
© 2004 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Reduced-Complexity SISO Equalization for Rayleigh Fading Channels with Known Statistics

Xiao-Ming Chen and Peter A. Hoeher
Information and Coding Theory Lab, University of Kiel
Kaiserstr. 2, D-24143 Kiel, Germany, E-mail: {xc,ph}@tt.uni-kiel.de

Abstract—For Rayleigh fading channels with known covariance matrix, reduced-complexity receivers can be derived which approximate the maximum likelihood (ML) receiver by defining additional constraints. The reduced-complexity receiver, which assumes that the current channel output depends only on previous channel outputs, is termed forced-folding sequence estimator (FSE). We propose another approach, where a long burst is divided into overlapped subbursts and the computations are performed over these short subbursts. This receiver is referred to as overlapped subburst sequence estimator (OSE). We compare the FSE and the OSE and derive their soft-input soft-output counterparts, which are termed as SISO-FSE and SISO-OSE.

From the viewpoint of linear estimation, the minimum mean-square error (MMSE) filtering/smoothing approaches are verified to be unsuitable for slowly fading channels in the context of trellis-based equalization. For symmetrical signal constellations, differential encoding is necessary to resolve the phase ambiguity. We show how to exploit the symmetry during the evaluation of branch/path metrics to obtain a reduced-complexity structure without loss of performance.

1. SYSTEM MODEL AND MAXIMUM-LIKELIHOOD RECEIVER

We consider a burst-wise transmission with $K$ linearly modulated data symbols pro burst. Throughout this paper we use the complex baseband notation. The output signal of the receive filter can be written as

$$y(t) = \sum_{k=0}^{K-1} z[k]h(t; t-kT) + n(t),$$

(1)

where $z[k]$, $h(t; \tau)$, and $n(t)$ denote the data symbols, the time-varying overall channel impulse response, and the additive Gaussian noise process, respectively. After receive filtering, $y(t)$ is sampled at a suitable sampling rate to provide a sufficient statistics for the detection of $\{z[k]\}$. For convenience, we select the sampling period $T_s = T/J$, where $J$ is the number of samples per symbol. The discrete-time system model is therefore represented as

$$y[n] = y(nT_s) = \sum_{l=0}^{L_s} d[n-l]h[n] + n[n] = d_T[n]h[n] + n[n],$$

(2)

where we define a new data sequence

$$d[n] = \begin{cases} z[k] & \text{if } \kappa = JK, \\ 0 & \text{otherwise}, \end{cases} \quad 0 \leq \kappa \leq KJ - 1,$$

and $\{h[n]\}$ denote the equivalent channel coefficients.

In vector/matrix notation, (2) can be rewritten as

$$y = Dh + n,$$

(3)

where $y = [y[0], \ldots, y[KJ - 1]]^T$ and

$$D = \text{diag}(d_T[0], \ldots, d_T[KJ - 1]) \in \mathbb{C}^{KJ \times KJ (L_s+1)},$$

$$h = [h_T[0], \ldots, h_T[KJ - 1]^T,$$

$$n = [n[0], \ldots, n[KJ - 1]^T.$$}

Assuming that there are $N_e$ resolvable paths, we can write the channel coefficients as

$$h_l[n] = \sum_{m=0}^{N_e-1} h_m(n)g(lT_s - \tau_m), \quad 0 \leq l \leq L_s,$$

(4)

where $h_m(n)$ characterizes the attenuation of the individual resolvable paths, $\tau_m$ is the propagation delay w.r.t. line-of-sight path, and $g(t)$ denotes the convolution of the pulse shaping filter and the receiving filter. Assuming 2-D isotropic scattering, the autocorrelation function of the channel coefficients can be evaluated as

$$E\{h_l[n]h_l^*[n-k]\} = \sum_{m=0}^{N_e-1} g_m[l_1]g_m^*[l_2]J_0(2\pi f_D T_s \tau_1),$$

(5)

where $J_0(\cdot)$ stands for the zeroth order Bessel function of first kind, and $f_D T_s$ denotes the normalized fading rate. The components in the summation are determined as

$$g_m[l] = \sqrt{P_m g(lT_s - \tau_m)}, \quad 0 \leq m \leq N_e - 1, 0 \leq l \leq L_s,$$

(6)

where $P_m = E\{|h_m[n]|^2\}$ is the average power of the $m$th resolvable fading path. Therefore, $\{P_m, \tau_m\}$ determine the delay power profile of the underlying fading channel, which is assumed to be known at the receiver.

Given the channel model (3), maximum-likelihood sequence estimation can be formulated as follows [1]:

$$\hat{x} = \arg\min_x \left\{ -\log p\left(y | \hat{D}, R_h\right) \right\}$$

$$= \arg\min_x \left\{ \log (\det R_y) + y^H R_y^{-1} y \right\},$$

(7)

where only the covariance matrix of random channel coefficients $R_h = E\{hh^H\}$ is available at the receiver. The conditional covariance matrix of channel outputs is denoted as

$$\hat{R}_y = E\{yy^H | \hat{D}, R_h\} = \hat{D}R_h\hat{D}^H + N_0 I_{KJ},$$

(8)
where a square-root Nyquist filter is applied at the receiver to obtain uncorrelated noise samples, and \( N_0 \) stands for the one-sided power spectral density of additive white Gaussian noise process before the receiving filter.

II. SOFT-INPUT SOFT-OUTPUT (SISO) MODULES DERIVED FROM THE ML RECEIVER

The ML solution (7) is actually an exhaustive approach because \( \tilde{R}_y \) depends on the whole data sequence hypothesis \( D \). To avoid a complexity exponentially increasing with the time index, additional constraints have to be employed on the likelihood function. Consequently, reduced-complexity approaches can be obtained, which makes a recursive, efficient evaluation of the argument in (7) possible.

A. SISO Forced-folding Sequence Estimator

Under the approximation that the current channel output vector statistically depends on \( M\bar{J} \) previous channel outputs, we get

\[
p(y[k] | \tilde{D}, R_h, y_0^{k-1}) \approx p(y[k] | \tilde{D}, R_h, y_1[k]),
\]

where \( y[k] = [y[k], \ldots, y[k+J-1]]^T \) and \( y_1[k] = [y[k], \ldots, y[k-M]]^T \). The approximation (9) was referred to as forced-folding condition in [2], where the dependence on the whole path history is folded to a finite memory length independent of time index \( k \). Accordingly, \( y[k] \) statistically merely depends on \( M + L + 1 \) data symbols \( z[k-M-L], \ldots, z[k] \), where \( L = (L_0 + 1)/J \). Therefore, the equalizer based on the approximation (9) is referred to as forced-folding sequence estimator (FSE) in the sequel. An evaluation of (7) can be carried out on a trellis with an equivalent memory length \( L_F = M + L \):

\[
\hat{x} = \arg \min \frac{1}{2L_F} \sum_{k=0}^{K-1} \log \left( \frac{p(y[k] | \tilde{x}[k], R_h, y_1[k])}{p(y[k] | \tilde{x}[k], R_h)} \right) = \arg \min \frac{1}{2L_F} \sum_{k=0}^{K-1} \log \left( \frac{p(y[k], y_1[k] | \tilde{x}[k], R_h)}{p(y[k], y_1[k] | \tilde{x}[k])} \right)
\]

(10)

The branch metrics \( \lambda_F(\tilde{x}[k]) \) depend on state transitions \( \tilde{x}[k] = [\tilde{x}[k] - L_F], \ldots, \tilde{x}[k] \), which is determined by the current state \( \tilde{s}[k] = [\tilde{x}[k] - L_F + 1], \ldots, \tilde{x}[k] \) and its predecessor \( \tilde{s}[k-1] \). Apply the determinant and inverse rules for a partitioned matrix in (10), cf. (8), the branch metrics can be evaluated as

\[
\lambda_F(\tilde{x}[k]) = \left\| \tilde{W}_y[k] y[k] - \tilde{W}_r[k] \tilde{R}_{y_1[k]}^{-1}[y[k]] y_1[k] \right\|^2 + \log \left( \det \tilde{W}_r^{-1}[k] \right)
\]

(12)

\[
= \left( y[k] - \tilde{R}_{y_1[k]}^{-1}[y[k]] \tilde{R}_{y_1[k]}^{-1}[y[k]] y_1[k] \right)^H \tilde{W}_y[k] \times \left( y[k] - \tilde{R}_{y_1[k]}^{-1}[y[k]] \tilde{R}_{y_1[k]}^{-1}[y[k]] y_1[k] \right) + \log \left( \det \tilde{W}_r^{-1}[k] \right)
\]

(13)

where \( \tilde{W}_y[k] \) and \( \tilde{W}_r[k] \) can be interpreted as a whitening filter and

\[
\tilde{W}_y[k] = \left( \tilde{R}_y[k] - \tilde{R}_{x_1[k]} \tilde{R}_{x_1[k]}^{-1}[y[k]] \right)^{-1},
\]

\[
\tilde{R}_{y_1[k]} = E \left\{ y[k] y_1[k] | \tilde{x}[k], R_h \right\},
\]

\[
\tilde{R}_{x_1[k]} = E \left\{ y[k], y_1[k] | \tilde{x}[k], R_h \right\}.
\]

Note that \( \tilde{R}_{x_1[k]}^{-1}[y[k]] \) is the MMSE linear prediction matrix for \( y[k] \) based on \( y_1[k] \), which depends on the data hypotheses. Accordingly, the inverse of \( \tilde{W}_y[k] \) can be regarded as the prediction error variance matrix. Similar results were derived from the MMSE linear prediction [3] with slightly different branch metrics. Moreover, \( \tilde{W}_y[k], \tilde{R}_y[k] \), and \( \tilde{R}_{y_1[k]} \) are only dependent on \( \tilde{x}[k] \), given the channel covariance matrix and the noise variance. Therefore, those terms relevant for the evaluation of branch metrics can be pre-calculated and stored in a look-up table.

In the Appendix, the SISO counterpart of FSE is derived based on the symbol-by-symbol maximum a posteriori probability (MAP) criterion, where the following folding condition is essential:

\[
p(|s[k], y[k], y_1[k], y_0^{k-1}|, R_h) = \exp (-\lambda_F(\tilde{x}[k])^2)
\]

(14)

Therefore, the SISO-FSE works in a similar manner as conventional SISO equalization [4] with appropriately defined branch metrics.

B. SISO Overlapped Subburst Sequence Estimator

Another reduced-complexity receiver technique is to divide a long burst into overlapping subbursts, where the computations are performed over these subbursts. Therefore, this kind of receiver is termed overlapped subburst sequence estimator (OSE). A simplified OSE has been proposed in [5].

Let us define \( K_s \) as the subburst length and \( M_s \) as the overlapped baud-rate symbols. Fig. 1 illustrates how to divide a long burst into overlapping subbursts.

![Fig. 1. Overlapping subbursts with parameters K_s and M_s](image)

The relevant assumption for OSE is statistical independence of adjacent subbursts, which results in

\[
\hat{x} = \arg \max \frac{1}{2L_F} \sum_{k=0}^{K-1} \log \left( \frac{1}{K_s} \prod_{k=0}^{K_s-1} p(y_{K_s}[k] | \tilde{D}_{K_s}[k], R_h) \right)
\]

(15)

where \( \tilde{D}_{K_s}[k] = \text{diag} (d^T[k + 1], \ldots, d^T[k + (k+1)J - 1]), y_{K_s}[k] = [y^T[k + 1 - K_s], \ldots, y^T[k]]^T \),

\[
\tilde{R}_{y_{K_s}[k]} = E \left\{ y_{K_s}[k] y_{K_s}^H[k] | \tilde{D}_{K_s}[k], R_h \right\}.
\]
and the maximal overlapping length has been selected, i.e., $M_s = K_s - 1$. Consequently, the evaluation of (15) can be accomplished by means of a trellis with equivalent memory length $L_O = M_s + L_t$ where the branch metrics are

$$\lambda_O(\tilde{x}[k]) = -\log \left( p \left( y_{K_s}[k] \mid \tilde{D}_{K_s}[k], R_h \right) \right)$$

$$= y_N^T[k] \tilde{R}_{y_{K_s}[k]}^{-1} y_{K_s}[k] + \log \left( \det \tilde{R}_{y_{K_s}[k]} \right), \quad (16)$$

where $\tilde{x}[k] = [\tilde{z}[k - L_O], \ldots, \tilde{z}[k]]^T$. The branch metrics defined in (16) can be reformulated as follows:

$$\lambda_O(\tilde{x}[k]) = \lambda_O(\tilde{R}_{y_{K_s}[k]}), \quad (17)$$

where $y_{K_s}[k]$ denotes the channel output vector with $(K_s - l)^{th}$ samples at the baud-rate time index $k - l$, and $\lambda_O(\tilde{R}_{y_{K_s}[k-l-1]}^{k-l}) = 0$. Comparing (17) with (10), for $M = M_s = K_s - 1$, the branch metric for the FSE is just the addend for $l = 0$ in (10). Therefore, the OSE can be interpreted as an extended FSE, which consists of a set of predictors from the one with order $M_s$ to the one with order 1.

Similar as in FSE, the inverse and determinant of $\tilde{R}_{y_{K_s}[k]}$ can be precalculated for possible state transitions $\tilde{x}[k]$. The SISO-OSE was derived in the Appendix following the assumption:

$$p(\tilde{x}[k]) = p(\tilde{x}[k-1], y_{K_s}[k], \ldots, y_{K_s}[k-1], R_h) \quad (18)$$

$$= p(\tilde{x}[k]) = y_{K_s}[k] \mid \tilde{x}[k-1], R_h = \exp(-\lambda_O(\tilde{x}[k])) P(\tilde{x}[k]),$$

and the approximated a posteriori probabilities can be evaluated as usual.

C. SISO Equalization with the Reduced Equivalent Memory Length Without Performance Loss

In (16), the relevant quantity dependent on data hypotheses is

$$\tilde{R}_{y_{K_s}[k]} = \tilde{D}_{K_s}[k] R_{h_{K_s}[k]} \tilde{D}_{K_s}^H[k] + N_0 I_{K_s}, \quad (19)$$

which implies a phase ambiguity for symmetrical signal constellations like PSK, i.e., $D_{K_s}[k]$ and $e^{j\phi} D_{K_s}[k]$ result in the same conditional covariance matrix. Similarly, FSE can also be shown to have such phase ambiguity, cf. (12). One solution to resolve the phase ambiguity for PSK constellations is the differential encoding, which is applied in this paper. Accordingly, the branch metrics of both SISO receivers can be formulated as

$$\gamma(\tilde{x}[k]) = \gamma(\tilde{x}[k - L_t], \ldots, \tilde{x}[k]) = \lambda(\tilde{x}[k]) - \log(\det(\tilde{R}[\tilde{x}[k]])), \quad (20)$$

where $\tilde{x}[k]$ and $\tilde{a}[k]$ denote the data symbols after and before the differential encoding, i.e., $\tilde{x}[k] = [\tilde{z}[k - 1], \tilde{a}[k]]$. The equivalent memory length of the DPSK/ISI super-trellis [6] is denoted as $L_t$, which is equal to $L_F$ and $L_O$ for SISO-FSE and SISO-OSE, respectively. Fig. 2 illustrates a binary PSK/ISI trellis and the corresponding DPSK/ISI super-trellis for $L_t = 2$, where solid lines and dashed lines correspond to +1 and -1, respectively.

Because of the differential encoding, $\tilde{z}[k - l] = (\prod_{i=k-L_t+1}^{k-1} a[i]) \tilde{z}[k - L_t], 0 \leq l \leq L_t$, where $\tilde{z}[k - L_t]$ serves as the phase reference for subbursts. Therefore, the branch metrics (20) are modified according to

$$\gamma(\tilde{x}[k]) = \gamma(\tilde{a}[k - L_t], \tilde{a}[k - L_t + 1], \ldots, \tilde{a}[k])$$

$$= \gamma(\tilde{a}[k - L_t + 1], \ldots, \tilde{a}[k]),$$

(21)

since $\lambda(\tilde{x}[k])$ is independent of $\tilde{z}[k - L_t]$ for both FSE and OSE, refer (19), (12) and (16). Consequently, the SISO-FSE and the SISO-OSE can equivalently be carried out on the PSK/ISI trellis with memory length $L_t - 1$ instead of on the DPSK/ISI super-trellis with memory length $L_t$ due to

$$\gamma(\tilde{x}[k], \ldots, \tilde{x}[k]) = \gamma(\tilde{a}[k - L_t + 1], \ldots, \tilde{a}[k]).$$

III. TRELLEIS-BASED EQUALIZATION BY MEANS OF LINEAR MMSE ESTIMATION

As stated before, the FSE actually predicts the channel output vector based on previously observed channel output vectors. In this section, we firstly show that the FSE actually performs MMSE channel estimation. Furthermore, the MMSE smoothing/filtering approaches are verified to be unsuitable for slowly fading channels.

A. Inherent MMSE Channel Estimation in the MMSE Channel Output Prediction

The prediction matrix for $y[k]$ based on $y_{1}[k]$ is as follows:

$$\tilde{R}_{y_{1}[k]} = \tilde{D}_{K_s}[k] R_{h_{K_s}[k]} \tilde{D}_{K_s}^H[k] + N_0 I_{K_s}, \quad (22)$$

$$\tilde{R}_{y_{1}[k]}^{-1} = \tilde{D}_{K_s}[k] R_{h_{K_s}[k]} \tilde{D}_{K_s}^H[k] + N_0 I_{K_s}, \quad (22)$$

where $y[k] = \tilde{D}[k] h_{1}[k] + n[k]$ and $y_{1}[k] = \tilde{D}[k] h_{1}[k] + n_{1}[k]$. By applying the matrix inversion lemma in (22), it is easy to show that

$$\tilde{R}_{\tilde{y}_{1}[k]}^{-1} = \tilde{D}_{K_s}[k] R_{h_{K_s}[k]} \tilde{D}_{K_s}^H[k], \quad (23)$$

Notice that MAP channel estimation of $h_{1}[k]$ based on $y_{1}[k]$ and $\tilde{D}[k]$ reads

$$\hat{h}_{1}[k] = \left( N_0 R_{h_{1}[k]}^{-1} + \tilde{D}_{K_s}^H[k] \tilde{D}_{K_s}[k] \right)^{-1} \tilde{D}_{K_s}^H[k] y_{1}[k], \quad (24)$$
where $h[k]$ is assumed to be complex Gaussian random vector. Moreover, (24) coincides with the MMSE channel estimation [7], which reduces to least-squares channel estimation for $N_0R_{h[k]}^{-1}$. Therefore, MMSE channel estimation for $h[k]$ based on $y[k]$ and $\bar{D}_k[k]$ can be obtained as

$$h[k] \mapsto \bar{D}_k[k] = R_{h[k]}h[k]R_{h[k]}^{-1} \bar{D}_k[k]$$

which is an efficient estimation [7]. Consequently, the predicted channel output vector at the time index $k$ can now be written as

$$\bar{y}[k] = \bar{D}_k[k] \bar{h}[k] \mapsto \bar{D}_k[k] \bar{h}[k] y[k] = \bar{D}_k[k] \bar{h}[k] y[\hat{k}]$$.

### B. Unsuitability of MMSE Filtering/Smoothing Under Slowly Fading Channels

Within this section, we assume that the channel coefficients remain unchanged for the considered time interval, i.e., $h[k] = h[k+k']$, $k' \in [-k_1, k_2]$, which is the case for slowly fading channels. Let

$$\bar{D}_k[k] = \left[\bar{D}_k[k-k_1], \ldots, \bar{D}_k[k+k_2]\right]^T \in \mathbb{C}^{(k_1+k_2+1)J \times (L_2+1)J},$$

$$y[k] = D[k]h[k] + n[k] \in \mathbb{C}^{(k_1+k_2+1)J},$$

$$R_h = E\{h[k]h^H[k]\} \in \mathbb{C}^{(L_2+1)J \times (L_2+1)J}.$$  

The linear MMSE estimator for $y[k]$ based on $y[k]$ and $\bar{D}_k[k]$ can be formulated as follows:

$$\hat{y}[k] = F(\bar{D}_k[k]) : y[k] = R_{\bar{D}_k[k]}^{-1} R_{\bar{y}[k]} y[k],$$

where $F(\bar{D}_k[k])$ denotes the MMSE estimation matrix. For $k_1, k_2 \geq 0$ (filtering or smoothing) we have

$$R_{\bar{y}[k]} R_{\bar{D}_k[k]}^{-1} = \bar{D}_k[k] R_h \bar{D}_k[k] + [O_{J \times k_1, J} N_0 I_{J} O_{J \times k_2, J}],$$

$$R_{\bar{D}_k[k]}^{-1} = \left(\bar{D}_k[k] R_h \bar{D}_k[k] + [O_{J \times k_1, J} N_0 I_{J} O_{J \times k_2, J}]^{-1}.$$

By applying the matrix inversion lemma again, we get

$$F(\bar{D}_k[k]) = R_{\bar{y}[k]} R_{\bar{D}_k[k]}^{-1} = [O_{J \times k_1, J} I_{J} O_{J \times k_2, J}],$$

which is independent of data hypotheses and hence proves the unsuitability of the MMSE filtering/smoothing for trellis-based equalization.

### IV. Numerical Results and Discussions

The performance of the proposed SISO-FSE and SISO-OSE was tested over a GSM-like system with burst length $K = 148$. At the transmitter, binary DPSK symbols are passed through a linearized Gaussian shaping filter, while a root-raised-cosine filter is used as receiving filter. The oversampling factor was selected to be $J = 2$. Simulations have been carried out for the GSM05.05 TU channel model, where a (5, 7) convolutional code was applied in conjunction with an inter-burst S-random interleaver (length 1480, $S = 20$). For SISO-FSE and SISO-OSE, $M = 4$ and $M_2 = 4$ were selected for a similar complexity. The Max-LogAPP algorithm [8] was employed both for SISO equalization and SISO channel decoding. The SISO equalization was performed on the PSK/ISI trellis with reduced memory length, as discussed in Section II-C.

![Fig. 3](image-url) **Fig. 3.** BER vs. average $E_b/N_0$ for quasi-static TU channel model. Solid lines correspond to SISO-FSE and dashed lines for SISO-OSE.

Fig. 3 demonstrates simulation results for the quasi-static case, where channel coefficients remain constant within a burst and change independently from burst to burst. After 3 iterations, the gain due to the iterative processing between the SISO equalizer and the SISO channel decoder is only marginal. The gain of the SISO-OSE over the SISO-FSE is around 1dB at a BER of $10^{-4}$.

![Fig. 4](image-url) **Fig. 4.** BER vs. average $E_b/N_0$ for the TU channel model given different normalized fade rates. Solid lines and dashed lines correspond to SISO-FSE and SISO-OSE, respectively.

In Fig. 4, different normalized fade rates were taken into consideration. For clarity, we only showed the simulation results after 3 iterations. For both SISO-OSE and SISO-FSE, the performance for $f_op = 0.0001$ is better than $f_op = 0.0001$. The reason may be that the time variation delivers further diversity compared to the quasi-static case. However, for fast fading channels, $f_op = 0.01$, the SISO-FSE exhibits an error floor at a BER of $2 \cdot 10^{-4}$, which may be that due to the inherent MMSE channel estimator the FSE fails under fast time-varying conditions. In this aspect, the SISO-OSE is much more robust.
than the SISO-FSE, although there is also a performance
degradation for $f_{cT_1} = 0.01$.

V. CONCLUSIONS

In this paper, we gave a unified review of reduced-
complexity sequence estimators for frequency-selective time-
varying Rayleigh fading channels with known statistics.
Two reduced-complexity receivers were derived by means of
putting additional constraints on the ML, receiver, which
were referred to FSE and OSE, respectively. Based on the
symbol-by-symbol MAP principle and exploiting the underlying
constraints, SISO-FSE and SISO-OSE have been derived,
which can be carried out by means of forward- and backward
recursions with suitably defined branch metrics. The SISO-
OSE is more robust than the SISO-FSE under a serve fre-
cquency dispersion and shows a better performance than the
SISO-FSE, both for quasi-static and time-varying channels.
To resolve the phase ambiguity in both SISO-FSE and SISO-
OSE, differential encoding was applied. On the other hand,
this phase ambiguity can be exploited to approach a trellis-
based equalizer with a reduced memory length, while no
performance loss is sacrificed.

APPENDIX: DERIVATION OF SISO-FSE AND SISO-OSE

The symbol-by-symbol (s.b.s.) MAP equalization with
known channel covariance matrix is formulated as follows:

$$
\hat{z}[k] = \arg \max_{\hat{z}[k]} P(\hat{z}[k] \mid y, R_h)
$$

$$
\triangleq \arg \max_{\hat{z}[k]} \left\{ \sum_{\hat{z}[k] \in \mathbb{Z}} p(\hat{z}[k], y \mid R_h) \right\}, \quad (29)
$$

where $\hat{z}[k]: \tilde{z}[k]$ denotes the state transitions consistent with $\hat{z}[k]$. Similar as in the derivation of conventional s.b.s. MAP, we decompose the terms in (29) as

$$
p(\hat{z}[k], y \mid R_h) = p(\hat{z}[k], y_0^{K-1}, y_1^{K-1}, y_2^{K-1}, \ldots, y_i^{K-1} \mid R_h)
$$

$$
\times p(y_0^{K-1}, \ldots, y_i^{K-1} \mid \hat{z}[k], y_i^{K-1}, y_i^{K-1}, \ldots, y_i^{K-1}, y_i^{K-1} \mid R_h)
$$

$$
= \tilde{\alpha}_F(\tilde{z}[k-1]) \tilde{\beta}_F(\tilde{z}[k]), \quad (30)
$$

where the forward-folding condition (14) is applied to get
the final result. The forward- and backward recursions are
evaluated as follows:

$$
\tilde{\alpha}_F(\tilde{z}[k]) = \sum_{\tilde{z}[k-1]} p(\tilde{z}[k], y_0^{K-1}) \mid R_h)
$$

$$
= \sum_{\tilde{z}[k-1]} p(\tilde{z}[k-1], y_0^{K-1}) \mid R_h) \times
$$

$$
\tilde{\alpha}_F(\tilde{z}[k-1]) \tilde{\beta}_F(\tilde{z}[k]), \quad (31)
$$

$$
The essential assumption used in the OSE is the indepen-
dence of adjacent subbursts:

$$
p(y \mid \tilde{D}, R_h) \approx \prod_{k=0}^{K-1} p(y_{K+1}, y_{K+2} \mid \tilde{D}_{K+1}, R_h).
$$

Therefore, the decomposition procedure for the relevant prob-
ability can be rewritten as

$$
p(y, \tilde{z}[k] \mid R_h) = p(y, y_{K+1}, y_{K+2} \mid \tilde{D}, R_h)
$$

$$
= p(y_{K+1}, y_{K+2} \mid \tilde{D}_{K+1}, R_h)
$$

$$
\sum_{\tilde{z}[k] \in \mathbb{Z}} p(\tilde{z}[k], y_{K+1}, y_{K+2} \mid \tilde{D}_{K+1}, R_h)
$$

$$
\times \gamma_{\tilde{z}[k]} \tilde{\alpha}_{\tilde{z}[k]}(\tilde{z}[k]) \tilde{\beta}_{\tilde{z}[k]}(\tilde{z}[k]), \quad (32)
$$

where (33) is essential for the derivation.

REFERENCES


with implications for fading channel MLSD,” IEEE Trans. Commun.,

MLSD algorithm using fractionally-spaced linear prediction,” in Proc.

art,” in Proc. IEEE International Symposium on Turbo Coders '97, pp. 1-

1680, Dec. 1996.

DPSK systems,” in Proc. 4th Int. ITG Conf. on Source and Channel


maximum a posteriori algorithms suitable for turbo-decoding,” ETT,