SIGNAL PROCESSING FOR TRANSMIT-REFERENCE UWB

Hieu Q. Dang and Alle-Jan van der Veen

TU Delft, Fac. EEMCS, Mekelweg 4, 2628 CD Delft, The Netherlands

Transmit-reference (TR) is known as a realistic but low data rate candidate for ultra-wideband (UWB) communication systems. Higher data rates are possible but give rise to interpulse, interframe, and intersymbol interference, and thus require some form of equalization. In this paper, we discuss a suitable receiver algorithm and its complexity, and derive the associated maximal data rate.

1. INTRODUCTION

Since 2002, ultra-wideband (UWB) has received special research interest as a promising technology for high speed, high precision, strong penetration short-range wireless communication applications. Impulse radio (IR UWB) schemes have been proposed but present significant challenges: it is not possible to sample and process at Nyquist rate (several GHz), and dense multipath requires RAKE receivers with more taps than feasible. Moreover, the ultra-short pulses with duration of only a fraction of a nanosecond requires strict timing synchronization [3].

The transmit-reference (TR) scheme first proposed for UWB in [4, 5] emerges as a realistic candidate that can effectively deal with these challenges. By transmitting pulses in pairs (or doublets) in which both pulses are distorted by the same channel, and using an autocorrelation receiver, the total energy of the channel is gathered to detect the signal without having to estimate individual channel multipath components. The simple delay (at the transmitter), correlation and integration operations (at the receiver) ease the timing synchronization requirements [6] and reduce the transceiver's complexity.

Most TR-UWB schemes for simplicity assume that the pulse spacing D in a doublet is longer than the channel length T_h to prevent inter-pulse interference (IPI). Also the frame period T_f is often chosen such that there is no inter-frame interference (IFI). Together this leads to $T_f > 3T_h$, resulting in a low data rate (T_h can be 50–200 ns). Also, wideband delay lines longer than a few pulse widths are difficult to implement with high accuracy [7]. Therefore, in [8], we have considered a TR-UWB scheme

*This research was supported in part by NWO-STW under the VICI programme (DTC.5893). Parts of this paper were also presented at conferences [1, 2]. A full version was submitted to IEEE J. Special Topics in Signal Proc., Dec. 2006.

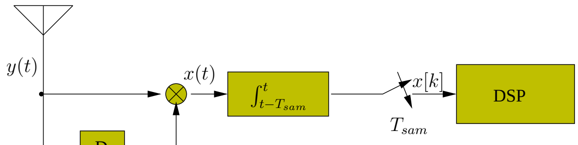

Figure 1: Autocorrelation receiver

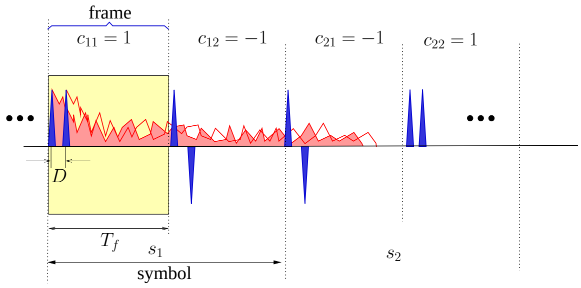

Figure 2: Pulse sequence structure

where the pulse spacing D is very short, much shorter than the channel length T_h. In [1], we further considered $T_f < T_h$, and introduced equalization schemes to remove the IPI and IFI. As a result, the frame rate can be at least three times higher than in the preceding schemes. In [2] we have extended this scheme to a CDMA-like multiuser context.

In this paper, we present the receiver algorithms for this system, derive the complexity, and analyze the constraints on the design parameters in relation to practical system design.

2. DATA MODEL

2.1. Single frame

In TR-UWB systems, pulses are transmitted in pairs (doublets), one doublet per frame. Within a frame, the first pulse is fixed, while the second pulse, delayed by D seconds, has information in its polarity: $s_0 \in \{-1, +1\}$. The received signal at the antenna output due to one transmit-

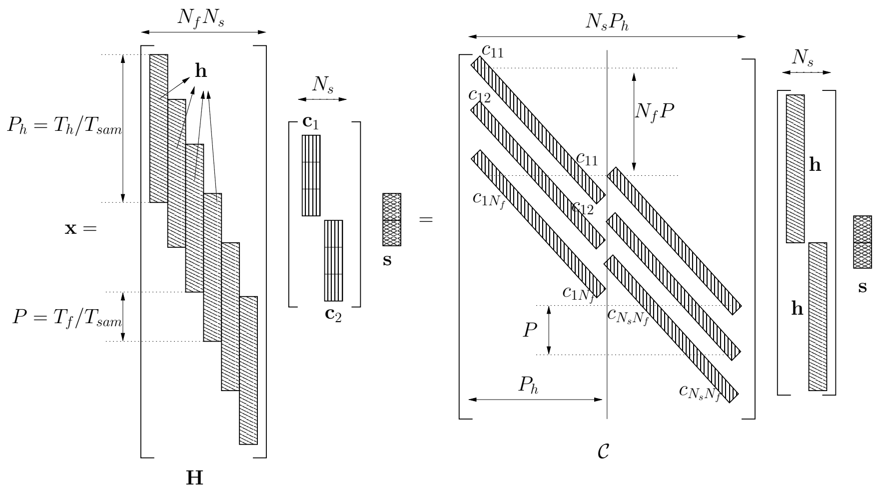

Figure 3: The data model for the single user, single delay case (time offset not shown)

ted frame is

$$y_0(t) = h(t) + s_0 \cdot h(t - D)\,,$$

where $h(t) = h_p(t) * g(t) * a(t)$ is the equivalent channel: the physical channel convolved with pulse shape and antenna response.

The receiver structure for a single frame is shown in Fig. 1, in which $y_0(t)$ is multiplied with a delayed (by D) version of itself before being integrated and dumped. The sampling period is T_{sam}, and we use oversampling by taking P samples per frame: $T_{sam} = \frac{T_f}{P}$. The resulting signal at the multiplier's output is

$$\begin{aligned} x_0(t) &:= y_0(t)y_0(t-D) \\ &= [h(t) + s_0 h(t-D)][h(t-D) + s_0 h(t-2D)] \\ &= [h(t)h(t-D) + h(t-D)h(t-2D)] \\ &\quad + s_0[h^2(t-D) + h(t)h(t-2D)]\,. \end{aligned}$$

Define the channel autocorrelation function as

$$R(\tau, n) = \int_{(n-1)T_{sam}}^{nT_{sam}} h(t)h(t-\tau)dt\,.$$

After integrate-and-dump, the received samples are

$$\begin{aligned} x_0[n] &= [R(0, n - \tfrac{D}{T_{sam}}) + R(2D, n)]s \\ &\quad + [R(D, n) + R(D, n - \tfrac{D}{T_{sam}})]\,. \end{aligned} \qquad (1)$$

The dominant term is the matched term, $R(0)$, which contains the energy of the channel segments. The unmatched terms $R(\tau)$ with $\tau \in \{D, 2D\}$ can be ignored if we choose $D > \tau_0$, where τ_0 is a certain correlation length, often very small (less than a nanosecond) for typical UWB channels, and dependent on channel statistics and antenna responses.

The oversampling process (by integrate and dump with $T_{sam} < T_f \ll T_h$) divides the spreading channel into $L_h = \lfloor \frac{T_h}{T_{sam}} \rfloor$ segments (or sub-channels). Each segment has its own "channel energy" and "channel autocorrelation function". The original channel $h(t)$ is now replaced by L_h parameters related to the energy of the channel segments:

$$h[n] = \int_{(n-1)T_{sam}}^{nT_{sam}} h^2(t)dt \qquad n = 1, \cdots, L_h\,. \qquad (2)$$

Define the corresponding TR-UWB "channel" vector as

$$\mathbf{h} = [h[1], \cdots, h[L_h]]^T\,. \qquad (3)$$

After stacking all discrete samples together in a vector $\mathbf{x}_0$ and ignoring the cross-terms in (1), we have a generic data model for a single frame as

$$\mathbf{x}_0 = \mathbf{h} \cdot s_0 + \text{noise}\,. \qquad (4)$$

2.2. Multiple frames

We extend the preceding model to the transmission of N_f consecutive frames. Each frame has duration T_f, and is assigned a data bit s_j in the polarity of its second pulse. The frame period T_f is much shorter than the channel length T_h. At the receiver, new crossterms due to inter-frame interference occur. As before, crossterms with unmatched delays can be ignored in the model and will be treated as a noise-like signal in the receiver algorithm.

However, we still have the matched term that spreads over some next frames because $T_h \gg T_f$. These overlapping parts are IFIs and can be modeled in a channel matrix $\mathbf{H}$ in the data model for multiple frames as

$$\mathbf{x} = \mathbf{H}\mathbf{s} + \text{noise} \qquad (5)$$

where $\mathbf{x}$ is the stacking of all received samples, $\mathbf{s}$ is the unknown data vector $\mathbf{s} = [s_1 \cdots s_{N_f}]^T$, and $\mathbf{H}$ is the channel matrix that contains shifted versions of the "channel" vector $\mathbf{h}$ in (3). The relation is illustrated in Fig. 3 (left part). The IFI effect is also visible in this figure from the fact that many rows in $\mathbf{H}$ have more than one nonzero entry.

2.3. Single user, single delay

The preceding preliminary models are extended to the reception of a batch of multiple symbols.

Consider the transmission of a packet of N_s data symbols $\mathbf{s} = [s_1 \cdots s_{N_s}]^T$, where each symbol $s_i \in \{+1, -1\}$ is "spread" over N_f frames of duration T_f. The spacing between two pulses in one frame is fixed at D. Each frame is assigned a known user code $c_{ij} \in \{+1, -1\}, j = 1, \cdots, N_f$. The code varies from frame to frame, and can vary from symbol to symbol similar to the long code concept in CDMA. The receiver still has the simple structure with only one correlator as illustrated in Fig. 1. The structure of the transmitted pulse sequence is shown in Fig. 2.

The received signal at the antenna output is

$$\begin{aligned} y(t) \;=\; & \textstyle\sum_{i=1}^{N_s} \sum_{j=1}^{N_f} [h(t - ((i-1)N_c + j - 1)N_f T_f) \\ & + s_i c_{ij} h(t - ((i-1)N_c + j - 1)N_f T_f - D)] \end{aligned} \tag{6}$$

where $\mathbf{c}_i = [c_{i1}, \cdots, c_{iN_f}]^T$ is the code vector for the i-th symbol s_i.

At the multiplier output, the signal $x(t) = y(t)y(t-D)$ will be integrated and dumped at the oversampling rate $P = T_f/T_{sam}$. Due to uncorrelated channels, the unmatched terms and the cross-terms can be ignored for the purpose of receiver design. The data model in (5) can be easily extended to include the code c_{ij}. The resulting discrete samples $x[n] = \int_{(n-1)T_{sam}}^{nT_{sam}} x(t)dt$, $n = 1, \cdots, (N_s N_f - 1)P + T_h/T_{sam}$ are stacked into a column vector $\mathbf{x}$, which can be expressed as (see Fig. 3)

$$\mathbf{x} = \mathbf{H}\text{diag}\{\mathbf{c}_1, \cdots, \mathbf{c}_{N_s}\}\mathbf{s} + \text{noise} \tag{7}$$

where, as before, $\mathbf{H}$ contains shifted versions of the "channel" vector $\mathbf{h}$, and the 'diag' operator puts the vectors $\mathbf{c}_1, \cdots, \mathbf{c}_{N_s}$ into a block diagonal matrix.

The data model (7) can also be written in another form,

$$\mathbf{x} = \mathcal{C}(\mathbf{I}_{N_s} \otimes \mathbf{h})\mathbf{s} + \text{noise} \tag{8}$$

where $\otimes$ denotes the Kronecker product and $\mathcal{C}$ is the code matrix of size $((N_f N_s - 1)T_f + T_h)/T_{sam} \times (T_h N_s)/T_{sam}$, with entries taken from $\mathbf{c}_i$ and structure illustrated in Fig. 3. This form of data model will be used to derive the data model for multiuser, multi-delay cases.

2.4. Multiple users, single delay

Now we derive the data model for an asynchronous multiuser system where the k-th user is characterized by a code matrix $[\mathbf{c}_{k1}, \cdots, \mathbf{c}_{k,N_s}]$, channel vector $\mathbf{h}_k$, and time offset $G_k = G'_k T_{sam} + g_k$, $0 \leq g_k < T_{sam}$. The code and the integer G'_k are known, the channel $\mathbf{h}_k$ and g_k are unknown. Since each user goes through a different channel, we can safely assume that two different channels are uncorrelated, which means that all the cross-terms between two users' channels are noise-like. Therefore, the received signal will be modeled as

$$\begin{aligned} \mathbf{x} &= \sum_{k=1}^{K} \mathbf{H}_k \text{diag}\{\mathbf{c}_{k1}, \cdots, \mathbf{c}_{kN_s}\}\mathbf{s}_k + \text{noise} \\ &= \sum_{k=1}^{K} \mathcal{C}_k(\mathbf{I} \otimes \mathbf{h}_k)\mathbf{s}_k + \text{noise} \end{aligned}$$

where $\mathbf{H}_k$, $\mathcal{C}_k$ are the channel matrix and code matrix for the k-th user. They have structure as in Fig. 3, except that the time offset G_k shows up as G'_k zero padding rows at the top of the matrices $\mathbf{H}_k$ and $\mathcal{C}_k$. The effect of the offset fraction g_k is not visible in the model: only the values of the entries of the channel vector $\mathbf{h}_k$ are slightly changed.

The multiuser data model can be straightforwardly derived as

$$\mathbf{x} = \mathcal{C}\mathcal{H}\mathbf{s} + \text{noise} \tag{9}$$

where $\mathcal{C} = [\mathcal{C}_1 \cdots \mathcal{C}_K]$ is the known code matrix; $\mathcal{H} = \text{diag}\{\mathbf{I} \otimes \mathbf{h}_1, \cdots, \mathbf{I} \otimes \mathbf{h}_K\}$ is the unknown channel matrix, in which $\mathbf{h}_k$ contains the unknown channel coefficients; and $\mathbf{s} = [\mathbf{s}_1^T \cdots \mathbf{s}_K^T]^T$ contains the unknown source symbols.

2.5. Multiple users, multiple delays

In the previous sections, we used a fixed delay between the two pulses in a doublet (frame) to simplify the mathematical expressions and the receiver structure. However, better performance can be achieved if the delay between two pulses in a doublet is made to vary from frame to frame, in a known pattern.

Let the spacing between two pulses in a frame be d_{ij}^k seconds (corresponding to the k-th user, i-th symbol, j-th frame). As before, we choose the delay d_{ij}^k to be very small compared to the frame period and the channel length, i.e., $d_{ij}^k \ll T_f < T_h$. The values of all the delays d_{ij}^k are chosen from a finite set $d_{ij}^k \in \{D_1, D_2, \cdots, D_M\}$, of which the pattern is known to the receiver.

At the receiver, we use a bank of correlators, each followed by an "integrate and dump" operator as shown in Fig.4. We have M equations corresponding to the M branches of correlators $D_1, \cdots, D_M$. In the single user

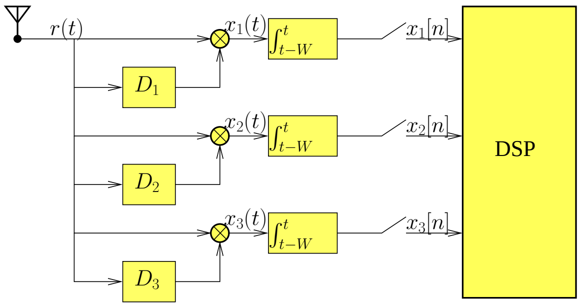

Figure 4: Receiver structure with multiple correlators

case, each equation has a similar expression to (7) and (8),

$$\mathbf{x}^{(m)} = \mathbf{H}^{(m)}\text{diag}\{\mathbf{c}'_1, \cdots, \mathbf{c}'_{N_s}\}\mathbf{s} + \text{noise}\,, \qquad (10)$$

$(m = 1, \cdots, M)$, where $\mathbf{x}^{(m)}$ is a vector containing the received samples of the m-th branch, and $\mathbf{H}^{(m)}$ is similar to $\mathbf{H}$ as before. The code vector $\mathbf{c}'_i$ has entries corresponding to each user, frame and delay. If the delay matches the delay code, the entry contains the corresponding chip value $\{+1, -1\}$, otherwise the entry is 0.
In the data model, we should take into account that all the branches share the same "channel" coefficients $\mathbf{h}$ and the symbol values $\mathbf{s}$. To this end, we first rewrite the data model of a single branch that corresponds to delay D_m (10) in the "code" by "channel" by "data" form, as

$$\mathbf{x}^{(m)} = \mathcal{C}^{(m)}(\mathbf{I} \otimes \mathbf{h})\mathbf{s} + \text{noise}\,, \qquad (11)$$

where $\mathcal{C}^{(m)}$ is a code matrix with structure as before, but with nonzero entries only for frames that have delay codes that match delay D_m. Stacking all received samples in all branches into a column vector, the data model for a single user, multi-delay receiver becomes

$$\mathbf{x} = \mathcal{C}(\mathbf{I} \otimes \mathbf{h})\mathbf{s} + \text{noise}\,, \quad \mathcal{C} = \begin{bmatrix} \mathcal{C}^{(1)} \\ \vdots \\ \mathcal{C}^{(M)} \end{bmatrix}. \qquad (12)$$

From this equation, the data model for multiuser, multi-delay receiver case can be straightforwardly derived in a similar way as presented in the previous section. The multiuser multi-delay data model becomes

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \mathcal{C}_1^{(1)} & \cdots & \mathcal{C}_K^{(1)} \\ \vdots & \ddots & \vdots \\ \mathcal{C}_1^{(M)} & \cdots & \mathcal{C}_K^{(M)} \end{bmatrix} \begin{bmatrix} \mathbf{I}\otimes\mathbf{h}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{I}\otimes\mathbf{h}_K \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 \\ \vdots \\ \mathbf{s}_K \end{bmatrix} \\ &=: \mathcal{C}\mathcal{H}\mathbf{s} + \text{noise} \end{aligned} \qquad (13)$$

where $\mathcal{C}_k^{(m)}$ is the code matrix corresponding to the k-user, m-th correlator branch. This matrix contains information regarding the user's chip code, delay code, and time offset.

By using a property of the Kronecker product: $(\mathbf{I} \otimes \mathbf{h}_k)\mathbf{s}_k = (\mathbf{s}_k \otimes \mathbf{I})\mathbf{h}_k$, the data model above ($\mathbf{x} = \mathcal{C}\mathcal{H}\mathbf{s}$) can be rewritten in another form ($\mathbf{x} = \mathcal{C}\mathcal{S}\mathbf{h}$) as

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \mathcal{C}_1^{(1)} & \cdots & \mathcal{C}_K^{(1)} \\ \vdots & \ddots & \vdots \\ \mathcal{C}_1^{(M)} & \cdots & \mathcal{C}_K^{(M)} \end{bmatrix} \begin{bmatrix} \mathbf{s}_1\otimes\mathbf{I} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{s}_K\otimes\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{h}_1 \\ \vdots \\ \mathbf{h}_K \end{bmatrix} \\ &=: \mathcal{C}\mathcal{S}\mathbf{h} + \text{noise}\,. \end{aligned} \qquad (14)$$

3. RECEIVER ALGORITHMS

3.1. Alternating least squares receiver algorithm

In equations (13) and (14), $\mathcal{H}$, $\mathcal{S}$ are matrices with known structures, constructed from the channel vector $\mathbf{h}$ and source symbols vector $\mathbf{s}$, respectively. In this equation, $\mathbf{x}$ is the (known) data sample vector, $\mathcal{C}$ is the known code matrix, while $\mathbf{s}$ and $\mathbf{h}$ are the unknowns. Based on these two forms, an alternating least squares (ALS) algorithm can be implemented as follows.
With an initial channel estimate $\mathbf{h}^{(0)}$, for iteration index $i = 1, 2, \cdots$ until convergence,

- keeping the channel $\mathbf{h}^{(i-1)}$ fixed, construct the $\mathcal{H}$ matrix, and estimate the source symbols via

$$\mathbf{s}^{(i)} = (\mathcal{C}\mathcal{H})^{\dagger}\mathbf{x}\,,$$

  where $(\cdot)^{\dagger}$ indicates the Moore-Penrose pseudo-inverse (in this case equal to the left inverse),

- keeping the source symbols $\mathbf{s}^{(i)}$ fixed, construct the $\mathcal{S}$ matrix, and estimate the channel coefficients via

$$\mathbf{h}^{(i)} = (\mathcal{C}\mathcal{S})^{\dagger}\mathbf{x}\,.$$

After these iterations, step 1 is repeated once more to get the final estimate of the source symbols. Hard decisions can be used in step 1 to further improve the performance.

3.2. Initialization—A blind algorithm

The ALS algorithm needs an initial channel estimate. A simple blind algorithm, similar to the algorithm in [9], is as follows. We assume that the code matrix in equation (13) is tall, which implies

$$M((N_sN_f - 1)T_f + T_h)/T_{sam} > KT_hN_s/T_{sam}$$

We can then pre-multiply both sides of (13) with the left-inverse of this known matrix. The resulting multiuser equation can be decomposed into K single user equations,

$$\mathbf{x}'_k \approx (\mathbf{I} \otimes \mathbf{h}_k)\mathbf{s}_k\,, \quad k = 1, \cdots, K\,,$$

where $\mathbf{x}'_k$ is the k-th segment of $\mathbf{x}' = \mathcal{C}^\dagger \mathbf{x}$.
After restacking the vector $\mathbf{x}'_k$ into a matrix $\mathbf{X}'_k$ of size $L_h \times N_s$ as in [9], we have

$$\mathbf{X}'_k \approx \mathbf{h}_k \mathbf{s}_k^T .$$

Subsequently, the channel vector $\mathbf{h}_k$ and the source symbols $\mathbf{s}_k$ of the k-th user are found, up to an unknown scaling, by taking a rank-1 approximation of $\mathbf{X}'_k$. This requires the computation of the SVD of $\mathbf{X}'_k$ and keeping the dominant component.

3.3. Computational complexity

The proposed algorithms are all two-step iterations. The complexity of one iteration is derived here. For simplicity of the expressions, we assume that all users have the same parameters and time offsets. As before, $L_h = \frac{T_h}{T_{sam}}$ is the channel length in terms of number of samples. Let $L = \frac{T_h}{T_f} = \frac{L_h}{P}$ be the channel length in terms of frames, assumed an integer number here.

1. *Given the channel coefficients* $\mathbf{h}$, *estimate the source symbols* $\mathbf{s}$ *by solving* $\mathbf{x} = \mathcal{CH}\mathbf{s}$. This is done by the following steps:

$$\begin{array}{lll} \mathbf{T} = \mathcal{CH} & : & KN_sN_fLP \quad \text{operations} \\ \mathbf{y} = \mathbf{T}^H\mathbf{x} & : & KN_sMP(N_f + L) \\ \mathbf{M} = \mathbf{T}^H\mathbf{T} & : & K^2N_sMP(N_f + L + \frac{L^2}{N_f}) \\ \text{Solve } \mathbf{Ms} = \mathbf{y} & : & N_sK^3(2 + \frac{L}{N_f})^2 \end{array}$$

   The dominant operation is the computation of $\mathbf{M}$. Thus, the order of complexity of the estimation of $\mathbf{s}$ is $K^2N_sMP(N_f + L + \frac{L^2}{N_f})$.

2. *Given* $\mathbf{s}$, *estimate the channel coefficients* $\mathbf{h}$ *by solving* $\mathbf{x} = \mathcal{CS}\mathbf{h}$. This is done by the following steps:

$$\begin{array}{lll} \mathbf{T} = \mathcal{CS} & : & \text{(only composition)} \\ \mathbf{y} = \mathbf{T}^H\mathbf{x} & : & KN_sN_fLP \quad \text{additions} \\ \mathbf{M} = \mathbf{T}^H\mathbf{T} & : & K^2N_sN_fPL^2 \quad \text{additions} \\ \text{Solve } \mathbf{Mh} = \mathbf{y} & : & K^2PL^2 \quad \text{operations} \end{array}$$

   In total, the complexity is $K^2N_sN_fPL^2$ additions plus K^2PL^2 multiply/additions.

Overall, solving for $\mathbf{s}$ gives the dominant complexity. One iteration thus has a complexity of order $K^2N_sMP(N_f + L + \frac{L^2}{N_f})$ operations. Per estimated symbol per user, the complexity is $KMP(N_f + L + \frac{L^2}{N_f})$. Compare this to a single antenna CDMA multiuser decorrelating receiver, which has complexity per user per symbol of order KN_f or LN_f, depending on the type of receiver [10]. The increased complexity (factor MP) is due to the multi-branch nature of the TR-UWB receiver structure, and would be similar to the use of multiple antennas.

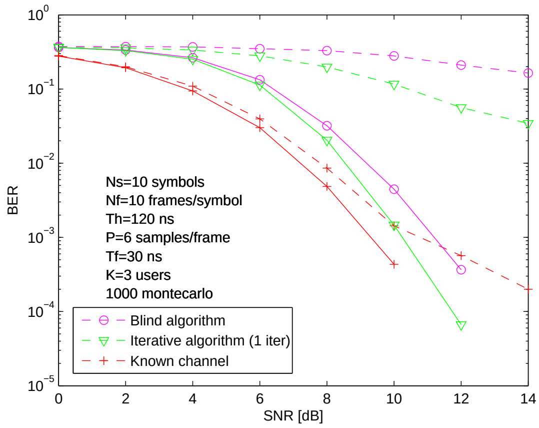

Figure 5: BER vs. SNR performance comparison between single delay and multi-delay schemes for CM2. Solid: multiple delays ($M = 4$), dashed: single delay ($M = 1$).

4. SIMULATIONS

We simulate an asynchronous multiuser TR-UWB system with $K = 3$ equal powered users transmitting Gaussian monocycle pulses of width 0.2 ns. The spacing between two pulses in a doublet may vary in frames, symbols and users, with values taken from the set $\{1, 2, 3, 4\}$ ns. In one user's data packet, we transmit $N_s = 10$ symbols, each symbol consists of $N_f = 10$ frames with duration $T_f = 30$ ns. All the users' symbols and codes are generated randomly. Each user signal is delayed by a random (but known) offset of up to one frame duration, rounded to an integer number of samples. The sampling rate is $T_{sam} = T_f/P$ and depends on the chosen over-sampling rate, which can be $P \in \{3, 6, 15\}$ samples per frame.
We use the IEEE channel model CM2, which is always longer than the frame period, implying that inter-frame interference (IFI) does exist. The non-ideal antenna effect is also included, i.e. a measured antenna response is convolved with the channel and the pulse.
A reference curve is the performance of the zero-forcing receiver when the channel coefficients are completely known.
Fig. 5 shows the BER performance gain of the multiple delay scheme (with $M = 4$ different delays in total) compared to the single delay scheme for the IEEE channel model CM2. The solid lines denote the multiple delay case, the dashed lines denote the single delay case. It is seen that the gain is significant, both for the blind algorithm used for initialization and for the iterative algorithm. The gaps widen as SNR increases. The main reason is the 'diversity' offered by the M correlation banks at the receiver: the code matrix $\mathcal{C}$ and the matrices $\mathcal{CH}$, $\mathcal{CS}$ are M

times taller, which will improve the algorithms' performance and eliminate the BER flooring effect in the high SNR region. $M = 4$ delays gives performance quite close to the reference curve (ZF receiver with known channel).

5. TRANSCEIVER DESIGN ISSUES

To conclude the paper, we will take into account some of the implications in this paper for the design of a practical TR-UWB system. What are the constraints on the system parameter values?

A first constraint is posed by the receiver bandwidth, which is limited by spectral masks or antenna design constraints. The finite bandwidth determines the correlation distance τ_0. In the receiver algorithm design, we ignored all correlations beyond τ_0. For the practical antenna used in the simulations (bandwidth 5 GHz), we found that we can safely choose $\tau_0 = 1$ ns. Therefore, the most closely spaced set of possible delays is $\{D_1, \cdots, D_M\} = \{1, 2, 3, \cdots\}$ ns.

The number of delays M is often constrained by practical considerations: the analog delay lines do take physical space in the receiver, and the receiver algorithm's complexity increases linearly with M. Therefore, we can often afford only a limited number of delays, say, $M \leq 5$.

Two constraints restrict the choice of the frame size T_f. Firstly, the last pulse of a frame must not overlap with the first pulse of the next frame, even after a maximal delay D_M. Therefore,

$$T_f > 2M\tau_0 \, .$$

Secondly, for the blind initialization algorithm described in section 3.2 to work, the code matrix $\mathcal{C}$ must be invertible, hence tall, which implies the condition: $M((N_sN_f - 1)T_f + T_h)/T_{sam} > KT_hN_s/T_{sam}$. This can be approximately reduced to:

$$MN_fT_f > KT_h \, .$$

This expression defines a trade-off between the coding gain (or the symbol period $T_s = N_fT_f$) and the number of users K given the number of delays M and the channel length T_h.

If our aim is to have as high-rate system as possible, then we would set $K = 1$ user, and $N_f = 1$ chips/symbol. The two preceding inequalities give

$$\frac{T_h}{T_f} < M < \frac{1}{2}\frac{T_f}{\tau_0}$$

which leads to

$$T_f > \sqrt{2T_h\tau_0} \, .$$

This provides a limit on the data rate. For example, if $T_h = 80$ ns and $\tau_0 = 1$ ns, then $T_f > 13$ ns. To have an integer M, we choose T_f a bit larger, e.g., $T_f = 15$ ns corresponding to a data rate of about 66 Mbps. It follows that $M \in \{6, 7\}$.

For a more economic receiver, we would probably take the code length N_f larger. This will lead to a lower data rate, and enables a lower M.

The oversampling rate P can be chosen based on the trade-off between the BER performance (shown in simulations) and the receiver's complexity (shown in section 3.3). Computationally, oversampling (P) and multiple delays (M) play almost equivalent roles. Both give rise to a multi-branch model. The difference is in the complexity of the analog hardware: oversampling requires faster samplers, whereas multiple delays require more circuitry that runs in parallel. Increasing the code length (N_f) does not cost additional hardware but slows down the data rate and improves the BER performance as usual.

6. REFERENCES

[1] Q.H. Dang and A.J. van der Veen, "Resolving inter-frame interference in a transmit-reference UWB communication system," in *ICASSP*, Toulouse, France, 2006.

[2] Q.H. Dang and A.J. van der Veen, "Signal processing model and receiver algorithms for a higher rate multiuser TR-UWB communication system," in *submitted to ICASSP*, Honolulu, HI, 2007.

[3] Z. Tian and G. Giannakis, "BER sensitivity to timing offset in ultra wideband communications," in *IEEE SPAWC*, 2003.

[4] R. Hoctor and H. Tomlinson, "Delay-hopped transmitted-reference RF communications," in *IEEE Conf. Ultra Wideband Syst. Techn.*, 2002, pp. 265–270.

[5] J. Choi and W. Stark, "Performance of ultra-wideband communications with suboptimal receivers in multipath channels," *IEEE J. Sel. Areas Comm.*, vol. 20, no. 9, pp. 1754–1766, Dec. 2002.

[6] S.R. Aedudola, S. Vijayakumaran, and T.F. Wong, "Acquisition of direct-sequence transmitted reference ultra-wideband signals," *IEEE J. Sel. Areas Comm.*, vol. 24, pp. 759–765, Apr. 2006.

[7] S. Bagga, Lujun Zhang, W.A. Serdijn, J.R. Long, and E.B. Busking, "A quantized analog delay for an IR-UWB quadrature downconversion autocorrelation receiver," in *IEEE Int. Conf. Ultra Wideband*, Zurich (CH), Sept. 2005.

[8] Q.H. Dang, A.J. van der Veen, A. Trindade, and G. Leus, "Signal model and receiver algorithms for a transmit-reference ultra-wideband communication system," *IEEE J. Sel. Areas Comm.*, Apr. 2006.

[9] L. Tong, A.J. van der Veen, P. Dewilde, and Y. Sung, "Blind decorrelating rake receivers for long-code WCDMA," *IEEE Tr. Signal Proc.*, vol. 51, no. 6, pp. 1642–1655, June 2003.

[10] Q.H. Dang and A.J. van der Veen, "A low-complexity blind multi-user receiver for long-code WCDMA," *EURASIP J. Wireless Comm. Netw.*, vol. 2004, no. 1, pp. 113–122, Aug. 2004.