LOW COMPLEXITY LINEAR DETECTION FOR ASYNCHRONOUS CDMA SIGNALS OVER FREQUENCY SELECTIVE CHANNELS

Armelle Wautier\textsuperscript{1}, Saloua Ammari\textsuperscript{2}, Walid Hachem\textsuperscript{1}

\textsuperscript{1}École Supérieure d'Électricité, 3 rue Joliot-Curie, plateau de Moulon, 91192 Gif sur Yvette cedex, France\email{armelle.wautier@supelec.fr, walid.hachem@supelec.fr}

\textsuperscript{2}Bouygues Telecom, 10 rue Paul Dautier, 78944 Vélizy Cedex, France\email{sammari@bouyguestelecom.fr}

ABSTRACT
In this paper, a suboptimal low complexity structure for joint linear detection of CDMA signals is proposed. The optimal joint linear zero-forcing or minimum mean squared error detectors use a vector model of the transmission where the channel is represented by a matrix. Due to the finite memory of the channel, most of the blocks of the channel autocorrelation matrix are equal to zero. It is then possible to put this structure into profit to derive low complexity implementations for these detectors. In this paper, the optimal linear system resolution is approximated by overlapped smaller subsystems. The performance and the complexity of the resulting detector are analyzed for both downlink and uplink communications over frequency selective channels for periodic spreading codes with a period equal to the symbol period as well as for random-like codes. The trade-off between complexity and performance is flexible, which offers a great advantage to the proposed solution.

1. INTRODUCTION
Code division multiple access communications suffer from both multiple access interference and inter symbol interference. The optimal multi-user receiver has a complexity which grows exponentially with the channel memory $P$ (measured in modulation symbol periods) and with the number of active spreading codes $K$. Therefore, sub-optimal receivers such as block linear receivers, interference cancellers or RAKE receivers have been proposed to handle the complexity issue.

The block linear multi-user receivers with MMSE (Minimum Mean Squared Error) or ZF (Zero Forcing) criteria give good performances \cite{1}. They make decisions on blocks corresponding to $N$ modulated symbols for $K$ spreading codes. The size of the blocks is usually linked to the size of the bursts of the transmission format, which can be very large. The optimal joint linear ZF or MMSE detectors use a vector model of the transmission where the channel is represented by a matrix. The optimal solution consists in solving a linear system of size $KN$. Several algorithms have been proposed to minimize the complexity of the system resolution by exploiting the property of the channel autocorrelation matrix. Efficient reduction of the complexity is obtained for synchronous downlink transmission where the spreading codes are periodic with period equal to the modulation symbol period \cite{2}. Due to the finite memory of the channel, most of the blocks of the channel autocorrelation matrix are equal to zero. Then, the truncation of the block processing window has been suggested in \cite{3}. Finite-memory MMSE linear equalizer has been proposed in \cite{4}, performance has been evaluated for asynchronous transmission over non selective propagation channels.

In this paper, a low complexity implementation for finite-memory detectors is proposed as follows: the resolution of the whole system is replaced by the resolution of $N$ overlapped smaller subsystems of size $nK$ with $n$ lower than $2P+1$. The algorithm is valid for periodic or non periodic spreading sequences, for both downlink and uplink communications over frequency selective channels. The paper is organized as follows. The CDMA transmission model and the block linear receiver are defined in section 2. The proposed algorithm is described in section 3. Complexity issues are discussed in section 4. Performance analysis is given in section 5. Numerical results are given for the TDD (Time Division Duplex) mode of UMTS (Universal Mobile Telecommunication System).

2. CDMA TRANSMISSION MODEL

In a CDMA cellular system, the spreading codes result from the product of a channelization code and a scrambling code. In the TDD mode of UMTS, for example, the spreading codes are periodic and they are furthermore orthogonal \cite{5}. In the FDD (Frequency Division Duplex) mode, the spreading codes have long period, i.e. the signature varies from one symbol to another. Even for orthogonal codes, because of multi-path propagation and of time misalignment between different mobile stations (in the uplink), signals lose their orthogonal property at the reception. This loss of orthogonality depends on the correlation properties of the spreading waveforms and on the propagation conditions as well.

2.1 Transmission model
The received CDMA signal is given by:
where $N$ is the number of the modulation symbols in the burst, $K$ is the number of active codes (or users), $u_{k,i}^{(n)}$ is the $i^{th}$ data symbol of user $k$, $p_{i}(t)$ is the spreading waveform of user $k$ during symbol $i$. $M_i$ is the number of coefficients in the channel impulse response model, $c_{k,i}^{(l)}$ and $\tau_{k,i}^{(l)}$ are respectively the $n^{th}$ complex coefficient and delay of the channel for user $k$. $\beta^{(l)}$ includes power amplification, path-loss and shadowing attenuations. The channel is assumed to be invariant during the burst. $v(t)$ is the white additive Gaussian noise with one side Power Spectral Density equal to $N_0$. The spreading waveform has the following expression:

$$p_{i}(t) = \sum_{n=0}^{Q} v_{Q,n}^{(l)} g(t - n T_c)$$

where $Q$ is the spreading factor, $v_{Q,n}^{(l)}$ is the $n^{th}$ chip of the spreading code during symbol $i$ of user $k$, $g(t)$ is chip waveform, which is usually a square root raised cosine filter.

The received signal can also be written as:

$$r(t) = \sum_{k=1}^{K} \sum_{i=1}^{N} u_{k,i}^{(l)} h_{i}(t - iT) + v(t)$$

$h_{i}(t)$ is the global channel response including the signature and the channel impulse response for user $k$ during symbol $i$.

### 2.2 Equivalent discrete-time channel model

At the receiver side, the signal is filtered and sampled at the rate $1/T_c = J/T_c$ (with $J \geq 2$).

$$z(t) = \sum_{k=1}^{K} \sum_{i=1}^{N} u_{k,i}^{(l)} h_{i}(t - iT_c) + w(t)$$

Let $Z$ be the vector of the received signal for the entire block of $N$ symbols, its length is $N_c = JN + L$, where $L$ is the maximum length of the sampled global responses (measured in $T_c/J$ intervals). Using a matrix-vector notation, it can be written as:

$$Z = AU + W$$

$W$ is the sampled filtered noise vector. $U$ is the vector of the data symbols ordered as follows:

$$U = [u_{1}^{(1)} \ldots u_{1}^{(K)} \ldots u_{2}^{(1)} \ldots u_{2}^{(K)} \ldots \ldots u_{N}^{(1)} \ldots u_{N}^{(K)}]^{t}$$

The matrix $A$ is built of $N$ sub-blocks $H_i$. Each column vector of $H_i$ represents the sampled global channel impulse response of user $k$ as shown below:

$$A = \begin{bmatrix}
    h_{0}^{i}[0] & \cdots & h_{0}^{i}[K-1] & \cdots & h_{Q}^{i}[0] & \cdots & h_{Q}^{i}[K-1] \\
    \vdots & & \vdots & & \vdots & & \vdots \\
    h_{0}^{i}[1] & \cdots & h_{0}^{i}[K] & \cdots & h_{Q}^{i}[1] & \cdots & h_{Q}^{i}[K] \\
    \vdots & & \vdots & & \vdots & & \vdots \\
    \vdots & & \vdots & & \vdots & & \vdots \\
\end{bmatrix}$$

2.3 Optimal joint block linear equalizer

The optimal joint block linear equalizer (BLE) makes decisions on the whole data vector $Z$ through a linear transformation given by: $U' = AZ$ where $A$ is a matrix of size $NK \times N$, utilising the ZF or MMSE criterion applied to $U' - U$. These criteria lead a solution given by:

$$U' = M^{-1}Y, \quad where \quad Y = A^H Z$$

The matrix $M$ equals $A^HA + \sigma^2 I$ for the MMSE criterion and equals $A^HA$ for the ZF criterion. The matrix $M$ is an autocorrelation matrix, having hermitian symmetry and block structure. The Cholesky algorithm can then be used to solve the system $MU' = Y$. Due to the finite memory of the autocorrelation of the channel $A^HA$, most of the blocks the matrix $M$ are equal to zero. For the particular case of periodic codes with period equal to the symbol period, the matrix is furthermore block Toeplitz with $2P+1$ block $B_0$ of size $KxK$ (cf. figure 1). Reduced complexity algorithms have been proposed in that case in [2] to solve the whole system.

![Figure 1: Block Toeplitz structure of the matrix $M$ for symbol-period codes](image)

3. PROPOSED SUB-OPTIMAL LINEAR EQUALIZER

Due to the finite memory of the autocorrelation of the channel $A^HA$, most of the blocks are null. This structure is put into profit to derive low complexity implementations for the detector. The proposed sub-optimal detector consists in considering smaller block matrices of size $nK$ with $n$ equal to $2P+1$ (cf. figure 1). For complexity reasons, the size can be educed to a lower value $n = 2D+1$.

Let $M_i$ and $Y_i$ denote the portion of matrix $M$ and vector $Y$ corresponding to the subsystem involving the finite-memory vector $U'_{i} = [u_{0}^{(1)} \ldots u_{i}^{(K)} \ldots u_{i}^{(1)} \ldots u_{i}^{(K)} \ldots u_{i}^{(1)} \ldots u_{i}^{(K)}]^{T}$ with $n = 2D+1$.

$$U'_{i} = (M_i)^{-1} Y_i$$

(1)
In order to avoid border effects, the decisions are made only on the $i^{th}$ centred data symbols of the $K$ users $U_{i,t} = [u_{i}^{(1)}...u_{i}^{(D)}]$, which leads to the truncated solution:

$$U_{i,t} = [M_{i}^{-1}], Y_{i}$$

([M_{i}^{-1}]) denote the $K$ central lines of the matrix $M_{i}^{-1}$. Therefore, there is an overlapping between subsystems, and the size of the truncated matrix is equal to $K$ for transmission with periodic codes with period equal to $N$ and then to compute the matrix inversion. In that particular case, it becomes interesting to invert the matrix once per burst and then to compute the decision vectors. This matrix inversion is more consuming than the Cholesky algorithm. The Cholesky algorithm having a computation complexity order given by $O(n^3)$ [6]. Then, the computation of decision symbols generates $K^2N$ multiplications and $K^2(n-1)$ additions.

4. COMPLEXITY EVALUATION

Assuming that the channel impulse responses are estimated with the help of training sequences, the blocks of the matrix $A^2A$ can be computed.

Then the size of subsystems $n$ is chosen so as to obtain the best trade-off between complexity and performance. Typically, $n$ equals $(2P+1)$. However, $n$ can be chosen lower than $(2P+1)$ if required by complexity constraints.

Note that for most channels of practical interest for the TDD mode of UMTS, $P$ equals 1 with a spreading factor of 16, except for the vehicular B channel model, where $P$ equals 5.

4.1 Subsystem resolution with block–Levinson algorithm for periodic codes

The complexity of the proposed receiver can be very low for periodic codes with period equal to the symbol period. This property is realized in practice for the TDD mode of UMTS in the downlink with spreading factor equal to 16. The matrices $M_i$ are all identical, and they are moreover block Toeplitz. In that particular case, it becomes interesting to invert the matrix once per burst and then to compute the $N$ sample vectors $U_{i,t}$ according to equation (2). This solution is computationally efficient because the computation of the matrix inversion $M_i^{-1}$ can be done by means of the block–Levinson algorithm having a computation complexity order given by $O(n^2K^3)$ [6]. Then, the computation of decision symbols generates $K^2N$ multiplications and $K^2(n-1)$ additions.

4.2 Subsystem resolution with Cholesky algorithm

When the matrix is not block Toeplitz, the inversion of the matrix is more consuming than the Cholesky algorithm. The algorithm is two steps: decomposition of the matrix $M_i$ into a product $LL_h$ where $L$ is a low triangular matrix and iterative resolution of two successive systems $LX=Y_i$ and $L_h^{-1}U_{i,t}=X_i$.

The first step generates a computational complexity which is proportional to $nK(nK+1)(nK-1)/6 \approx n^3K^3/6$. For a periodic code with period equal to $mQ$, this decomposition is made $mQ$ times per burst. For random-like codes, it is realized $N$ times since the equivalent channel is varying from one symbol to another.

The second step leads to $(N-2D)(DK+1)(3DK+2)$ operations for $n$ equal to $2D+1$.

4.3 Numerical results

Some numerical results are here given to illustrate complexity issues of the proposed algorithm.

A first example is given for periodic codes: $Q = 16$, $m = 1$, $K = 8$, $N = 61$.

For small channel memory ($P = 1$) and therefore for small subsystems ($n = 3$) both proposed algorithms are equivalent in terms of complexity. The algorithm described in 4.1 leads to $408 + 11712 = 16120$ computations (MAC, multiplications, additions and comparisons). The algorithm described in 4.2 gives $2300 + 13806 = 16120$ computations.

For long channel memory, the first proposed algorithm becomes much more efficient than the second one. For $P$ equal to 4 and $n$ equal to 9, the complexity of the first algorithm leads to $41472 + 35136 = 76608$ computations instead of $62196 + 172402 = 233598$ for the second one.

A second numerical example is given for varying spreading factors between 4 and 16, the codes are periodic with a period equal to 16 chip periods (like in the uplink of TDD UMTS mode), and for different channel memory lengths given in chip periods $pT_c$. Table I gives the complexity for a constant throughput given by $NK = 488$ data symbols per burst. It is worthwhile to note that computation limitation may influence the choice of the spreading factor according to the channel length (cf. Table I).

5. PERFORMANCE ANALYSIS

5.1 Performance evaluation

Decisions are made on the sample vectors $U_{i,t}$, which have the following expressions:

$$U_{i,t} = [M_{i}^{-1}]A_h^b AU + [M_{i}^{-1}]A_h^b W$$

We can easily derive the signal to noise ratio at the output of the receiver on the decision samples, which can be expressed by:

$$SNR_{i,k} = \frac{1}{\lambda_{i,k}} \left( \frac{P_{i,k}}{N_0 + \sigma^2_{i,k}} \right)$$

Table I. Complexity evaluation of the 4.2 algorithm for different channel lengths and spreading factors with a fixed throughput.

<table>
<thead>
<tr>
<th>$Q$</th>
<th>$K$</th>
<th>$N$</th>
<th>$P$</th>
<th>$n$</th>
<th>1st step</th>
<th>2nd step</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>8</td>
<td>61</td>
<td>57</td>
<td>4</td>
<td>9</td>
<td>62196</td>
<td>171402</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>122</td>
<td>57</td>
<td>8</td>
<td>17</td>
<td>104788</td>
<td>342804</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>244</td>
<td>57</td>
<td>15</td>
<td>31</td>
<td>158844</td>
<td>610328</td>
</tr>
</tbody>
</table>

Some numerical results are here given to illustrate complexity issues of the proposed algorithm.

A first example is given for periodic codes: $Q = 16$, $m = 1$, $K = 8$, $N = 61$.

For small channel memory ($P = 1$) and therefore for small subsystems ($n = 3$) both proposed algorithms are equivalent in terms of complexity. The algorithm described in 4.1 leads to $408 + 11712 = 16120$ computations (MAC, multiplications, additions and comparisons). The algorithm described in 4.2 gives $2300 + 13806 = 16120$ computations.

For long channel memory, the first proposed algorithm becomes much more efficient than the second one. For $P$ equal to 4 and $n$ equal to 9, the complexity of the first algorithm leads to $41472 + 35136 = 76608$ computations instead of $62196 + 172402 = 233598$ for the second one.

A second numerical example is given for varying spreading factors between 4 and 16, the codes are periodic with a period equal to 16 chip periods (like in the uplink of TDD UMTS mode), and for different channel memory lengths given in chip periods $pT_c$. Table I gives the complexity for a constant throughput given by $NK = 488$ data symbols per burst. It is worthwhile to note that computation limitation may influence the choice of the spreading factor according to the channel length (cf. Table I).

Table I. Complexity evaluation of the 4.2 algorithm for different channel lengths and spreading factors with a fixed throughput.
5.2 Performance results

In this subsection, we present some performance results. The evaluation has been carried out with UMTS TDD transmission parameters [5]: the chip rate is equal to 3,84 Mchip/s and the roll-off of the chip waveform is equal to 0.22, the spreading factor is 16.

When $n$ is taken equal to $2^p+1$, the performance of the proposed algorithm is very close to the performance of the ideal block linear detector for any number of users and for any channel profile.

In order to manage complexity issues, the size of subsystems can be reduced to a fixed arbitrary value $n$ lower than $2^p+1$. Figures 2 and 3 evaluate the degradation due to truncation when the channel memory is long ($p = 5$). It is worthwhile to note that the degradation increases with the number of codes when $n$ is smaller than $2^p+1$.

Figure 2 shows the performance of the proposed algorithm with the lowest value for $n$ with 8 users. The obtained performance is compared to the ideal joint detector performance and to the RAKE receiver performance. The channel impulse response is constant with amplitude in dB equal to $-2.5, 0, -12.8, -10, -25.2$ and delays expressed in nanoseconds given by $0, 300, 8900, 12900, 17100$.

Figure 3 evaluates the effect of truncation over a random channel for the downlink ($p = 5$). It gives the mean bit error rate (averaged over users, symbols and channel responses) as a function of the mean energy per bit $E_b$ over $N_0$. This shows the trade-off between complexity and performance.

6. CONCLUSION

A low complexity joint detector has been proposed and performance analysis shows that it can be a good candidate for CDMA communications using either short periodic codes or long codes. Its complexity can be constrained for long memory channels, and the performance degradation is not severe, which gives an alternate solution between the ideal block linear detector and the rake receiver.

ACKNOWLEDGMENT

This research work was carried out in the framework of the French RNRT project PETRUS in partnership with Mitsubishi Electric ITE, Bouygues Telecom, Supelec, CEA-LETI and ENST-Bretagne.

REFERENCES