}. \tag{50}$$

4.5 BCJR for the Outer Code [5]

The BCJR for the outer code operates on the encoder trellis with S_{E} states. Let $\alpha_{i,n}$ $(0 \le i \le L_d/2 - 1)$ and $\beta_{i,n}$ $(1 \le i \le L_d/2)$ denote the forward and backward SOP at time instant i and encoder state n $(0 \le n \le S_{\mathrm{E}} - 1)$. Let $\mathcal{D}_{E,n}$ denote the encoder states that diverge from encoder state n. Let $\mathcal{C}_{E,n}$ denote the set of encoder states that converge to encoder state n.

4.5.1 Forward Recursion

Assuming that the receiver knows the starting state s of the encoder, we first initialize $\alpha_{0,n}$ as:

$$\alpha_{0,n} = \begin{cases} 1, & \text{if} \quad n = s, \\ 0, & \text{if} \quad n \neq s \end{cases} \tag{51}$$

and then calculate $\alpha_{i+1,n}$ $(0 \leq i \leq L_d/2 - 2)$ through recursions as:

$$\begin{aligned} \alpha'_{i+1,n} &= \sum_{m \in \mathcal{C}_{E,n}} \alpha_{i,m} \gamma_{sys,i,m,n} \gamma_{par,i,m,n} P\left(g_{i,m,n}\right) \\ \alpha_{i+1,n} &= \frac{\alpha'_{i+1,n}}{\sum_{n=0}^{S_E - 1} \alpha'_{i+1,n}} \end{aligned} \tag{52}$$

where $P\left(g_{i,m,n}\right)$ denotes the a priori probability of the systematic bit corresponding to the transition from encoder state m to n at time instant i. In the absence of any other information, this is fixed at 0.5 [4]. Now

$$\begin{aligned} \gamma_{sys,i,m,n} &= \begin{cases} \mathcal{E}\left(c_{\pi(2i)} = +1\right), & \text{if} \quad \text{condition } \mathcal{H}_1 \\ \mathcal{E}\left(c_{\pi(2i)} = -1\right), & \text{if} \quad \text{condition } \mathcal{H}_2 \end{cases} \\ \gamma_{par,i,m,n} &= \begin{cases} \mathcal{E}\left(c_{\pi(2i+1)} = +1\right), & \text{if} \quad \text{condition } \mathcal{H}_3 \\ \mathcal{E}\left(c_{\pi(2i+1)} = -1\right), & \text{if} \quad \text{condition } \mathcal{H}_4 \end{cases} \end{aligned} \tag{53}$$

where $\mathcal{E}(c_k)$ is given in (50) and

$$\begin{aligned} &\mathcal{H}_1 : \text{systematic bit from state } m \text{ to } n \text{ is } +1 \\ &\mathcal{H}_2 : \text{systematic bit from state } m \text{ to } n \text{ is } -1 \\ &\mathcal{H}_3 : \text{parity bit from state } m \text{ to } n \text{ is } +1 \\ &\mathcal{H}_4 : \text{parity bit from state } m \text{ to } n \text{ is } -1 \end{aligned} \tag{54}$$

and $\pi(\cdot)$ denotes the interleaver map.

4.5.2 Backward Recursion

We first initialize $\beta_{L_d/2,n}$ as $\beta_{L_d/2,n} = 1$ for $(0 \leq n \leq S_E - 1)$ and then compute $\beta_{i,n}$ $(1 \leq i \leq L_d/2 - 1)$ as:

$$\begin{aligned} \beta'_{i,n} &= \sum_{m \in \mathcal{D}_{E,n}} \beta_{i+1,m} \gamma_{sys,i,n,m} \gamma_{par,i,n,m} P\left(g_{i,n,m}\right) \\ \beta_{i,n} &= \frac{\beta'_{i,n}}{\sum_{n=0}^{S_E - 1} \beta'_{i,n}}. \end{aligned} \tag{55}$$

Let $\rho^+(n)$ denote the next encoder state when the present encoder state is n and the input symbol g_k is $+1$. Similarly, $\rho^-(n)$ denotes the next encoder state when the present encoder state is n and the input symbol g_k is -1.

Now, for $(0 \leq k \leq L_d/2 - 1)$ define:

$$\begin{aligned} B_{2k+} &= \sum_{n=0}^{S_E-1} \alpha_{k,n}\gamma_{par,k,n,\rho^+(n)}\beta_{k+1,\rho^+(n)} \\ B_{2k-} &= \sum_{n=0}^{S_E-1} \alpha_{k,n}\gamma_{par,k,n,\rho^-(n)}\beta_{k+1,\rho^-(n)}. \end{aligned} \tag{56}$$

Let $\mu^+(n)$ denote the next encoder state when the present encoder state is n and the output parity bit is $+1$. Similarly, $\mu^-(n)$ denote the next encoder state when the present encoder state is n and the output parity bit is -1.

For $(0 \le k \le L_d/2 - 1)$ define:

$$\begin{aligned} B_{2k+1+} &= \sum_{n=0}^{S_E-1} \alpha_{k,n}\gamma_{sys,k,n,\mu^+(n)}\beta_{k+1,\mu^+(n)} \\ B_{2k+1-} &= \sum_{n=0}^{S_E-1} \alpha_{k,n}\gamma_{sys,k,n,\mu^-(n)}\beta_{k+1,\mu^-(n)}. \end{aligned} \tag{57}$$

Finally, the extrinsic information that is fed to the inner code via the interleaver is given as (for $0 \le k \le L_d - 1$):

$$\mathcal{E}(b_k = +1) = \frac{B_{k+}}{B_{k+} + B_{k-}},\ \mathcal{E}(b_k = -1) = \frac{B_{k-}}{B_{k+} + B_{k-}}. \tag{58}$$

4.5.3 The Final Hard Decision

The final hard decision is taken on the a posteriori probability $P\left(g_k|\tilde{\mathbf{Y}}_0, \ldots, \tilde{\mathbf{Y}}_{N_r-1}\right)$. The expressions in (56) are modified as (for $0 \le k \le L_d/2 - 1$):

$$\begin{aligned} B'_{2k+} &= \sum_{n=0}^{S_E-1} \alpha_{k,n}\gamma_{sys,k,n,\rho^+(n)}\gamma_{par,k,n,\rho^+(n)}\beta_{k+1,\rho^+(n)} \\ B'_{2k-} &= \sum_{n=0}^{S_E-1} \alpha_{k,n}\gamma_{sys,k,n,\rho^-(n)}\gamma_{par,k,n,\rho^-(n)}\beta_{k+1,\rho^-(n)}. \end{aligned} \tag{59}$$

The expressions for the a posteriori probabilities are given as (for $0 \le k \le L_d/2 - 1$):

$$\begin{aligned} P\left(g_k = +1|\tilde{\mathbf{Y}}_0, \ldots, \tilde{\mathbf{Y}}_{N_r-1}\right) &= \frac{B'_{2k+}}{B'_{2k+} + B'_{2k-}} \\ P\left(g_k = -1|\tilde{\mathbf{Y}}_0, \ldots, \tilde{\mathbf{Y}}_{N_r-1}\right) &= \frac{B'_{2k-}}{B'_{2k+} + B'_{2k-}}. \end{aligned} \tag{60}$$

4.6 Reduced State (RS)-BCJR

The basic idea behind the reduced-state BCJR is to reduce the number of supertrellis states in the inner decoder, in Fig. 2. We present two heuristic methods of reducing the supertrellis states, that are based on per-survivor-processing [3, 5].

The number of supertrellis states in the RS-BCJR is given by:

$$S_{\mathrm{ST}} = S_{\mathrm{E}} \times N^{Q} \quad \text{where} \quad Q < P. \tag{61}$$

Here $\tilde{z}_{i,m,n,l}$ in (39) is replaced by $\tilde{z}^{(j)}_{i,m,n,l}$, where

$$\begin{aligned} \tilde{z}^{(j)}_{i,m,n,l} &= \tilde{X}_{i,l} - \hat{X}^{(j)}_{i,l} \\ &= \begin{cases} \sum_{\tau=0}^{Q} \tilde{a}_{P,\tau} \dfrac{\tilde{Y}_{i-\tau,l}}{S_{\tau,m,n}} + \sum_{\tau=Q+1}^{P} \tilde{a}_{P,\tau} \dfrac{\tilde{Y}_{i-\tau,l}}{S^{(j)}_{\tau-Q,m}} & \text{for} \quad i \geq P, \\ \sum_{\tau=0}^{Q} \tilde{a}_{i,\tau} \dfrac{\tilde{Y}_{i-\tau,l}}{S_{\tau,m,n}} + \sum_{\tau=Q+1}^{i} \tilde{a}_{i,\tau} \dfrac{\tilde{Y}_{i-\tau,l}}{S^{(j)}_{\tau-Q,m}} & \text{for} \quad Q < i < P, \\ \sum_{\tau=0}^{i} \tilde{a}_{i,\tau} \dfrac{\tilde{Y}_{i-\tau,l}}{S_{\tau,m,n}} = \tilde{z}_{i,m,n,l} & \text{for} \quad 0 \leq i \leq Q. \end{cases} \end{aligned} \tag{62}$$

The QPSK symbols $S_{\tau,m,n}$ $(0 \leq \tau \leq Q)$ are extracted from the supertrellis state m and the remaining QPSK symbols are taken from the path history leading to supertrellis state m. Since there is more than one path that leads to supertrellis state m, the superscript j refers to the jth such path. Similarly $\gamma_{i,m,n}$ in (39) must be replaced by $\gamma^{(j)}_{i,m,n}$, where

$$\gamma^{(j)}_{i,m,n} = \exp\left(- \frac{\sum_{l=0}^{N_r-1} \left| \tilde{z}^{(j)}_{i,m,n,l} \right|^2}{2\hat{\sigma}_D^2} \right). \tag{63}$$

The forward and backward SOP are given by (38) and (48) respectively, where $\gamma_{i,m,n}$ is

$$\gamma_{i,m,n} = \sum_{j} \gamma^{(j)}_{i,m,n} \tag{64}$$

where $\gamma^{(j)}_{i,m,n}$ is given by (63).

There is yet another method of implementing the RS-BCJR. Here the forward and backward SOP are given by (38) and (48) respectively. However, $\gamma_{i,m,n}$ is now given by

$$\gamma_{i,m,n} = \max_{j} \gamma^{(j)}_{i,m,n} \tag{65}$$

where $\gamma^{(j)}_{i,m,n}$ is given by (63).

5 Pilot Based Coherent Receiver

In this section, an FFT-based channel estimation method which requires pilots, and operates in the frequency domain, is presented. The L_p equally spaced pilots are embedded into the OFDM frame as shown in Fig. 5.

The algorithm for performing the channel estimation is as follows:

1. Obtain the least-squares (LS) channel estimates at the pilot locations as:

$$\hat{H}_{\mathrm{LS},k,l} = \frac{\tilde{Y}_{k,l}}{S_k} \quad 0 \leq k \leq L_p - 1, \ 0 \leq l \leq N_r - 1. \tag{66}$$

The time index k in (66) is obtained after decimation, in the frequency domain.

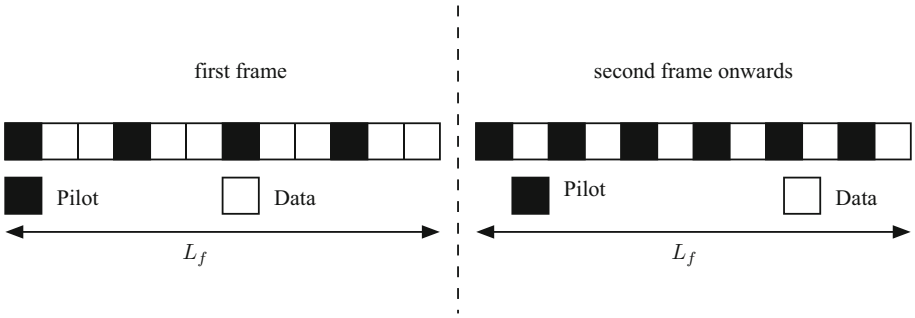


Fig. 5 OFDM frames in the frequency domain for the pilot-based approach

2. Obtain the L_p-pt. IFFT of $\hat{H}_{\mathrm{LS},k,l}$ to get the channel impulse response as:

$$\hat{h}_{\mathrm{LS},i,l} = \frac{1}{L_p} \sum_{k=0}^{L_p-1} \hat{H}_{\mathrm{LS},k,l} e^{\mathrm{j}2\pi ik/L_p} \quad 0 \le i \le L_p - 1,\ 0 \le l \le N_r - 1. \tag{67}$$

3. Since the length of the channel impulse response is L_h where $L_h \ll L_p$, the samples $\hat{h}_{\mathrm{LS},i,l}$ $(L_h \le i \le L_p - 1)$ in (67) are supposed to be zero. However, in the presence of the noise, these samples are not zero. Hence they are set to zero as follows:

$$\hat{h}'_{\mathrm{LS},i,l} = \begin{cases} \hat{h}_{\mathrm{LS},i,l}, & \text{if} \quad 0 \le i \le L_h - 1 \\ 0, & \text{if} \quad L_h \le i \le L_f - 1. \end{cases} \tag{68}$$

The channel frequency response is then obtained as:

$$\hat{H}'_{\mathrm{LS},k,l} = \sum_{i=0}^{L_f-1} \hat{h}'_{\mathrm{LS},i,l} e^{-\mathrm{j}2\pi ik/L_f} \quad 0 \le k \le L_f - 1, 0 \le l \le N_r - 1. \tag{69}$$

The estimate of the channel in (69) is used in (5), (6), (7) and (8) of [8] to perform coherent detection.

Table 3 Simulation parameters

Parameter	Value
Frame size L_f	1024
Channel memory $L_h - 1$	9
Length of the cylic prefix L_{CP}	9
No. of turbo iterations	10
No. of frames simulated	2×10^5
Transmitter antennas	1
Receiver antennas N_r	2
1D channel fade variance σ_f^2	0.5
Generator matrix for the inner and outer encoder	$\left[1 \quad \frac{1+D^2}{1+D+D^2}\right]$

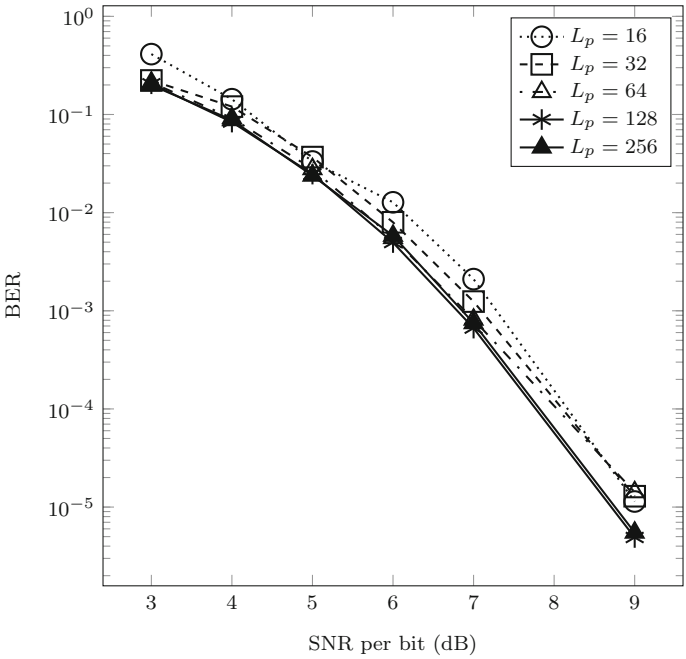


Fig. 6 BER performance of linear prediction-based receiver for serially concatenated turbo coded OFDM, $P = 3$ for different L_p

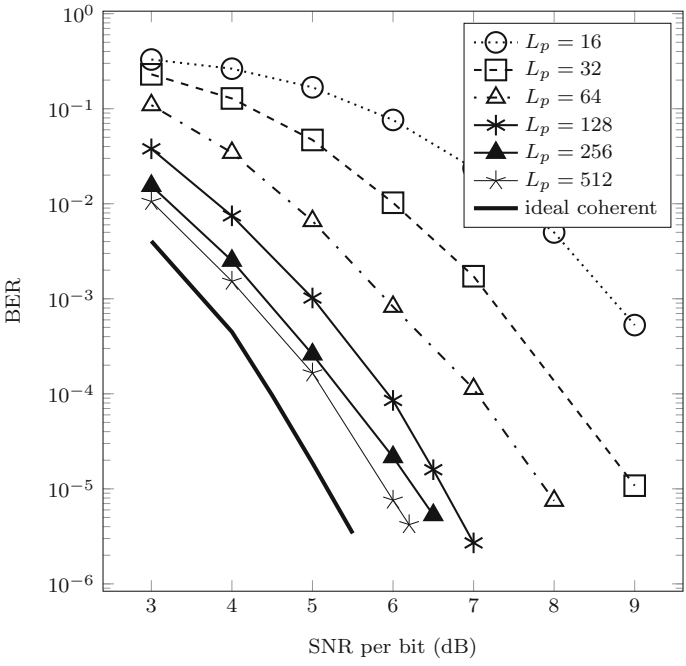


Fig. 7 BER performance of coherent detection using pilot-based channel estimation for serially concatenated turbo coded OFDM, for different L_p

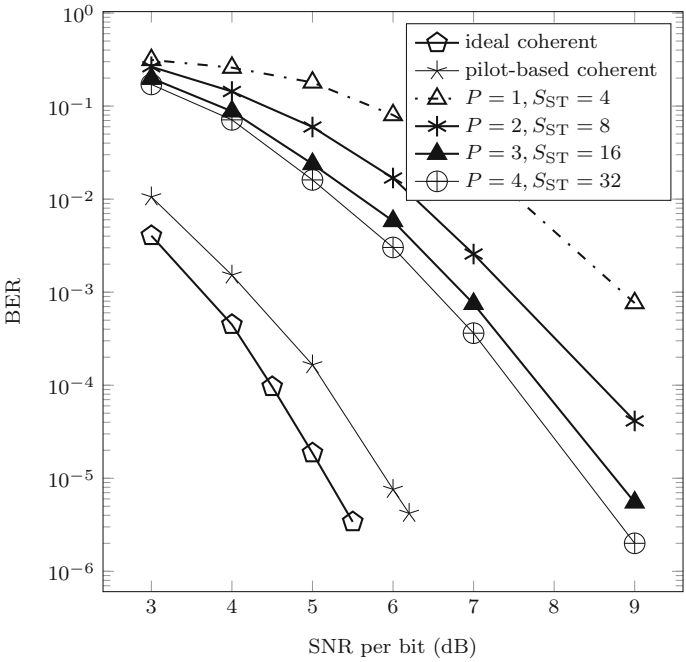


Fig. 8 BER performance of linear prediction-based receiver for serially concatenated turbo coded OFDM, $L_p = 256$

6 Simulation Results

Since the overall rate is 1/2, the average SNR per bit for each receive diversity arm is defined as [8]:

$$\begin{aligned} \text{SNRperbit} &= \frac{2N_r \times E\left[\left|\tilde{H}_{k,l}S_k\right|^2\right]}{E\left[\left|\tilde{W}_{k,l}\right|^2\right]} \\ &= \frac{2N_r \times \left(L_h \times 2\sigma_f^2\right) \times |S_k|^2}{L_f \times 2\sigma_w^2}. \end{aligned} \tag{70}$$

The throughput for the rate-1/2 system in Fig. 2 is calculated as

$$\text{Throughput} = \frac{L_d/2}{L_d + L_p + L_{\text{CP}}} \times 100. \tag{71}$$

Note that for the throughput formula given in (71), $L_p = 0$ for the proposed linear prediction-based approach, since from the second frame onwards there is no training. The simulation parameters are given in Table 3.

Simulation results for the proposed approach are compared against the ideal coherent detector where perfect channel-state information (CSI) is assumed [6] and a practical coherent detector where the channel frequency response is estimated using pilots, as discussed in Sect. 5. Note that in [6] a rate-1 / 2 SISO-OFDM is considered, in [8] a rate-1 SIMO-OFDM system is simulated, whereas in this paper, a rate-1 / 2 SIMO-OFDM

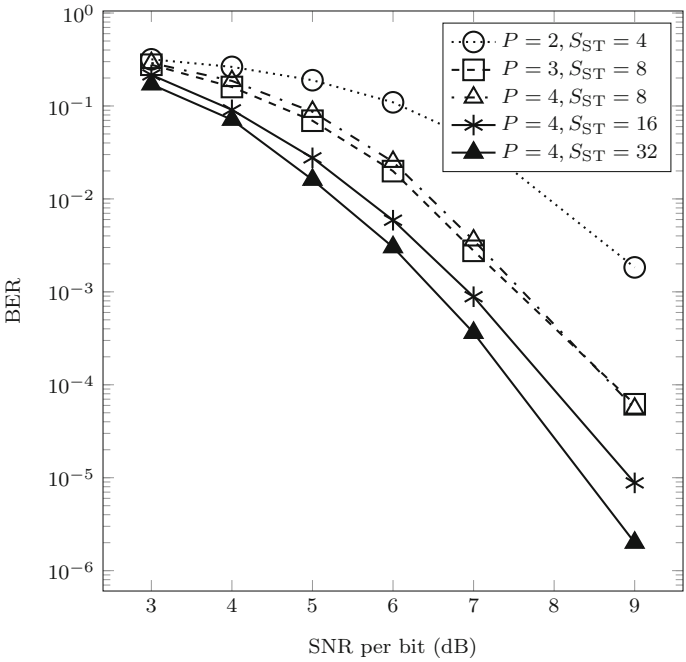


Fig. 9 BER performance of linear prediction-based RS-BCJR receiver for serially concatenated turbo coded OFDM, $L_p = 256$, using (64)

system is analyzed. Moreover in [6] and [8], a parallel concatenated turbo code is considered, whereas in this paper a serial concatenated turbo code is used, since it offers better BER performance for the linear prediction-based receiver. In Fig. 6, we compare the performance of the proposed receiver for different training lengths. Note that for $L_p = 128$ and $L_p = 256$ gives the best performance. There is a 0.5 dB difference in performance between $L_p = 64$ and $L_p = 128$. This clearly shows that increasing the training length L_p above 128, does not result in improved BER performance. In Fig. 7, we present the results for the practical coherent receiver with estimated channel frequency response as explained in Sect. 5, for different L_p. We find that $L_p = 512$ gives the best performance. Again, increasing L_p above 512 does not result in significant improvement in BER performance. It is observed that the practical coherent receiver is just 0.5 dB inferior to the ideal coherent receiver. From Figs. 8 and 11, we find that the performance of the ideal coherent detector for both serial and parallel concatenated code is similar. However, the performance of the proposed prediction filter-based detector with $P = 3$ is much better for serial concatenation as compared to parallel concatenation [6] which was mentioned earlier. In Figs. 9 and 10, we give the performance of the predictive RS-BCJR. From Figs. 7 and 8, we observe that the detection with channel estimation is only 2.2 dB superior in terms of BER. The complexities and the throughputs for the two approaches is given in Table 4.

Simulation results show that with a diversity order of two, the proposed receiver achieves a bit-error-rate (BER) of 10^{-5} for an signal-to-noise ratio (SNR) per bit of 8.2 dB with 32 states in the inner encoder trellis and 4 states in the outer encoder trellis. The reduced-state BCJR algorithm achieves a BER of 10^{-5} for an SNR per bit of 9 dB with just

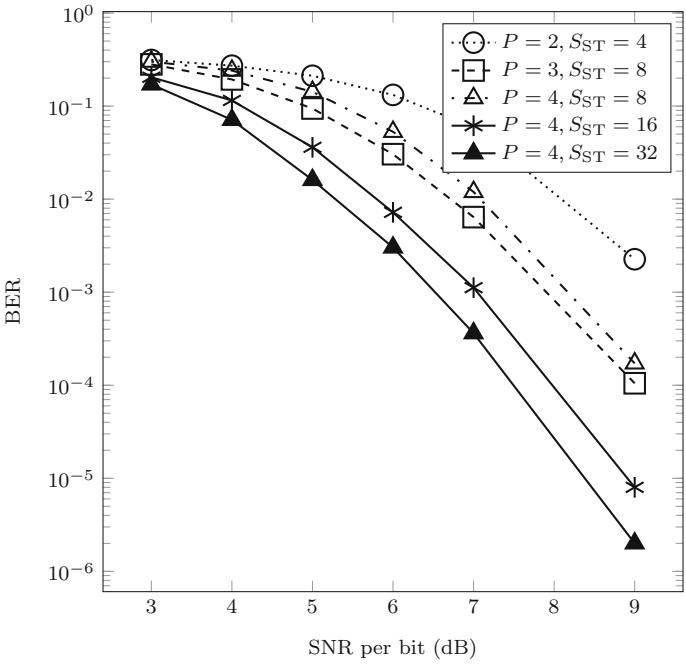


Fig. 10 BER performance of linear prediction-based RS-BCJR receiver for serially concatenated turbo coded OFDM, $L_p = 256$, using (65)

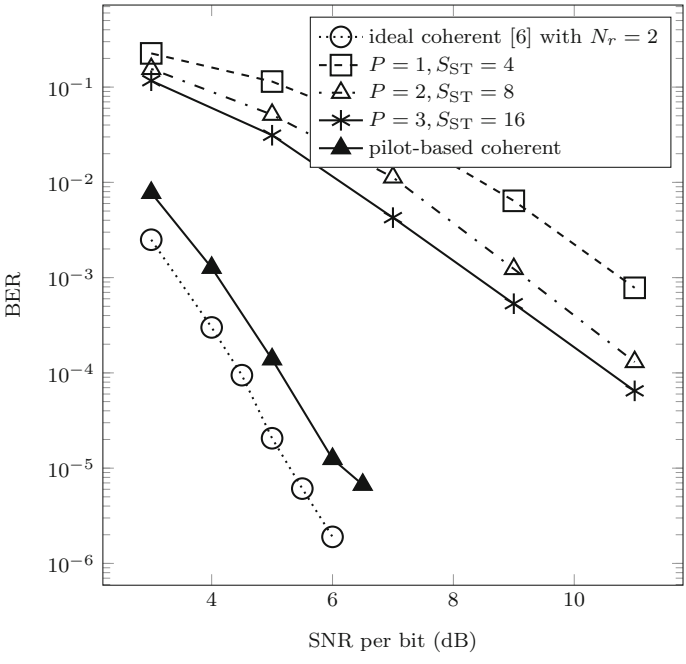


Fig. 11 BER performance of linear prediction-based receiver for parallelly concatenated turbo coded OFDM [6], $L_p = 256$. Note that for practical coherent, $L_d/2 = 384$

Table 4 Comparision of methods

Method	Throughput (%)	Complexity	
		Inner encoder trellis	Outer encoder trellis
Pilot-based ($L_p = 512$)	24.7	4 states	4 states
Linear prediction-based ($P = 4$)	49.5	32 states	4 states
Reduced-state BCJR-based ($P = 4$)	49.5	16 states	4 states

16 states in the inner encoder trellis. Though the FFT-based channel estimation approach achieves a BER of 10^{-5} at an SNR per bit of just 6 dB, its throughput is only 24.7% compared to the proposed method whose throughput is 49.5% for the rate-1/2 serial concatenated turbo code.

7 Conclusion

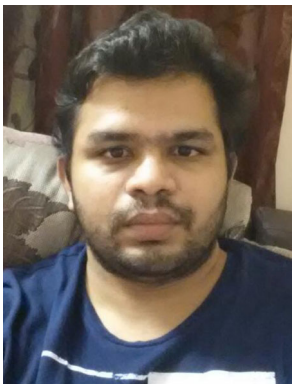
In this paper, an approach for the linear prediction-based detection of serially concatenated turbo coded signals in SIMO-OFDM, to achieve throughput close to 50%, is proposed. Simulation results show that a BER of 10^{-5} for an SNR per bit of 8.2 dB is possible.

Future work can be focussed on increasing the overall rate to unity and also exploiting low-density parity-check codes.

References

1. Barbieri, A. (2010). Blind per-state detection of DPSK over correlated fading channels. *IEEE Transactions on Vehicular Technology*, *59*(5), 2320–2327.
2. Proakis, J. (2008). *Digital communications*. Boston: McGraw-Hill.
3. Raheli, R., Polydoros, A., & Tzou, C. K. (1995). Per-survivor processing: A general approach to mlse in uncertain environments. *IEEE Transactions on Communications*, *43*(2/3/4), 354–364. doi:10.1109/26.380054.
4. Tuchler, M., & Singer, A. C. (2011). Turbo equalization: An overview. *IEEE Transactions on Information Theory*, *57*(2), 920–952.
5. Vasudevan, K. (2007). Turbo equalization of serially concatenated turbo codes using a predictive-DFE based receiver. *Signal, Image and Video Processing, Springer*, *1*(3), 239–252.
6. Vasudevan, K. (2013). Coherent detection of turbo coded OFDM signals transmitted through frequency selective Rayleigh fading channels. In *2013 IEEE international conference on signal processing, computing and control (ISPCC), Shimla* (pp. 1–6).
7. Vasudevan, K. (2015a) Coherent detection of turbo-coded OFDM signals transmitted through frequency selective Rayleigh fading channels with receiver diversity and increased throughput. CoRR abs/1511.00776. arXiv:1511.00776.
8. Vasudevan, K. (2015b). Coherent detection of turbo-coded OFDM signals transmitted through frequency selective Rayleigh fading channels with receiver diversity and increased throughput. *Wireless Personal Communications*, *82*(3), 1623–1642.
9. Vasudevan, K. (2016). Coherent turbo coded MIMO OFDM. In *The twelfth international conference on wireless and mobile communications, Barcelona, ICWMC 2016* (pp. 91–99).
10. Vasudevan, K. (2017). *Digital Communications and signal processing* (3rd ed.). http://home.iitk.ac.in/~vasu/book0.pdf.

11. Vasudevan, K., Giridhar, K., & Ramamurthi, B. (2001). Efficient suboptimum detectors based on linear prediction in Rayleigh flat-fading channels. *Signal Processing Journal*, *81*(4), 819–828.
12. Ylioinas, J., & Juntti, M. (2009). Iterative joint detection, decoding, and channel estimation in turbo-coded MIMO-OFDM. *IEEE Transactions on Vehicular Technology*, *58*(4), 1784–1796.



Vineel K. Veludandi completed his Bachelor of Engineering from the Department of Electronics and Communication Engineering, University College of Engineering, Osmania University, India and his Master of Technology from the Department of Electrical Engineering, Indian Institute of Technology, Guwahati. Currently he is pursuing his Ph.D. at the Department of Electrical Engineering, Indian Institute of Technology, Kanpur. His research areas includes channel estimation and turbo equalization.

K. Vasudevan completed his Bachelor of Technology (Honours) from the Department of Electronics and Electrical Communication Engineering, IIT Kharagpur, India, in the year 1991, and his MS and Ph.D. from the Department of Electrical Engineering, IIT Madras, in the years 1996 and 2000 respectively. During 1991–1992, he was employed with Indian Telephone Industries Ltd, Bangalore, India. He was a Post Doctoral Fellow at the Mobile Communications Lab, EPFL, Switzerland, between Dec. 1999 to Dec. 2000, and an engineer at Texas Instruments, Bangalore, between Jan 2001 and June 2001. Since July 2001, he has been a faculty at the Electrical department at IIT Kanpur, where he is now an Associate Professor. His interests lie in the area of communications and signal processing.