SYNCHRONISATION AND DC-OFFSET ESTIMATION FOR CHANNEL ESTIMATION USING DATA-DEPENDENT SUPERIMPOSED TRAINING

E. Alameda-Hernandez[1], D. C. McLernon[1], A. G. Orozco-Lugo[2], M. Lara[2] and M. Ghogho[1]

[1] School of Electronic and Electrical Engineering, University of Leeds, LS2 9JT, U.K.
Tel: +44 (0) 113 343 2050 Fax: +44 (0) 113 343 2032
[2] CINVESTAV-IPN, Sección de Comunicaciones, C.P.07360, México City, México
Tel: +52 (55) 5061 3768 Fax: +52 (55) 5747 7088
e-mails: {eenea,d.c.mclernon,m.ghogho}@leeds.ac.uk and {aorozco,mlara}@cinvestav.mx

ABSTRACT

The problem of channel estimation, under the data-dependent superimposed training (DDST) scheme, when no synchronisation between transmitter and receiver exists, is considered in this paper. The structure induced by the training sequence embedded in the transmitted signal is used to achieve synchronisation via projection operators, from which first, the value of the DC-offset and second, the channel coefficients can be extracted. The conditions for synchronisation, channel estimation and DC-offset estimation are derived. Note that this is the first synchronisation and DC-offset estimation method presented for channel estimation using the recently published DDST algorithm. Finally, simulations are presented that illustrate the successful practical application of our proposed method.

1. INTRODUCTION

In wireless communications, the channel estimation problem benefits from the inclusion of a training sequence, as opposed to the long data-record demanding blind-identification techniques. Traditionally, the training sequence and the data sequence were allocated in separate time slots, as in TDM, thus wasting bandwidth. An alternative method is the implicit/superimposed training approach (IT/ST) [1–3], where a periodic training sequence is actually added to the data prior to transmission, at the expense of a small data power loss.

The performance of the ST method can be enhanced using the recently published data-dependent ST (DDST) method [4], where the interference between data and training is removed. This is achieved by adding an additional data dependent sequence (to the original superimposed training sequence) at the transmitter. The effect of this data-dependent sequence is that in the DFT bins of the input signal where the training sequence has energy, the contribution from the data sequence is now effectively cancelled out, thus improving the channel estimates over ST.

In both ST and DDST, synchronisation between transmitter and receiver at training sequence level is required. Synchronisation for ST was first studied in [2] in conjunction with DC-offset estimation. The synchronisation was based on higher-order statistics (HOS) and polynomial rooting and only required that the training sequence period (P) is equal to the number of channel taps (M) —i.e. $P = M$. The use of HOS and polynomial rooting was avoided in the synchronisation method presented in [5], but as a consequence required $P \geq 2M+1$. These two training sequence synchronisation methods can be applied to DDST as well, but DDST additionally requires data block synchronisation to benefit from channel estimates free from data 'noise'.

In this paper, we concentrate on full synchronisation of the more appealing DDST method. The new training sequence synchronisation part has a much lower computational burden that the ones in [2,5]. An extension to the (necessary) block synchronisation is then included. Finally, simulations illustrate the excellent performance of this proposed method.

2. PROBLEM DESCRIPTION AND GEOMETRICAL INTERPRETATION

The received data block for the DDST scheme is [4],

$$x(k) = \sum_{l=0}^{M-1} h(l)b(k-l) + \sum_{l=0}^{M-1} h(l)c(k-l) + + \sum_{l=0}^{M-1} h(l)e(k-l) + n(k) + m \tag{1}$$

with $k = 0, 1, \dots, N-1$, and where $b(k)$ is the information bearing sequence, $h(k)$ is the channel impulse response of order $M-1$, i.e. $h(0) \neq 0$ and $h(M-1) \neq 0$, $n(k)$ is the noise and m represents an unknown DC-offset term due to using first-order statistics (see (2)) with non-ideal RF receivers (see [2]). Furthermore, $c(k)$ is the superimposed training sequence with mean $\bar{c}$ and power σ_c^2, and $e(k)$ corresponds to a data-dependent training sequence term. All terms can be complex valued. Both $c(k)$ and $e(k)$ are periodic with period $P \geq M$. Following [4], $e(k) = -\frac{1}{N_P}\sum_{i=0}^{N_P-1} b(iP+k)$, $k = 0, \dots, P-1$, with $N_P = \frac{N}{P}$. The channel coefficients, according to [2, 4], can then be obtained from the cyclostationary mean $y_0(j)$ of the output $x(k)$

$$y_0(j) := \mathrm{E}[x(iP+j)] = \sum_{l=0}^{M-1} h(l)c(j-l)_P + m \tag{2}$$

with $j = 0, \dots, P-1$, where $(\cdot)_P$ indicates arithmetic modulo-P, and the subscript '0' indicates that it is a fixed (deterministic) value as opposed to a general variable $y(j)$. This nomenclature will be used throughout the rest of the paper. Equation (2) can be written in matrix form as

$$\mathbf{y}_0 = \mathbf{C}_{[M]}\mathbf{h}_0 + \mathbf{m}_0 \tag{3}$$

E. Alameda-Hernandez is funded by Secretaría de Estado de Educación y Universidades of Spain and the European Social Fund.

where $\mathbf{C}_{[M]}$ is $P \times M$ and $\mathbf{h}_0 = [h(0),\ h(1),\ \dots,\ h(M-1)]^T$ is $M \times 1$; $\mathbf{y}_0 = [y_0(0),\ y_0(1),\ \dots,\ y_0(P-1)]^T$ and $\mathbf{m}_0 = [m,\ \dots,\ m]^{\mathrm{T}}$ are both $P \times 1$; matrix $\mathbf{C} = \mathrm{circ}(c(0),\ c(P-1),\ c(P-2),\ \dots,\ c(1))$, where 'circ' produces a circulant matrix [6]. Finally, we have adopted the nomenclature that for any matrix $\mathbf{A}$, then $\mathbf{A}_{[L]}$ and $\mathbf{A}_{\langle L\rangle}$ correspond respectively to the first and last L columns of $\mathbf{A}$. So matrix $\mathbf{C}$ is thus composed of $\mathbf{C}_{[M]}$ in (3) and its 'complement' $\mathbf{C}_{\langle P-M\rangle}$, where

$$\mathbf{C} \equiv \left[\mathbf{C}_{[M]} | \mathbf{C}_{\langle P-M\rangle}\right]. \tag{4}$$

To make the subspace interpretation that follows meaningful, we require $\mathbf{C}$ to be full rank. This can be accomplished by using optimum channel independent (OCI) training sequences [2] that have the extra advantage of satisfying $\mathbf{C}^{\mathrm{H}}\mathbf{C} = \mathbf{C}\mathbf{C}^{\mathrm{H}} = P\sigma_c^2 \mathbf{I}_{P\times P}$, thus simplifying the projection operation between subspaces. So, from (3), the channel coefficients are obtained by (note that $\mathbf{C}_{[M]}^{\mathrm{H}}\mathbf{C}_{[M]} = P\sigma_c^2\mathbf{I}_{M\times M}$)

$$\mathbf{h}_0 = \frac{1}{P\sigma_c^2}\mathbf{C}_{[M]}^{\mathrm{H}}\left(\mathbf{y}_0 - \mathbf{m}_0\right). \tag{5}$$

In the case that there is no synchronisation between transmitter and receiver (as regards the training sequence $c(k)$), once the cyclostationary means in (2) are computed, there is no way to say which of them corresponds to $j = 0,\ 1$, etc., because of the synchronisation offset. So the actual computed cyclostationary mean vector will be a cyclic permutation ($\mathbf{P}_0\mathbf{y}_0$) of the true one ($\mathbf{y}_0$), because of the arithmetic modulo-P operation in (2). It is important to note that a matrix $\mathbf{P}_0$ that performs a cyclic permutation operation on any vector is a circulant matrix.

For $P = M$, and when we do not have synchronisation at the receiver as regards $c(k)$, then $\mathbf{P}_0\mathbf{y}_0$ replaces $\mathbf{y}_0$ in (5) and note that $\mathbf{P}_0\mathbf{m}_0$ can always replace $\mathbf{m}_0$. So we obtain $\mathbf{P}_0\mathbf{h}_0$ (instead of the true $\mathbf{h}_0$). This is because circulant matrices ($\mathbf{P}_0$ and $\mathbf{C}_{[M=P]} = \mathbf{C}$) commute [6]. Thus, synchronisation reduces to finding the correct permutation matrix $\mathbf{P}_0$, as was done for example in [2]. For $P > M$, the matrices $\mathbf{C}_{[M]}$ and $\mathbf{P}_0$ do not commute, and so the solution to (3) is not as previously stated. For this case we need to pre-process $\mathbf{P}_0\mathbf{y}_0$ and select the correct solution among a set of candidates.

Important clues to develop a method for DDST synchronisation can be derived from the previous paragraphs. Recall that the method in [2] (for $P = M$) requires HOS and polynomial rooting, and so is very complex. On the other hand, the method in [5] ($P > M$) uses the FFT and is simpler that the former (both developed for ST). Furthermore, (5) obtains the channel vector just by projecting on the subspace spanned by the columns of $\mathbf{C}_{[M]}$ (recall that $\mathbf{C}$ is OCI). So we will set out to investigate the advantages of using an overdetermined system of equations ($P > M$) and try to interpret the problem as a projection process.

To begin, the following lemma illustrates what happens to the cyclostationary mean $\mathbf{y}_0$ after a cyclic permutation. But as we are interesting in extracting the properties induced by the training sequence, we examine $\mathbf{C}_{[M]}\mathbf{h}_0$ in (3) and note that $\mathbf{m}_0$ is not affected by any (cyclic) permutation.

Lemma 1 *For $\boldsymbol{C}$ full rank and $\boldsymbol{P}$ any cyclic permutation matrix, then $\boldsymbol{PC}_{[M]}\boldsymbol{h}$ can be uniquely decomposed as*

$$\begin{aligned}\boldsymbol{PC}_{[M]}\boldsymbol{h} =& \boldsymbol{C}_{[M]}(\boldsymbol{P}[\boldsymbol{h}^T\ \boldsymbol{0}_{P-M}]^T)_{[M]} + \\ &+ \boldsymbol{C}_{\langle P-M\rangle}(\boldsymbol{P}[\boldsymbol{h}^T\ \boldsymbol{0}_{P-M}]^T)_{\langle P-M\rangle}\end{aligned} \tag{6}$$

where $\boldsymbol{0}_{P-M}$ is a $1 \times (P-M)$ vector of zeros and for a vector $\boldsymbol{v}$, $\boldsymbol{v}_{[M]}$ ($\boldsymbol{v}_{\langle P-M\rangle}$) are its first M (last $P-M$) elements.

Proof: First note that $\mathbf{C}_{[M]}\mathbf{h} = \mathbf{C}[\mathbf{h}^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}} \Rightarrow \mathbf{PC}_{[M]}\mathbf{h} = \mathbf{PC}[\mathbf{h}^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}}$. Now, because circulant matrices commute [6], $\mathbf{PC}[\mathbf{h}^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}} = \mathbf{CP}[\mathbf{h}^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}}$, and (6) follows from (4). The uniqueness follows because $\mathbf{C}$ is full rank. Q.E.D

The interpretation of Lemma 1 is clear. Consider the vector space spanned by the columns of matrix $\mathbf{C}$, which are also a base for this space because $\mathbf{C}$ is full rank. In turn, $\mathbf{C}_{[M]}$ and $\mathbf{C}_{\langle P-M\rangle}$ span two subspaces V and $V^{\perp}$ respectively, that are orthogonal because $c(k)$ is OCI. Assume for the moment that $m = 0$ in (2). The true cyclostationary mean vector $\mathbf{y}_0$ lies exactly on V—i.e. it is a linear combination of the columns of $\mathbf{C}_{[M]}$—but any cyclic permutation $\mathbf{P} \neq \mathbf{I}$ of $\mathbf{y}_0$ will have components in $V^{\perp}$ as well. This important property can be used to achieve synchronisation in the DC-offset free case [7]. The next section develops the idea and proposes a general method to deal with synchronisation in the presence of a DC-offset.

3. PROPOSED ALGORITHM FOR DDST SYNCHRONISATION, ETC.

3.1 Training sequence synchronisation

Assume that because of lack of synchronisation what is available is a cyclic permutation of the cyclostationary mean vector —i.e. $\mathbf{P}_0\mathbf{y}_0$. Let us now consider its decomposition in V and $V^{\perp}$, and also consider a possible DC-offset. Now, we can decompose the DC-offset term $\mathbf{m}_0$ in (3) as

$$\mathbf{m}_0 = \mathbf{C}_{[M]}\tilde{\mathbf{m}}_{[M]} + \mathbf{C}_{\langle P-M\rangle}\tilde{\mathbf{m}}_{\langle P-M\rangle} \tag{7}$$

where $\tilde{\mathbf{m}}$ is a $P \times 1$ vector of constant elements $\frac{m}{P\bar{c}}$, as can be easily ascertained. So from (3), using (7) and Lemma 1,

$$\begin{aligned}\mathbf{P}_0\mathbf{y}_0 =& \mathbf{C}_{[M]}(\mathbf{P}_0[\mathbf{h}_0^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}})_{[M]} + \\ &+ \mathbf{C}_{\langle P-M\rangle}(\mathbf{P}_0[\mathbf{h}_0^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}})_{\langle P-M\rangle} + \\ &+ \mathbf{C}_{[M]}\tilde{\mathbf{m}}_{[M]} + \mathbf{C}_{\langle P-M\rangle}\tilde{\mathbf{m}}_{\langle P-M\rangle}.\end{aligned} \tag{8}$$

Consider now the projection of $\mathbf{P}_0\mathbf{y}_0$ onto the $V^{\perp}$ space. So multiply both sides of (8) by $\frac{1}{P\sigma_c^2}\mathbf{C}_{\langle P-M\rangle}^{\mathrm{H}}$:

$$\frac{1}{P\sigma_c^2}\mathbf{C}_{\langle P-M\rangle}^{\mathrm{H}}\mathbf{P}_0\mathbf{y}_0 = (\mathbf{P}_0[\mathbf{h}_0^{\mathrm{T}}\ \mathbf{0}_{P-M}]^{\mathrm{T}})_{\langle P-M\rangle} + \tilde{\mathbf{m}}_{\langle P-M\rangle}. \tag{9}$$

Now, two different cases are clearly distinguishable:

C1) For $\mathbf{P}_0 = \mathbf{I}$ the RHS of (9) reduces to $\tilde{\mathbf{m}}_{\langle P-M\rangle}$, which is a vector with all its components of equal value $\frac{m}{P\bar{c}}$;

C2) For $\mathbf{P}_0 \neq \mathbf{I}$ the first term of the RHS of (9) does not vanish in general, and thus, we will not have a vector of equal components.

The use of C1 and C2 for training sequence synchronisation is formalised in the next proposition, and we define the operator $\mathscr{J}\{\mathbf{v}\} = \|\mathbf{v} - \bar{\mathbf{v}}\|^2$, where $\bar{\mathbf{v}} = [\bar{v},\ \dots,\ \bar{v}]^{\mathrm{T}}$ and $\bar{v}$ is simply the mean of all the elements of $\mathbf{v}$. Note that $\mathscr{J}\{\mathbf{v}\} = 0$ iff all the elements of $\mathbf{v}$ are equal.

Proposition 1 *Let $P \geq 2M + 1$, hereafter known as the strong constraint, then $\mathscr{J}\left\{\boldsymbol{C}_{\langle P-M\rangle}^{H}\boldsymbol{P}_0\boldsymbol{y}_0\right\} = 0$ iff $\boldsymbol{P}_0 = \boldsymbol{I}$.*

Training sequence synchronisation: $\{\mathbf{P}_l\}_{l=1}^{P}$ =set of all cyclic permutation matrices of P elements.   Compute $\mathbf{P}_{\text{opt}} = \arg\min_{\mathbf{P}_l} \left\{ \mathscr{J}\left\{ \mathbf{C}^{\text{H}}_{\langle P-M\rangle} \mathbf{P}_l \hat{\mathbf{y}}_0^{(\mathbf{P}_0)} \right\} \right\}$
Block synchronisation   Let $\{\hat{\mathbf{y}}[l]\}_{l=0}^{N_P}$ be the set of estimates defined in section 3.2.   Compute $\hat{\mathbf{y}}_{\text{opt}} = \arg\min_{\hat{\mathbf{y}}[l]} \left\{ \mathscr{J}\left\{ \mathbf{C}^{\text{H}}_{\langle P-M\rangle} \hat{\mathbf{y}}[l] \right\} \right\}$
DC-offset estimation: From (10), $\hat{m} = \frac{\bar{c}}{\sigma_c^2} \frac{1}{P-M} \underbrace{[1,\ \dots\ ,\ 1]}_{P-M} \mathbf{C}^{\text{H}}_{\langle P-M\rangle} \hat{\mathbf{y}}_{\text{opt}}$
Channel estimation: From (5), $\hat{\mathbf{h}}_0 = \frac{1}{P\sigma_c^2} \mathbf{C}^{\text{H}}_{[M]} (\hat{\mathbf{y}}_{\text{opt}} - \underbrace{[\hat{m},\ \dots\ ,\ \hat{m}]}_{P})$.

Table 1: Proposed method for DDST channel estimation in the presence of a DC-offset when the transmitter and the receiver are not synchronised.

Proof: The proof uses the property of (9) (under case C1) and consists in finding the conditions under which the property under case C2 is always true for all $\mathbf{P}_0$, $\mathbf{y}_0$ and OCI $c(k)$. So let us work with the worst case scenario—i.e. when all the M components of $\mathbf{h}_0$ are equal. So, if we require $(\mathbf{P}_0[\mathbf{h}_0^{\text{T}}\ \mathbf{0}_{P-M}]^{\text{T}})_{\langle P-M\rangle}$ not to be a vector of equal components for any $\mathbf{P}_0 \neq \mathbf{I}$ and $\mathbf{h}_0 \neq \mathbf{0}_M^{\text{T}}$, then we require that its length is larger that M—i.e. $P-M > M$. Q.E.D

Training sequence synchronisation is achieved as follows. The cyclic permutation $\mathbf{P}\mathbf{P}_0\mathbf{y}_0$ of $\mathbf{P}_0\mathbf{y}_0$ minimising the operator $\mathscr{J}\left\{ \mathbf{C}^{\text{H}}_{\langle P-M\rangle} \mathbf{P}\mathbf{P}_0\mathbf{y}_0 \right\}$ is the true cyclostationary mean vector $\mathbf{y}_0$. This follows because by Proposition 1 $\mathbf{P}\mathbf{P}_0 = \mathbf{I}$, and thus $\mathbf{P}\mathbf{P}_0\mathbf{y}_0 = \mathbf{y}_0$.

3.2 Block synchronisation

That we have selected the correct permutation of $\mathbf{P}_0\mathbf{y}_0$ does not mean that in an actual application, under DDST, we have the best possible estimate for $\mathbf{y}_0$. For this, we do not only need to locate the start of a training sequence $\{c(k)\}_{k=0}^{P-1}$ period, but also the start of each received block $\{x(k)\}_{k=0}^{N-1}$. This is due to the fact that in (1) DDST transmits an extra training sequence $e(k)$, that is dependent upon the data sent during a block, so that when the cyclostationary mean is estimated, all dependency on the actual data is removed. If data from two different N-length blocks enters into the estimation of $\mathbf{y}_0$ this independency property is lost and so (unlike ST) we require block synchronisation. To introduce the block synchronisation method, remember that $\mathbf{y}_0$ has to be estimated using, as usual, time averages: $\hat{y}_0(j) = \frac{1}{N_P}\sum_{i=0}^{N_P-1} x(iP+j)$, $j = 0,\ 1,\ \dots\ ,\ P-1$. But because of lack of synchronisation this estimate will corresponds to an unknown permutation of $\mathbf{y}_0$, i.e. $\mathbf{P}_0\hat{\mathbf{y}}_0 := \hat{\mathbf{y}}_0^{(\mathbf{P}_0)}$.

$\mathbf{P}_{\text{opt}}$, as given in the training sequence synchronisation step of Table 1, tells us about the estimated indexes where the training sequence periods start. Assuming that the first element of $\mathbf{P}_{\text{opt}}\hat{\mathbf{y}}_0^{(\mathbf{P}_0)}$ was the d^{th} element of $\hat{\mathbf{y}}_0^{(\mathbf{P}_0)}$, then each sample of $x(k)$ that entered into the estimation of the d^{th} element of $\hat{\mathbf{y}}_0^{(\mathbf{P}_0)}$ marks the start of a training sequence period. Denote by q the smallest value of k (for $x(k)$) used in the estimation of this d^{th} element. Now compute $\hat{\mathbf{y}}[l] = \frac{1}{N_P}\sum_{i=0}^{N_P-1}[x(q+(i+l)P),\ x(q+(i+l)P+1),\ \dots\ ,\ x(q+(i+l)P+P-1)]^{\text{T}}$, $0 \leq l \leq N_P$. Note that $\hat{\mathbf{y}}[0] = \mathbf{P}_{\text{opt}}\hat{\mathbf{y}}_0^{(\mathbf{P}_0)}$, and for $\hat{\mathbf{y}}[1]$ the N-point window has moved P samples to the right and so on. Only the vector $\hat{\mathbf{y}}_{\text{opt}}$ (see Table 1) encompassing a full DDST block will provide a cyclostationary mean vector independent of the data sequence—i.e. with a reduced 'data' noise compared with the rest of the $\hat{\mathbf{y}}[l]$ estimates—and so it will minimise the cost function, $\hat{\mathbf{y}}_{\text{opt}} = \arg\min_{\hat{\mathbf{y}}[l]} \left\{ \mathscr{J}\left\{ \mathbf{C}^{\text{H}}_{\langle P-M\rangle} \hat{\mathbf{y}}[l] \right\} \right\}$, as in Table 1.

3.3 DC-offset and channel estimation

Once the best possible estimate for $\mathbf{y}_0$ is known, the DC-offset m can be computed using the property of (9) under case C1,

$$m = \frac{\bar{c}}{\sigma_c^2} \frac{1}{P-M} \underbrace{[1,\ \dots\ ,\ 1]}_{P-M} \mathbf{C}^{\text{H}}_{\langle P-M\rangle} \mathbf{y}_0. \tag{10}$$

Thus, m is a normalised mean of the elements of $\mathbf{C}^{\text{H}}_{\langle P-M\rangle}\mathbf{y}_0$, where the normalisation factor is the quotient of the mean ($\bar{c}$) and the power (σ_c^2) of the training sequence. The implementation of this part is shown in Table 1.

Finally, after synchronisation and DC-offset estimation, the channel can then be estimated as also indicated in Table 1.

3.4 Important remark

The *strong* constraint (in Proposition 1) assumes that the channel order $M-1$ is known. If the channel order is over-estimated, $\mathbf{h}_0$ will have extra zero taps at the tails. This will allow operator $\mathscr{J}$ to have more than one possible solution following Proposition 1. But all the provided solutions are related by a linear shift and the only effect on equalisation is a delay of $z^{-\Delta}$, for integer Δ. The same can happen in the practical implementation if the elements in the tails of the channel vector are finite but small, even if M is known.

4. SIMULATION

Three-tap Rayleigh fading channels were simulated. The channel coefficients were complex Gaussian, i.i.d. with unit variance. The average energy of the channel was set to unity. The data was a BPSK sequence, to which an OCI training sequence (see [2]) was added before transmission. As regards (1), the training to information power ratio (TIR$=\frac{\sigma_c^2}{\sigma_{(b+e)}^2}$) was set to -6.9798 dB, $P=7$ and $N=399$ samples. These are the same values as used in [5]. In each block a cyclic prefix was added to ensure full independency of the DDST $\mathbf{y}_0$ estimates with respect to the information sequence $b(k)$. We generated $N_B = 300$ blocks at the transmitter, and while only N samples were used for channel estimation, all the blocks were used for BER computation. A deterministic DC-offset (m) was added at the channel output, together with zero-mean white Gaussian noise. The value of the DC-offset was determined by the DC-offset to signal AC-component (DCAC) power ratio $m^2/\text{E}\left[|x(k)-n(k)-m|^2\right]$ as defined in [1]. In these simulations this was set to DCAC$=0.1$. At each realisation, a random synchronisation offset between 0 and $N+P-1$ was introduced between transmitter and receiver, so we could be

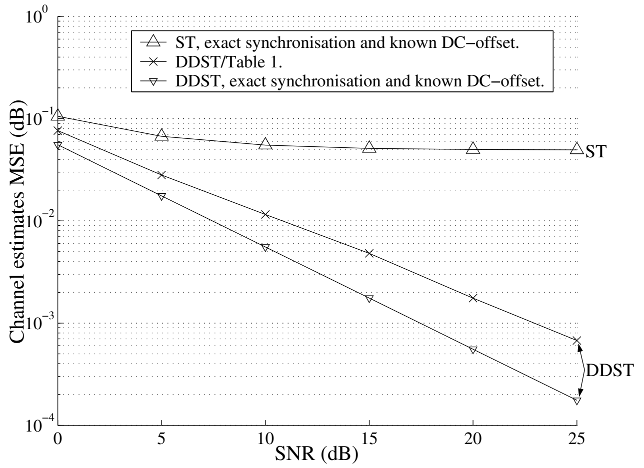

Figure 1: MSE of channel estimates, as a function of the SNR, computed following Table 1. The identification delay (Δ) has also been considered. The channel estimates assuming known DC-offset and perfect synchronisation (for both ST and DDST) are included for comparison purposes.

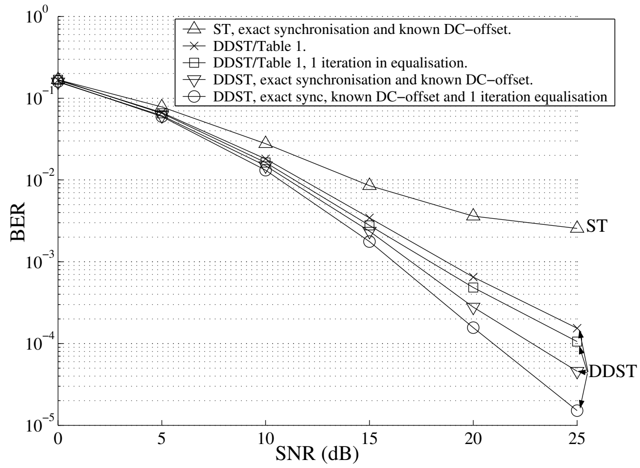

Figure 2: BER versus SNR obtained using the channel estimates from Table 1 to construct a MMSE equaliser.

at *any* sample index within the first *block*. After channel estimation, an MMSE equaliser (based on the channel estimates) of length 11 and optimum delay was used to compute the BER. 1000 realisations were averaged. Now, as already mentioned at the end of section 3.3, the estimated channel may incur a $z^{-\Delta}$ delay element, especially when the channel coefficients at the tails are very small. This *identification delay*, which in practise has no major consequences, can worsen the simulated MSE and BER, distorting the performance analysis of the method. To avoid this, the identification delay was computed comparing the equalised symbols with the true ones affected by different delays to compute the BER. Then, the delay giving the smallest BER is chosen as the identification delay. The problem of the identification delay was reported in [5] as well.

The MSE of the channel estimates obtained with the proposed method is presented in figure 1. The MSE obtained from the ST and DDST methods assuming perfect synchronisation and known DC-offset are also included for comparison. As shown in [4], DDST delivers much better channel estimates than ST. From figure 1, we can say that this is even true if synchronisation and DC-offset for DDST are unknown, and have to be estimated via the proposed method. Note from figure 1 that DDST with estimated synchronisation is closer to DDST with perfect synchronisation than it is to ST with perfect synchronisation.

Similar conclusions can be drawn from the BER graph in figure 2. It is important to notice that the BER can be reduced if the periodic sequence $e(k)$ in (1), inherent to DDST, is now estimated and removed as reported in [4]. To do this, the output $x(k)$ is equalised to give $b_{\text{eq}}(k)$, and then a hard decision is carried out to give $b_{\text{hd}}^{(0)}(k)$. Define $e^{(0)}(j) = -\frac{1}{N_P}\sum_{i=0}^{N_P-1} b_{\text{hd}}^{(0)}(iP+j)$, for $j = 0, \ldots, P-1$. Subtract this from $b_{\text{eq}}(k)$ and then follow with a hard decision to yield $b_{\text{hd}}^{(1)}(k)$. This is called an *iteration in the equalisation* (see figure 2). Note that $b_{\text{eq}}(k)$ is unchanged throughout this process.

5. CONCLUSIONS

A method for synchronisation and DC-offset estimation for channel estimation has been presented under the new DDST scheme, and this is the first such proposal. To exploit the full potential of the DDST method, both training sequence and block synchronisation are required. The presented method is based on projection operators. The conditions for synchronisation, DC-offset removal and channel estimation are presented. The simulations show that the performance, in terms of MSE and BER, of the overall synchronisation method, is similar to that of the DDST method, when we know exactly the DC-offset and have perfect synchronisation.

REFERENCES

[1] J. K. Tugnait and W. Luo, “On channel estimation using superimposed training and first-order statistics,” *IEEE Commun. Lett.*, vol. 7, pp. 413–415, 2003.

[2] A. G. Orozco-Lugo, M. M. Lara, and D. C. McLernon, “Channel estimation using implicit training,” *IEEE Trans. Signal Processing*, vol. 52, pp. 240–254, 2004.

[3] M. Ghogho and A. Swami, “Improved channel estimation using superimposed training,” in *Fifth IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2004*, 11-14 July, 2004, Lisbon, Portugal.

[4] M. Ghogho, D. C. McLernon, E. Alameda-Hernandez, and A. Swami, “Channel estimation and symbol detection for block transmission using data-dependent superimposed training,” *IEEE Signal Processing Lett.*, vol. 12, pp. 226–229, 2005.

[5] J. K. Tugnait and X. Meng, “Synchronization of superimposed training for channel estimation,” *Proceedings of International Conference on Acoustics, Speech and Signal Processing ICASSP04*, vol. IV, pp. 853–856, 2004.

[6] P. J. Davis, *Circulant Matrices*. New York, N.Y.: Chelsea Publishing, 1994.

[7] E. Alameda-Hernandez, D. C. McLernon, A. G. Orozco-Lugo, M. Lara, and M. Ghogho, “Synchronisation for superimposed training based channel estimation,” to appear in *Electronics Letters*, 2005.