On Iterative Receivers with Channel Estimation for Serially Concatenated CPM

Michael Anderson, Zhenning Shi, Mark C. Reed and Gerard Borg

***Abstract*— A novel iterative receiver for the highly energy and bandwidth efficient serially concatenated continuous phase modulation (SCCPM) is proposed. The receiver uses soft feedback information from a simplified demodulator to perform iterative equalization and estimation while also utilising iterative decoding. The Laurent signal model for CPM is exploited to reduce the complexity of all the receiver components. The proposed equalizer consists of interference cancellation followed by linear filtering under the MMSE criteria. Three well-known iterative channel impulse response (CIR) estimation techniques that use training symbols and soft information are compared for CPM, and the least squares (LS) algorithm is found to be the best practical choice. Iterative SNR estimation based on a simple averaging technique is also integrated into the receiver. Heuristic methods for switching between one-shot training based and iterative CIR and SNR estimation are proposed. These methods are shown to be effective through simulations based on randomly generated *a-priori* information. Numerical BER simulations show that the proposed receiver performs well compared to the case when the channel parameters are perfectly known.**

***Index Terms*— Continuous Phase Modulation, iterative detection, equalization, channel estimation.**

I. Introduction

Continuous Phase Modulation (CPM) is widely used in commercial and military wireless communications due to its excellent energy and bandwidth efficiency and its constant envelope [1]. It is well known that iterative (turbo) processing can produce excellent results with reasonable complexity for a variety of estimation and detection problems [2]. Given these two facts, it is our aim to design a high performance receiver for CPM using turbo methods. Turbo decoding has been applied to serially concatenated CPM (SCCPM) to yield powerful communications systems in the context of additive white Gaussian noise (AWGN) [3] and Rayleigh fading channels [4].

In this work, we are interested in the case of CPM signals transmitted over frequency-selective channels. This requires equalization to combat inter-symbol interference (ISI). In the first turbo equalization scheme [5] a MAP equalizer was proposed. In more recent work, several reduced complexity turbo equalization schemes have been presented mainly based on soft interference cancellation and/or linear filtering, e.g. [6]–[8]. We propose an iterative interference cancellation type equalizer for CPM combined with minimum mean-squared error (MMSE) filtering. Similar equalizers have been proposed for linear modulations in [7] and CDMA in [9]. Non-iterative equalizers have been derived for CPM in [10] and [11], which only consider the ISI inherent in CPM signals, not the ISI caused by a physical channel. An equalizer similar to the one we propose has been suggested independently in [12]. Our equalizer however, has the advantage that it is based on the Laurent signal model, which subsequently allows for a significant decrease in the decoding complexity as outlined in [11].

Many authors when dealing with detection problems assume that the channel impulse response (CIR) and signal-to-noise ratio (SNR) are known at the receiver. We are interested in the realistic case where these parameters need to be estimated. Recently, iterative methods for doing this have shown excellent results, see for example [13], [14]. To the author's knowledge, iterative channel estimation for CPM has not yet been treated in the literature. We consider a system that employs training symbols in a block fading environment, such that the channel is time-varying but assumed to be constant for a given number of symbols. We aim to evaluate several well-known channel estimation techniques when employed as part of our iterative CPM receiver. It is also of interest to investigate when soft data-aided channel estimation is beneficial over purely training-aided estimation. Analytical criteria for doing this have been derived when estimating the CIR in [15] and [16]. Such criteria can often be complex to determine and inaccurate for long CIRs. We therefore propose a simple empirical method for switching between training and soft data-aided channel estimation.

SNR estimation algorithms that use soft data, and therefore make up part of an iterative channel estimator, have also been studied in the literature, see for example [17]. An estimate of the SNR may be required for the CIR estimator and/or the equalizer. Based on the findings in [18], we integrate a simple averaging SNR estimator into our receiver, and examine the performance of the overall system through simulation. To obtain an estimate of the SNR, an estimate of the CIR is required. We therefore also study the effect that an imperfect CIR estimate has on that of the SNR. As with the CIR, it is necessary to have some criteria for switching between training-aided and soft data-aided estimation. An analytical method for doing this is again difficult to derive, so we suggest another empirical technique based upon simulation results.

The remainder of this paper is arranged as follows. Section II describes the system model including the signal, channel

M. Anderson is with the Research School of Physical Sciences and Engineering, The Australian National University, Canberra ACT 0200, Australia and National ICT Australia. e-mail: michael.anderson@anu.edu.au

Z. Shi and M. C. Reed are with National ICT Australia and affiliated with the Australian National University. e-mail: {mark.reed,zhenning.shi}@nicta.com.au

G. Borg is with the Research School of Physical Sciences and Engineering and the Department of Engineering, The Australian National University. e-mail: gerard.borg@anu.edu.au

and receiver models, while Section III describes the proposed iterative equalizer for CPM. Section IV outlines several CIR estimation algorithms and presents results for these algorithms assuming a known SNR. Section V describes the SNR estimation algorithm that we employ and presents results for the overall iterative SCCPM receiver. Finally, Section VI concludes the paper.

II. System Model

A. Signal Model

We make use of the Laurent decomposition of CPM [19], whereby the complex baseband signal can be represented as the sum of $Q = 2^{L-1}$ PAM pulses, where L is the memory of the modulation, i.e. for normalized pulses

$$s(t, \mathbf{u}) \quad = \quad \sum_{k=0}^{Q-1} \sum_{n} a_{k,n} g_k(t - nT), \tag{1}$$

where the $g_k(t)$ are pulse shaping filters with lengths ranging from T to $(L+1)T$ and the $a_{k,n}$ have a non-linear relationship to the input bits $u_n \in \{-1, 1\}$. The Laurent decomposition is extremely useful in receiver design because for M-ary modulation, the majority of the signal energy lies in the first $M-1$ pulses. Therefore the receiver can be simplified by only accounting for these pulses resulting in a very small sacrifice in performance. In this work we focus on GMSK binary signalling with a modulation index of $h = 1/2$ because of the useful properties of this class of signals. In this case the $a_{k,n}$, which are referred to as "pseudosymbols", are in the set $\{\pm 1, \pm j\}$ and alternate between real and imaginary values. For each k, the sequence can be considered the output of a recursive encoder [11] allowing, for example, MAP decoding to be applied. Since we only need to consider one principal pulse for M=2, the simplified receiver consists of just one filter, for example $g_0(-t)$, and a decoder that only considers $a_{0,n}$. For this modulation we have $a_{0,n} = a_{0,n-1} u_n j$. From this point forward, as in [10] for GMSK, we approximate the CPM signal with the first two pulses, i.e.

$$s(t) \quad \approx \quad \sum_{n} \left(a_{0,n} g_0(t - nT) + a_{1,n} g_1(t - nT) \right) \tag{2}$$

B. Channel Model

We choose to align ourselves with the GSM standard and use the specified EQ channel [20] as the physical channel in all our simulations. We assume wide-sense stationary uncorrelated scattering (WSSUS) and use a Jake's simulator [21] with a low enough Doppler frequency to effectively achieve block fading. Considering just the first term of the approximation in (2) for now, the equivalent channel for $a_{0,n}$ can be written as

$$h_0(t) \quad = \quad f_T(t) \otimes p(t) \otimes f_R(t), \tag{3}$$

where $f_T(t) = g_0(t)$, $p(t)$ is the physical channel, $f_R(t)$ is the receiver filter and $\otimes$ denotes the convolution operator. The part of the received filtered signal that contains $a_{0,n}$, denoted $r_0(t)$, can be written as

$$r_0(t) \quad = \quad \sum_{j} a_{0,j} h_0(t - jT) \tag{4}$$

We decide to sample at the symbol rate because it provides a very convenient signal model and has also been used successfully in similar situations where sufficient statistics are not easily obtained [22]. The sampled signal is

$$r_0(nT) = \sum_{l} a_{0,n-l} h_0(lT) = \mathbf{h_0}^T \mathbf{a_{0,n}} \tag{5}$$

where

$$\mathbf{h_0} \quad = \quad [h_{0,-L_0^1}, \cdots, h_{0,0}, \cdots, h_{0,L_0^2}]^T \tag{6}$$

$$\mathbf{a_{0,n}} \quad = \quad [a_{0,n-L_0^1} \cdots a_{0,n} \cdots a_{0,n+L_0^2}]^T, \tag{7}$$

and $\mathbf{h_0}$ contains $L_0^1 + 1$ and L_0^2 causal and non-causal symbol-spaced taps respectively. A similar channel can be derived for $a_{1,n}$. The sampled received signal can then be written as

$$r_n \quad \approx \quad \mathbf{h_0}^T \mathbf{a_{0,n}} + \mathbf{h_1}^T \mathbf{a_{1,n}} + w_n \;, \tag{8}$$

where w_n is filtered AWGN that has two-sided power $N_0 = 2\sigma^2$.

C. Overall System Model

Fig. 1 shows the overall system model. A block of source bits are convolutionally encoded, randomly interleaved and then conceptually passed through a set of Laurent encoders to give Q blocks of N pseudosymbols. P training symbols are then multiplexed into the a_0 data stream which will be explained further in Section III. The symbols are then modulated and transmitted over the physical channel $p(t)$ which is assumed to be constant for the duration of the block. The signal is subject to noise and filtered to given r_n as given in (8). The training symbols are then used to give an initial estimate of $\mathbf{h_0}$ and possibly $\mathbf{h_1}$, as well as the SNR, before the signal is equalized. The output of the equalizer is then passed to a CPM MAP decoder which combined with the channel MAP decoder performs standard SCCPM iterative decoding by exchanging log-likelihood ratios (LLRs) denoted $\lambda(\cdot;\cdot)$. After the zeroth iteration, the LLRs $\lambda(\mathbf{a_0}; \mathbf{O})$ are available and are used for iterative equalization and possibly re-estimation of the channel.

III. Iterative Equalization

As noted in Section II-A and shown in [11], only a very small performance degradation is suffered when decoding over $a_{0,n}$ only as opposed to all Q pseudosymbols. Since we are interested in keeping complexity as low as possible, we decide to design our CPM MAP decoder in this way. As such, the turbo equalizer need only produce LLRs for $a_{0,n}$, i.e. $\lambda(\mathbf{a_0}; \mathbf{I})$. Since we plan to use linear filtering as part of the equalization process, it is useful to define a window of received data as

$$\mathbf{r_n} \quad \approx \quad \mathbf{H_0} \mathbf{a_{0,n}} + \mathbf{H_1} \mathbf{a_{1,n}} + \mathbf{w_n}, \tag{9}$$

where

$$\mathbf{r_n} \quad = \quad [r_{n-N_1}, \cdots, r_n, \cdots, r_{n+N_2}]^T \tag{10}$$

$$\mathbf{a_{0,n}} \quad = \quad [a_{0,n-N_1-L_0^1}, \cdots, a_{0,n}, \cdots, a_{0,n+N_2+L_0^2}]^T \tag{11}$$

$$\mathbf{w_n} \quad = \quad [w_{n-N_1}, \cdots, w_n, \cdots, w_{n+N_2}]^T, \tag{12}$$

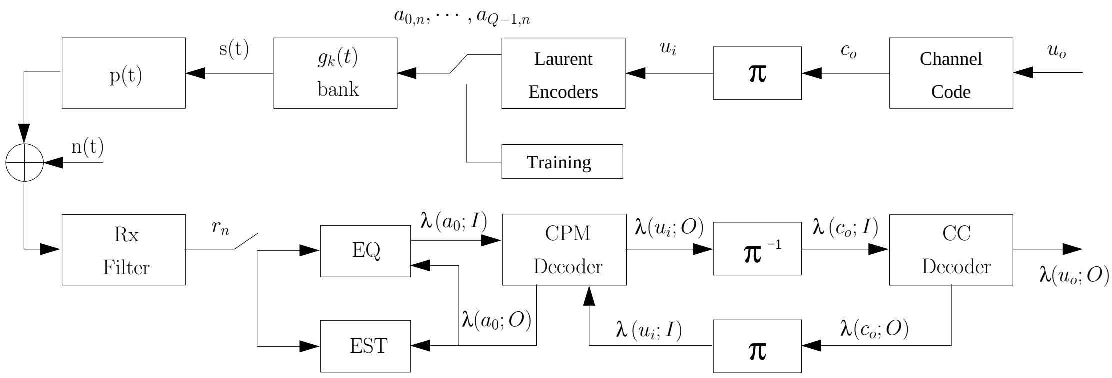

Fig. 1. CPM based communication system employing full turbo estimation/detection

and $\mathbf{H_0}$ is the $(N_1+N_2+1) \times (N_1+N_2+L_0^1+L_0^2+1)$ channel matrix where $[\mathbf{H_0}]_{i,j} = h_{-L_0^1+j-i}$ when $0 \leq j-i \leq L_0^1+L_0^2$ and 0 otherwise. Similar definitions exist for $\mathbf{H_1}$ and $\mathbf{a_{1,n}}$ as for $\mathbf{H_0}$ and $\mathbf{a_{0,n}}$ respectively. We now define the LLR output from the CPM decoder as (for $a_{0,n}$ real)

$$\lambda(a_{0,n}; O) = \log \frac{\Pr\{a_{0,n} = 1 | \lambda(\mathbf{a_0}; \mathbf{I})\}}{\Pr\{a_{0,n} = -1 | \lambda(\mathbf{a_0}; \mathbf{I})\}} \quad (13)$$

A-posteriori as opposed to extrinsic information is deliberately fed back to the equalizer as this was found to produce better results. This is consistent with the observations of other researchers [22]. These LLRs are now used as *a-priori* information for equalization. We remark that in a typical turbo system we would have an interleaver between the decoder and the equalizer. Unfortunately this is not practically possible in the case of CPM due to the statistical dependencies between the pseudosymbols. The exception to this is the special case of full-response CPM [23]. Soft data symbols can now be defined as

$$\tilde{a}_{0,n} = \begin{cases} \tanh\left(\frac{\lambda(a_{0,n};O)}{2}\right) & \text{n even} \\ \tanh\left(\frac{\lambda(a_{0,n};O)}{2}\right) * j & \text{n odd} \end{cases} \quad (14)$$

If the channel is assumed to be known, these soft data symbols can be used to cancel interference from neighbouring bits to give

$$\mathbf{y_n} \approx \mathbf{H_0 a_{0,n}} - \mathbf{H_0 \tilde{a}_{0,n}} + \mathbf{H_1 a_{1,n}} + \mathbf{w_n}, \quad (15)$$

where

$$\mathbf{a_{0,n}} = [\tilde{a}_{0,n-N_1-L_0^1}, \cdots, \tilde{a}_{0,n-1}, 0, \tilde{a}_{0,n+1}, \cdots, \tilde{a}_{0,n+N_2+L_0^2}]^T \quad (16)$$

Since only $a_{0,n}$ statistics are used for decoding, estimates of $a_{1,n}$ and the other pseudosymbols cannot be calculated. It is therefore not possible to cancel the interference caused by these terms. However, this detrimental effect will be minimal compared to the interference caused by the $a_{0,n}$ terms. If we consider y_n to be an updated estimate of $a_{0,n}$, it can be improved through linear filtering using the MMSE criteria. As there is *a-priori* information available, the coefficients, $\mathbf{f_n}$, are time-varying. The filter output is given by $\hat{a}_{0,n} = \mathbf{f_n}^H \mathbf{y_n}$, where

$$\mathbf{f_n} = [f_{n-N_1}, \cdots, f_n, \cdots, f_{n+N_2}], \quad (17)$$

such that the number of filter coefficients is $N_1 + N_2 + 1$. These coefficients are calculated according to

$$\begin{aligned} \mathbf{f_n} &= \arg\min_{\mathbf{f_n}} \mathsf{E}\{|a_{0,n} - \mathbf{f_n}^H \mathbf{y_n}|^2\} \\ &= \arg\min_{\mathbf{f_n}} \mathbf{f_n}^H \mathsf{E}\{\mathbf{y_n y_n}^H\}\mathbf{f} - 2\mathbf{f_n}^H \mathsf{E}\{a_{0,n}\mathbf{y_n}\} \end{aligned} \quad (18)$$

Due to space limitations we do not complete the full derivation of $\mathbf{f_n}$. After some manipulations, it is given by

$$\mathbf{f_n} = \left[\mathbf{H_0 V_0 H_0}^H + \mathbf{H_1 H_1}^H + 2\sigma^2 \mathbf{W}\right]^{-1} \mathbf{H_0}^{N_1+L_0^1+1}, \quad (19)$$

where $\mathbf{H_0}^i$ denotes the i^{th} column of $\mathbf{H_0}$, the elements of $\mathbf{W}$ are defined by

$$[\mathbf{W}]_{i,j} = \int_{-\infty}^{\infty} f_R^*(-t) f_R((i-j)T - t) dt \quad (20)$$

and

$$\mathbf{V_0} = \text{diag}[v_{0,n-N_1-L_0^1}, \cdots, v_{n-1}, 1, v_{n+1}, \cdots, v_{n+N_2+L_0^2}], \quad (21)$$

where $v_i = 1 - |\tilde{a}_i|^2$. Here we have used the fact that for $h = 1/2$, $\mathsf{E}\{a_{k,n} a_{l,m}^*\} = \mathsf{E}\{a_{k,n}\}\mathsf{E}\{a_{l,m}^*\}$. We are also making the assumption that $\mathsf{E}\{a_{0,n}\} = \tilde{a}_{0,n}$. The output of this filter is very well approximated by a Gaussian distribution in similar contexts [7] [9]. We therefore write $\hat{a}_{0,n} = \mu_n a_{0,n} + \eta_n$ where μ_n is the equivalent gain and $\eta_{2n} \sim \mathcal{N}(0, \rho^2)$. It can be shown that these values are given by $\mu_n = \mathbf{f}^H \mathbf{H_0}^{N_1+L_0^1+1}$ and $\rho^2 = \mu - |\mu|^2$. The distribution $p(\hat{a}_{0,n} | a_{0,n}{=}a_0)$ required to calculate $\lambda(a_{0,n}; I)$ is now fully specified. It can be seen from (19) that estimates of both $\mathbf{h_0}$ and $\mathbf{h_1}$ are required to implement the equalizer. Since a composite symbol-spaced channel is being estimated from an underlying physical channel that is not symbol-spaced, determining an estimate of one of these channels from an estimate of the other may be non-trivial. Given this, and since $a_{1,n}$ carries only a fraction of the amount of energy that $a_{0,n}$ carries, it is of interest to know how much benefit is actually gained from including the $a_{1,n}$ statistics in the calculation of $\mathbf{f_n}$. We determine this by running a simulation to compare the performance of the system when $\mathbf{f_n}$ is calculated as in (19) to the performance of the system when the $\mathbf{H_1 H_1}^H$ term is simply neglected. The results are not presented here, but we find that including the $a_{1,n}$ statistics provides a negligible improvement in terms of bit-error-rate

(BER). We therefore conclude that it is not worth estimating $\mathbf{h_1}$ for the purpose of equalization. As such, we only focus on estimating $\mathbf{h_0}$ in the sequel.

To fully evaluate the performance of the proposed equalizer, its complexity must be considered. The most computationally expensive operation is the inversion of the matrix in (19). In general, the complexity of inverting a $K \times K$ matrix is $O(K^3)$. In this case $K = N_1 + N_2 + 1$. In [9] a procedure is described for recursively calculating the inverse of a similar matrix to reduce the complexity to $O(K^2)$. This technique can also be applied here. The complexity of the simplified CPM MAP decoder is only $O(2)$ making the total complexity of the equalizer/decoder $O(K^2) + O(2)$. The complexity of a full CPM MAP detector with modulation index $h{=}q/p$ is $O(p2^{L+L_c+1})$ where $L_c{=}L_0^1 + L_0^2 + 1$ is the total memory of the physical channel. The number of coefficients of the filter must be increased for longer channels, but the complexity of the proposed detector is still only polynomial in K. This is compared to the MAP detector that has exponential complexity in both the channel and modulation memory.

IV. Iterative CIR Estimation

We now turn our attention to determining the performance of the detector proposed in the previous Section when the CIR must be estimated. We consider three well-known estimation algorithms, namely the least squares (LS), the linear minimum mean-square error (LMMSE) [24] and the least mean-square (LMS) [25] algorithms. The system being considered uses training symbols to obtain an initial estimate of the CIR. On average, a GSM burst contains 26 training symbols and 116 data symbols [20]. We therefore consider a block fading system whereby the channel remains approximately constant for 142 symbols with a preamble of 26 training symbols followed by 116 data symbols.

A. Optimal training sequence

Optimal training sequences for the GSM and EDGE standards are well known. Although these standards employ GMSK modulation (or at least Laurent filtering in the case of EDGE), they don't treat the CPM modulation as a recursive code and therefore don't use a true SCCPM system. Instead the entire modulation, from the output of the interleaver onwards, is simply considered part of the channel. Referring to Fig. 1, it is then the u_i channel that needs to be estimated as opposed to the $a_{0,n}$ channel as in our case. Since a non-linear relationship exists between u_n and $a_{0,n}$, there is no trivial translation that can be performed on the known GSM training sequences to obtain the required $a_{0,n}$ sequences. If in general the first P symbols in a block are training symbols, then the approximate received signal due to these symbols can be written as

$$\mathbf{r_p} \approx \mathbf{A_{0,p}h_0} + \mathbf{A_{1,p}h_1} + \mathbf{w}, \tag{22}$$

where $\mathbf{r_p}{=}[r_{L_0^1}, \cdots, r_{P-L_0^2-1}]^T$, $\mathbf{w}{=}[w_{L_0^1}, \cdots, w_{P-L_0^2-1}]^T$, $[\mathbf{A_{0,p}}]_{i,j}{=}a_{0,L_0^1+L_0^2+i-j}$ and a similar definition exists for $\mathbf{A_{1,p}}$. Although we do not intend to estimate $\mathbf{h_1}$ directly for equalization, it may also be possible to obtain a better estimate of $\mathbf{h_0}$ if we estimate $\mathbf{h_0}$ and $\mathbf{h_1}$ jointly. However as with the equalization process, we find that only a negligible improvement in the estimate can be achieved by doing this. For the purpose of estimation, we therefore further approximate the received signal as $\mathbf{r_p} \approx \mathbf{A_{0,p}h_0} + \mathbf{w}$. Given that for $L{=}4$ approximately 99.2% of the energy is accounted for by $a_{0,n}$, we feel comfortable with this approximation.

To obtain the optimal training sequence, we maximize the normalized signal to estimation error as described in [26]. For LS and LMMSE estimation, this is done by maximizing $\mathrm{tr}\{\mathbf{M}\}$[1] where $\mathbf{M} = [\mathbf{A_{0,p}^H A_{0,p}}]^{-1}$. It turns out that this also gives a good training sequence for LMS estimation [14]. Based on this criteria, a computer search was carried out to find the best length 26 training sequence. Care must be taken in multiplexing the training sequence into the data stream since the $a_{0,n}$ have memory. We require two training sequences with different final values to ensure that the trellis "fits together" for decoding when the training information is removed. The two sequences used, both of which are optimal, are shown in Table I. We remark that these sequences are not unique.

TS1	j,1,j,1,-j,1,j,1,-j,-1,-j,1,j,1,j,1,-j,-1,-j,-1,-j,1,-j,-1,-j,1
TS2	j,-1,j,-1,-j,-1,j,-1,-j,1,-j,-1,j,-1,j,-1,-j,1,-j,1,-j,-1,-j,1,-j,-1

TABLE I
Optimal training sequences

B. Training based CIR estimation

Given the approximation that we are using for the received signal, we use the notation $\mathbf{h}$ in place of $\mathbf{h_0}$ for convenience. The one-shot training-aided channel estimates using the LS, LMMSE and LMS algorithms respectively are

$$\hat{\mathbf{h}}_{LS}^P = [\mathbf{A_{0,p}^H A_{0,p}}]^{-1}\mathbf{A_{0,p}^H r_p} \tag{23}$$

$$\hat{\mathbf{h}}_{LMMSE}^P = \mathbf{C_{hh}A_{0,p}^H}[\mathbf{A_{0,p}C_{hh}A_{0,p}^H} + 2\sigma^2\mathbf{W}]^{-1}\mathbf{r_p} \tag{24}$$

$$\hat{\mathbf{h}}_{LMS,p}^P = \hat{\mathbf{h}}_{LMS,p-1}^P + \mu\mathbf{a_{0,p}^*}(r_p - \hat{\mathbf{h}}_{LMS,p-1}^T\mathbf{a_{0,p}}), \quad p = 0 \cdots P-1 \tag{25}$$

In (24), $\mathbf{C_{hh}}$ is the channel correlation matrix, with elements defined by

$$[\mathbf{C_{hh}}]_{i,j} = \int_{-\infty}^{\infty} R(t) f_E(-t-mT) f_E^*(-t-nT)dt,$$

where $m = -L_0^1 + i - 1$, $n = -L_0^1 + j - 1$ and under the assumption of a WSSUS channel, we define

$$\mathsf{E}\{p(t)p^*(\tau)\} = \begin{cases} R(t) & t = \tau \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

In (25), $\hat{\mathbf{h}}_{LMS,p-1}^P$ is the estimate from the previous iteration with $\hat{\mathbf{h}}_{LMS,0}{=}\mathsf{E}\{\mathbf{h}\}{=}\mathbf{0}$, $\mathbf{a_{0,p}} = [a_{0,p-L_0^1} \cdots a_{0,p} \cdots a_{0,p+L_0^2}]^T$ which is padded with zeros when required, and $0 < \mu \leq 1$ is the step size.

Each of the estimation techniques has its pros and cons. The biggest disadvantage of the LS algorithm is its complexity due to the required matrix inversion. This also requires $\mathbf{A_{0,p}}$

[1] tr denotes the trace of a matrix

to be full rank which is not guaranteed if the number of training symbols is small compared to the memory of the channel. LMMSE estimation also involves a matrix inversion, only without the same full-rank requirement of $\mathbf{A_{0,p}}$. It does however rely upon *a-priori* information being available for the channel, i.e. it requires $R(t)$ to be known which may not be the case. Depending on the physical channel, it might also be possible to estimate $R(t)$ with a predetermined function, for example an exponential profile. The LMS algorithm has low-complexity, but generally speaking does not perform as well as the other estimators.

C. Iterative data-aided CIR estimation

After the zeroth iteration, soft data values are available for estimation. As an example, the data-aided LS channel estimate is given by

$$\hat{\mathbf{h}}_{LS}^{D} \;=\; [\mathbf{A}_{\mathbf{0,d}}^{H}\mathbf{A}_{\mathbf{0,d}}]^{-1}\mathbf{A}_{\mathbf{0,d}}^{H}\mathbf{r}_{\mathbf{d}}\,, \qquad (27)$$

where $[\mathbf{A_{0,d}}]_{i,j}=[\mathbf{a_{0,d}}]_{L_0^1+L_0^2+i-j}$, the vector of data estimates is given by $\mathbf{a_{0,d}}=[a_{0,0},\cdots,a_{0,P-1},\tilde{a}_{0,P},\cdots,\tilde{a}_{0,N-1}]$ and $\mathbf{r_d}=[r_{L_0^1},\cdots,r_{N-L_0^2-1}]^T$. LMMSE and LMS estimation using soft values are performed in a similar way. It has been established in the literature that when using iterative estimation, the CIR should only be re-estimated when an improvement on the initial training based estimate can be gained [15]. Ideally this means that data-aided estimation should only be used when the criteria $\sigma^2_{\Delta h,d} < \sigma^2_{\Delta h,p}$ where $\sigma^2_{\Delta h,p}$ and $\sigma^2_{\Delta h,d}$ are the analytically determined training and data-aided estimation error variances respectively. Defining the error as $\mathbf{\Delta h} = \hat{\mathbf{h}} - \mathbf{h}$, and again using LS estimation as an example, it is easy to show that $\sigma^2_{\mathbf{\Delta h,p}} = 2\sigma^2\mathrm{diag}\{\mathbf{M}\}$. This is straightforward to evaluate assuming the noise variance is known. Similarly, the data-aided error is found to be

$$\mathbf{\Delta h_d} \;=\; [\mathbf{M}\mathbf{A}_{\mathbf{0,d}}^{H}\mathbf{A_0} \;-\mathbf{I}]\mathbf{h} + \mathbf{M}\mathbf{A}_{\mathbf{0,d}}^{H}\mathbf{w}, \qquad (28)$$

which makes calculating $\sigma^2_{\mathbf{\Delta h,d}}$ difficult since it is dependent on $\mathbf{h}$. Empirical approximations for finding $\sigma^2_{\mathbf{\Delta h,d}}$ for LMS and other estimation techniques have been suggested in [15] and [16]. We have found however, that these are not necessarily robust for long CIRs, and therefore suggest a simple alternate adaptive criteria. Firstly, note that the average variance of a block of soft symbols is given by $\overline{v} = \sum_{i=P}^{P+N-1} v_i$, where v_i is defined in (21), and the average training-aided estimation error is given by $\overline{\sigma}^2_{\Delta h,p} = \frac{1}{L_0^1+L_0^2+1}\sum_{i=-L_0^1}^{L_0^2}[\sigma^2_{\Delta h,p}]_i$ where $[\sigma^2_{\Delta h,p}]_i$ is the error variance of tap estimate $\hat{h}_i$. We then propose that the initial training based channel estimate should only be used when the normalized training error is less than $\overline{v}$, i.e. when $\overline{\sigma}^2_{\Delta h,p} < \alpha\frac{\hat{E}_h}{N}\overline{v}$, where $\hat{E}_h = \sum_{i=-L_0^1}^{L_0^2}\hat{h}_i^D(\hat{h}_i^D)^*$ and α is a constant. To test this hypothesis we plot the normalized mean-squared error,

$$MSE_h \;=\; \frac{\sum_{i=-L_0^1}^{L_0^2}\mathsf{E}\{|\hat{h}_i - h_i|^2\}}{\sum_{i=-L_0^1}^{L_0^2}\mathsf{E}\{|h_i|^2\}}, \qquad (29)$$

by generating channel estimates based on random Gaussian i.i.d $\lambda(a_0;O)$ and corresponding $\tilde{a}_{0,n}$ for a given average mutual information I_A as described in [27]. This reflects an approximation of the real system where the $\tilde{a}_{0,n}$ are correlated, and the signal also includes $a_{1\cdots Q-1,n}$. We remark that in most standard turbo systems, the estimates will be roughly independent due to interleaving, so these results readily lend themselves to such systems. The block size, and all other relevant simulation parameters, are the same as previously described. Fig. 2 shows numerical results for low $I_A(I_A \approx 0.1)$, mid-range $I_A(I_A \approx 0.5)$ and high $I_A(I_A \approx 0.9)$ for $\alpha = 1.1$. Every point is averaged over 100,000 blocks, each with a different EQ channel realization. We note that the training-aided estimation error is independent of I_A. The use

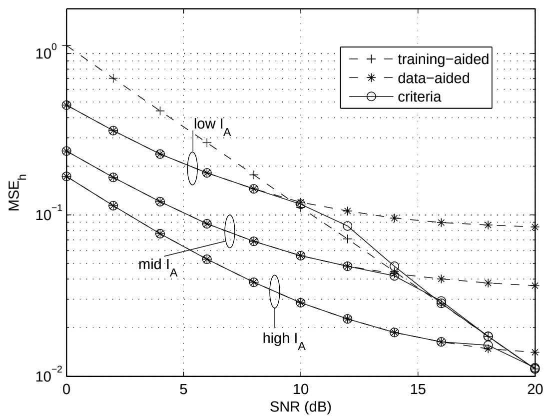

Fig. 2. Effect of using switching criteria in channel estimation

of this criteria (referred to as "criteria" in Fig. 2) appears to work well, because the *average* performance over the considered SNR range is better than that of training or data-aided estimation alone. In this case, α was chosen to give the best performance over a large SNR range for demonstration purposes. In general however, the approximate operating range of the receiver should be taken into account. The proposed method should also be effective when using LMMSE or LMS estimation, however since LS estimation is considered for the overall system design in the sequel, these cases were not considered further.

Next we compare the BER performance for each of the considered estimation algorithms for a known SNR. The simulation parameters used are as follows. Referring to Fig. 1, we consider a convolutional rate $1/2$ non-recursive channel code with generator [5,7]. The interleaver is pseudorandom of size 2048. The CPM scheme employed is binary GMSK with modulation index $h{=}1/2$ and memory $L{=}4$. As previously stated, the GSM EQ physical channel is used. The receiver filter used was a square-root raised cosine (SRRC) filter with roll-off factor 0.3. While this doesn't generate sufficient statistics, it turns out to be a good choice practically when the channel is unknown. It is straightforward to implement and also has the advantage that AWGN sampled at the symbol rate at the filter output remains white. To implement the equalizer, the channel estimate is used to obtain $\hat{\mathbf{H}}_\mathbf{0}$ which is used in place of $\mathbf{H_0}$ in (19). We remark that a small performance gain may be achieved by including the statistics of the channel estimation error in the derivation of the equalizer, however doing so is outside the scope of this work. Numerical results are shown in Fig. 3 for six iterations of the receiver. It was found that little to no improvement can be gained after six iterations when the CIR has to be estimated. $R(t)$ is assumed

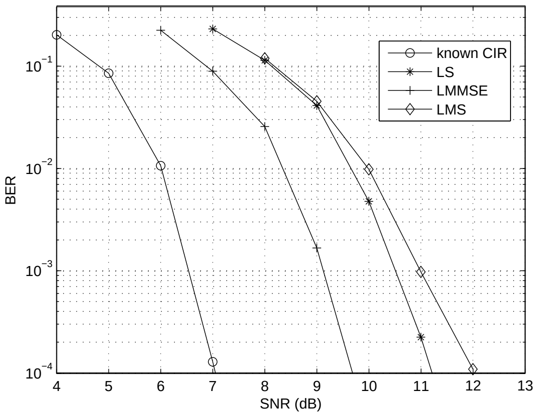

Fig. 3. Comparison of iterative channel estimation methods

to be known perfectly for the purpose of LMMSE estimation in this case. The step size μ is set to 0.05 for training symbols and 0.02 for soft data estimates for LMS estimation. As expected, LMMSE estimation performs the best in terms of BER, and is only approximately 2.5 dB worse than when the CIR is known perfectly. LS and LMS estimation are approximately 4.2 dB and 5 dB worse than perfect CIR knowledge respectively. The use of the switching criteria with the LS algorithm only gave a negligible performance gain in this case, so the results for when it wasn't used are omitted. It is unlikely that $R(t)$ would be known exactly *a-priori*, meaning that it would need to be approximated with a known function. An estimate of the noise variance is also required to use LMMSE. This creates a difficult situation because an estimate of the channel is required to estimate the variance as will be shown in the sequel. Given these facts, we conclude that LS is likely to be preferable to LMMSE, especially considering that its performance is only about 1.5 dB worse than that of LMMSE. We remark that a method for utilising LMMSE estimation when the noise variance is unknown is currently under investigation. Fig. 3 shows that LMS is only approximately 0.7 dB worse than LS estimation. It also has lower complexity. In the block fading environment that we consider however, the estimation complexity is very low compared to the equalization complexity. We therefore conclude that under the considered conditions, LS is superior to both LMMSE and LMS. As such, it is the estimation technique considered in the sequel.

V. Iterative SNR Estimation

We now focus on the case when the noise variance must be estimated in addition to the CIR for use in (19). Just as with CIR estimation, it is possible to use soft data estimates in addition to training symbols to iteratively estimate the noise variance. It has been shown in [18] that the minimum variance unbiased estimator (MVUE) outperforms other standard SNR estimation algorithms, and it is therefore the one we concentrate on. Its output when using training symbols and soft estimates is given by

$$\hat{N}_0^D = \frac{1}{P+N-(L_0^1+L_0^2+1)} \sum_{i=L_0^1}^{P+N-L_0^2-1} \left| r_i - \sum_{j=-L_0^1}^{L_0^2} \hat{h}_j [\mathbf{a_{0,d}}]_{i+j} \right|^2 \qquad (30)$$

which can easily be altered to use training symbols only. The higher order pseudosymbols are again neglected due to the trivial amount of signal energy they account for. Analogous to estimating the CIR, we require a criteria for switching between $\hat{N}_0^P$ and $\hat{N}_0^D$ based on the quality of the soft information. The situation is complicated by the fact that (30) requires an estimate of the CIR. An analysis is again very difficult, so we search for other heuristic solutions. Fig. 2 shows that at low SNR, it is better to use $\hat{\mathbf{h}}^D$ than $\hat{\mathbf{h}}^P$ (based on the LS algorithm). We use the notation $(\hat{N}_0^D)^D$ to denote a variance estimate that is calculated using soft information directly according to (30), and is also based on a CIR estimate that was calculated using soft information. $(\hat{N}_0^D)^P$ on the other hand, denotes an estimate calculated using soft information, but is based on a CIR estimate calculated using only training symbols.

Simulations also show that at low SNR, there is a crossing point between $(\hat{N}_0^P)^D$ and $(\hat{N}_0^D)^D$. The switching criteria that we propose to deal with this is based upon one suggested in [18], however our analysis treats the case where the channel energy is not necessarily unity, and the CIR may be estimated using training and soft data symbols. The main idea is that the soft data should only be used to re-estimate the noise variance if the initial training based estimate is significantly greater than the variance of the soft information, i.e. when $(\hat{N}_0^P)^D > \beta \hat{E}_h \overline{v}$. We must also employ the switching criteria from the previous Section to ensure that we are using the best possible CIR estimate to calculate the noise variance. This is difficult because the noise variance is required to calculate $\sigma_{\mathbf{\Delta h,p}}$. We therefore must define an approximate value as $\hat{\sigma}_{\mathbf{\Delta h,p}} = (\hat{N}_0^P)^P \text{diag}\{M\}$ and the corresponding average as $\hat{\overline{\sigma}}^2_{\Delta h,p} = \frac{1}{L_0^1+L_0^2+1} \sum_{i=-L_0^1}^{L_0^2} [\overline{\sigma}^2_{\Delta h,p}]_i$. We expect that the best value of α will be different to that from the previous Section since we can no longer calculate the channel estimate error variance exactly.

To sum up, the algorithm used to calculate the variance $\hat{N}_0$ is shown in Fig. 4. Using the same technique as in the previous Section, the normalized noise variance error $\Delta\overline{N}_0 = \mathsf{E}\{(\hat{N}_0 - 2\sigma^2)/(2\sigma^2)^2\}$ is plotted in Fig. 5 for a high range average $I_A (I_A \approx 0.9)$ with $\alpha = 0.6$ and $\beta = 2.2$. Also shown for comparison purposes is the resulting error if perfect switching could be performed. In this case, as the SNR increases, the estimate would ideally follow the sequence $(\hat{N}_0^D)^D \rightarrow (\hat{N}_0^P)^D \rightarrow (\hat{N}_0^P)^P$. This reflects the need for the two separate criteria. An intuitive explanation for why $\Delta\overline{N}_0$ tends to infinite when using soft information is provided in [18]. Fig. 5 shows that the switching criteria (labelled "criteria") works well since it is close to the perfect

IF $(\hat{N}_0^P)^D > \beta \hat{E}_h \overline{v}$

  $\hat{N}_0 = (\hat{N}_0^D)^D$

ELSE

  IF $\hat{\overline{\sigma}}^2_{\Delta h,p} < \alpha \frac{\hat{E}_h}{N} \overline{v}$

    $\hat{N}_0 = (\hat{N}_0^P)^P$

  ELSE

    $\hat{N}_0 = (\hat{N}_0^P)^D$

  END

END

Fig. 4. Switching algorithm for SNR estimation

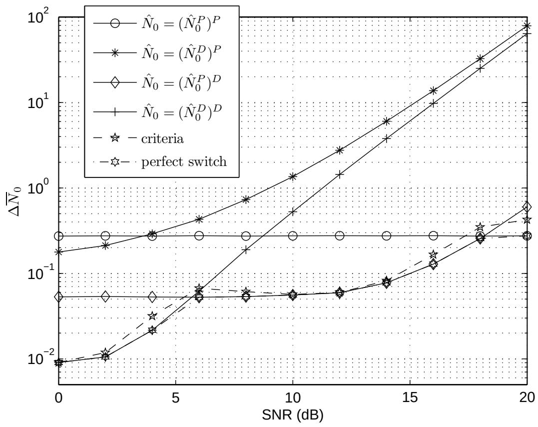

Fig. 5. Effect of using switching criteria in noise variance estimation

switching performance. For illustration purposes, α and β have again been chosen to provide good average performance over the entire considered SNR range. In practice, the operating range of the receiver should be considered.

We are now able to study the BER performance of the overall system of Fig. 1 by replacing $2\sigma^2$ in (19) by $\hat{N}_0$. Numerical results are presented in Fig. 6 using the standard set of simulation parameters for six iterations with the switching criteria in place for both CIR and SNR estimation. To focus on the performance of the SNR algorithm, results are plotted for a known CIR. For this case, there is only an approximate 0.3 dB loss in performance when the noise variance has to be estimated. We note that this is partly due to the strong performance of the estimation algorithm over iterations and partly due to the relative insensitivity of the equalizer to errors in the noise variance. When the CIR is iteratively estimated, the performance loss due to an estimated noise variance is still approximately only 1 dB. The most interesting comparison from Fig. 6 is between the case when the CIR and SNR are known and the case when both need to be estimated. This shows that we only lose about 5 dB in overall performance for the EQ channel with the iterative estimation techniques that have been presented.

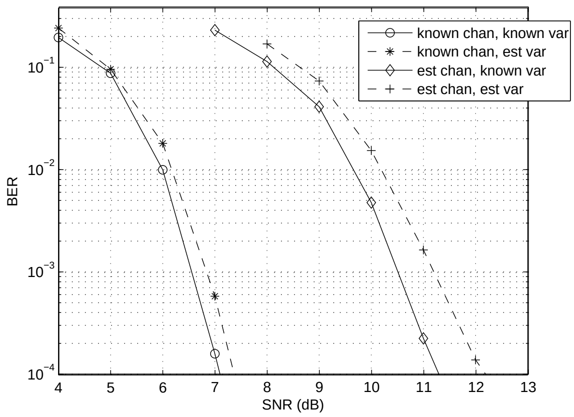

Fig. 6. Overall system performance

VI. Conclusion

We have proposed a novel receiver for SCCPM that combines iterative estimation, equalization and decoding by feeding back soft information from the CPM demodulator. The Laurent signal model has been exploited to greatly simplify the algorithms utilised by the system. The performance of the receiver indicates that it would be a good alternative to highly complex optimal receivers. Future work will focus on extending the capabilities of the receiver to deal with fast-fading channels and also further analysis techniques.

Acknowledgements

The first author would like to thank Markus Dangl from the University of Ulm and Raphaël Le Bidan from ENST for several valuable discussions on iterative channel estimation and equalization. M. Anderson, Z. Shi and M. C. Reed are with National ICT Australia and affiliated with the Australian National University. National ICT Australia is funded through the Australian Government's *backing Australia's ability* initiative and in part through the Australian Research Council.

References

[1] J. B. Anderson, T. Aulin, and C.-E. Sundberg, *Digital Phase Modulation*. Plenum Press, 1986.

[2] J. Hagenauer, "The turbo principle:Tutorial introduction and state of the art," in *Int. Symp. on Turbo Codes and Applications*, (Brest, France), pp. 1–11, Sept. 1997.

[3] P. Moqvist and T. M. Aulin, "Serially concatenated continuous phase modulation with iterative decoding," *IEEE Trans. on Commun.*, vol. 49, pp. 1901–1915, Nov. 2001.

[4] K. Narayanan and G. L. Stüber, "Performance of trellis-coded CPM with iterative demodulation and decoding," *IEEE Trans. Commun.*, vol. 49, pp. 676–687, April 2001.

[5] C. Douillard, C. B. Michel Jezequel, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalisation," *European Transactions on Telecommunications*, vol. 6, pp. 507–511, Sept. 1995.

[6] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. on Turbo Codes and Applications*, (Brest, France), pp. 96–102, Sept. 1997.

[7] M. Tüchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Sig. Proc.*, vol. 50, pp. 673–683, March 2002.

[8] P. D. Alexander, A. J. Grant, and M. C. Reed, "Performance analysis of an iterative decoder for code-division multiple-access," *European Trans. on Telecom.*, vol. 9, pp. 419–426, Sep./Oct. 1998.
[9] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July 1999.
[10] G. K. Kaleh, "Simple coherent receivers for partial response continuous phase modulation," *IEEE J. Selected Areas Commun.*, vol. 7, pp. 1427–1435, Dec. 1989.
[11] M. R. Shane and R. D. Wesel, "Reduced complexity iterative demodulation and decoding of serial concatenated continuous phase modulation," in *Proc. Int. Conf. on Commun.*, vol. 3, pp. 1672–1676, April 2002.
[12] B. Özgul, M. Koca, and H. Deliç, "Doubly iterative MMSE equalization of binary continuous phase modulation," in *Proc. Int. Symp. on Wireless Commun. Systems*, pp. 108–112, Sept. 2005.
[13] M. Sandell, C. Luschi, P. Strauch, and R. Yan, "Iterative channel estimation using soft decision feedback," in *Proc. IEEE Globecom*, vol. 6, (Sydney,NSW), pp. 3728–3733, Nov. 1998.
[14] M. Dangl, C. Sgraja, and J. Linder, "An improved block equalization scheme for uncertain channel estimation," *IEEE. Trans. Wireless Commun.*, 2006. To Appear.
[15] R. Otnes, *Improved Receivers for Digital High Frequency Communications: Iterative Channel Estimation, Equalization and Decoding (Adaptive Turbo Equalization)*. PhD thesis, Norwegian University of Science and Technology, 2002.
[16] C. Kuhn, "Iterative kanalschätzung, kanalentzerrung und decodierung für den EDGE-standard," Diplomarbeit, Technical University of Munich, Institute for Communications Engineering, May 2002.
[17] S. Talakoub and B. Shahrrava, "Turbo equalization with iterative online SNR estimation," in *Proc. IEEE WCNC*, (New Orleans, USA), pp. 1097–1102, Mar. 2005.
[18] M. Dangl and J. Linder, "Turbo equalization with parametric uncertainties: Comparison of snr estimation algorithms," in *Proc. 2nd COST 289 Workshop*, (Antalya, Turkey), pp. 129–133, July 2005.
[19] P. Laurent, "Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP)," *IEEE Trans. Commun.*, vol. 34, pp. 150–160, Feb. 1986.
[20] ETSI, *GSM-5.05, Digital Cellular Telecommunications system (Phase 2+):Radio transmission and reception*, 5.2.0 ed., July 1996.
[21] W. C. Jakes, *Microwave Mobile Communications*. John Wiley and Sons, 1974.
[22] C. Laot, R. L. Bidan, and D. Leroux, "Low-complexity MMSE turbo equalization: A possible solution for EDGE," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 965–974, May 2005.
[23] M. Anderson, M. C. Reed, and G. Borg, "A comparison of optimal and sub-optimal iterative equalization techniques for full-response CPM," in *4th Int. Symp. on Turbo Codes*, (Munich,Germany), April 2006.
[24] S. Kay, *Fundamentals of Statistical Signal Processing*, vol. 1. New Jersey, USA: Prentice Hall, 1993.
[25] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall, 3rd ed., 1996.
[26] S. Crozier, D. Falconer, and S. Mahmoud, "Least sum of squared errors (lsse) channel estimation," *IEE Proceedings, part F*, vol. 138, pp. 371–378, Aug 1991.
[27] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.