Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2009, Article ID 315264, 17 pages
doi:10.1155/2009/315264

Research Article

Digital Receiver Design for Transmitted Reference Ultra-Wideband Systems

Yiyin Wang, Geert Leus, and Alle-Jan van der Veen

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS), Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

Correspondence should be addressed to Yiyin Wang, yiyin.wang@tudelft.nl

Received 30 June 2008; Revised 6 November 2008; Accepted 1 February 2009

Recommended by Erdal Panayirci

A complete detection, channel estimation, synchronization, and equalization scheme for a transmitted reference (TR) ultra-wideband (UWB) system is proposed in this paper. The scheme is based on a data model which admits a moderate data rate and takes both the interframe interference (IFI) and the intersymbol interference (ISI) into consideration. Moreover, the bias caused by the interpulse interference (IPI) in one frame is also taken into account. Based on the analysis of the stochastic properties of the received signals, several detectors are studied and evaluated. Furthermore, a data-aided two-stage synchronization strategy is proposed, which obtains sample-level timing in the range of one symbol at the first stage and then pursues symbol-level synchronization by looking for the header at the second stage. Three channel estimators are derived to achieve joint channel and timing estimates for the first stage, namely, the linear minimum mean square error (LMMSE) estimator, the least squares (LS) estimator, and the matched filter (MF). We check the performance of different combinations of channel estimation and equalization schemes and try to find the best combination, that is, the one providing a good tradeoff between complexity and performance.

1. Introduction

Ultra-wideband (UWB) techniques can provide high speed, low cost, and low complexity wireless communications with the capability to overlay existing frequency allocations [1]. Since UWB systems employ ultrashort low duty cycle pulses as information carriers, they suffer from stringent timing requirements [1, 2] and complex multipath channel estimation [1]. Conventional approaches require a prohibitively high sampling rate of several GHz [3] and an intensive multidimensional search to estimate the parameters for each multipath echo [4].

Detection, channel estimation, and synchronization problems are always entangled with each other. A typical approach to address these problems is the detection-based signal acquisition [5]. A locally generated template is correlated with the received signal, and the result is compared to a threshold. How to generate a good template is the task of channel estimation, whereas how to decide the threshold is the goal of detection. Due to the multipath channel, the complexity of channel estimation grows quickly as the number of multipath components increases, and because of the fine resolution of the UWB signal, the search space is extremely large.

Recent research works on detection, channel estimation, and synchronization methods for UWB have focused on low sampling rate methods [6–9] or noncoherent systems, such as transmitted reference (TR) systems [5, 10], differential detectors (DDs) [11], and energy detectors (EDs) [9, 12]. In [6], a generalized likelihood ratio test (GLRT) for frame-level acquisition based on symbol rate sampling is proposed, which works with no or small interframe interference (IFI) and no intersymbol interference (ISI). The whole training sequence is assumed to be included in the observation window without knowing the exact starting point. Due to its low duty cycle, an UWB signal belongs to the class of signals that have a finite rate of innovation [7]. Hence, it can be sampled below the Nyquist sampling rate, and the timing information can be estimated by standard methods. The theory is developed under the simplest scenario, and extensions

are currently envisioned [13]. The timing recovery algorithm of [8] makes cross-correlations of successive symbol-long received signals, in which the feedback controlled delay lines are difficult to implement. In [9], the authors address a timing estimation comparison among different types of transceivers, such as stored-reference (SR) systems, ED systems, and TR systems. The ED and the TR systems belong to the class of noncoherent receivers. Although their performances are suboptimal due to the noise contaminated templates, they attract more and more interest because of their simplicity. They are also more tolerant to timing mismatches than SR systems. The algorithms in [9] are based on the assumption that the frame-level acquisition has already been achieved. Two-step strategies for acquisition are described in [14, 15]. In [14], the authors use a different search strategy in each step to speed up the procedure, which is the bit reversal search for the first step and the linear search for the second step. Meanwhile, the two-step procedure in [15] finds the block which contains the signal in the first step, and aligns with the signal at a finer resolution in the second step. Both methods are based on the assumption that coarse acquisition has already been achieved to limit the search space to the range of one frame and that there are no interferences in the signal.

From a system point of view, noncoherent receivers are considered to be more practical since they can avoid the difficulty of accurate synchronization and complicated channel estimation. One main obstacle for TR systems and DD systems is the implementation of the delay line [16]. The longer the delay line is, the more difficult it is to implement. For DD systems [11], the delay line is several frames long, whereas for TR systems, it can be only several pulses long [17], which is much shorter and easier to implement [18]. ED systems do not need a delay line, but suffer from multiple access interference [19], since they can only adopt a limited number of modulation schemes, such as on-off keying (OOK) and pulse position modulation (PPM). A two-stage acquisition scheme for TR-UWB systems is proposed in [5], which employs two sets of direct-sequence (DS) code sequences to facilitate coarse timing and fine aligning. The scheme assumes no IFI and ISI. In [20], a blind synchronization method for TR-UWB systems executes an MUSIC-kind of search in the signal subspace to achieve high-resolution timing estimation. However, the complexity of the algorithm is very high because of the matrix decomposition.

Recently, a multiuser TR-UWB system that admits not only interpulse interference (IPI), but also IFI and ISI was proposed in [21]. The synchronization for such a system is at low-rate sample-level. The analog parts can run independently without any feedback control from the digital parts. In this paper, we develop a complete detection, channel estimation, synchronization, and equalization scheme based on the data model modified from [21]. Moreover, the performance of different kinds of detectors is assessed. A two-stage synchronization strategy is proposed to decouple the search space and speed up synchronization. The property of the circulant matrix in the data model is exploited to reduce the computational complexity. Different combinations of channel estimators and equalizers are evaluated to find the one with the best tradeoff between performance and complexity. The results confirm that the TR-UWB system is a practical scheme that can provide moderate data rate communications (e.g., in our simulation setup, the data rate is 2.2 Mb/s) at a low cost.

The paper is organized as follows. In Section 2, the data model presented in [21] is summarized and modified to take the unknown timing into account. Further, the statistics of the noise are derived. The detection problem is addressed in Section 3. Channel estimation, synchronization, and equalization are discussed in Section 4. Simulation results are shown and assessed in Section 5. Conclusions are drawn in Section 6.

Notation. We use upper (lower) bold face letters to denote matrices (column vectors). $x(\cdot)(x[\cdot])$ represents a continuous (discrete) time sequence. $\mathbf{0}_{m\times n}$ ($\mathbf{1}_{m\times n}$) is an all-zero (all-one) matrix of size $m \times n$, while $\mathbf{0}_m$ ($\mathbf{1}_m$) is an all-zero (all-one) column vector of length m. $\mathbf{I}_m$ indicates an identity matrix of size $m \times m$. $\star$, $\otimes$ and $\odot$ indicate time domain convolution, Kronecker product, and element-wise product. $(\cdot)^\dagger$, $(\cdot)^T$, $(\cdot)^H$, $|\cdot|$, and $\|\cdot\|_F$ designate pseudoinverse, transposition, conjugate transposition, absolute value, and Frobenius norm. All other notation should be self-explanatory.

2. Asynchronous Single User Data Model

The asynchronous single user data model derived in the following paragraphs uses the data model in [21] as a starting point. We take the unknown timing into consideration and modify the model in [21].

2.1. Single Frame. In a TR-UWB system [10, 21], pairs of pulses (doublets) are transmitted in sequence as shown in Figure 1. The first pulse in the doublet is the reference pulse, whereas the second one is the data pulse. Since both pulses go through the same channel, the reference pulse can be used as a "dirty template" (noise contaminated) [8] for correlation at the receiver. One frame-period T_f holds one doublet. Moreover, N_f frames constitute one symbol period $T_s = N_f T_f$, which is carrying a symbol $s_i \in \{-1, +1\}$, spread by a pseudorandom code $c_j \in \{-1, +1\}$, $j = 1, 2, \ldots, N_f$, which is repeatedly used for all symbols. The polarity of a data pulse is modulated by the product of a frame code and a symbol. The two pulses are separated by some delay interval D_m, which can be different for each frame. The delay intervals are in the order of nanoseconds and $D_m \ll T_f$. The receiver employs multiple correlation branches corresponding to different delay intervals. To simplify the system, we use a single delay and one correlation branch, which implies $D_m = D$. Figure 1 also presents an example of the receiver structure for a single delay D. The integrate-and-dump (I&D) integrates over an interval of length T_{sam}. As a result, one frame results in $P = T_f/T_{\text{sam}}$ samples, which is assumed to be an integer.

The received one-frame signal (jth frame of ith symbol) at the antenna output is

$$r(t) = h(t-\tau) + s_i c_j h(t - D - \tau) + n(t), \tag{1}$$

where τ is the unknown timing offset, $h(t) = h_p(t) \star g(t)$ of length T_h with $h_p(t)$ the UWB physical channel and $g(t)$ the pulse shape resulting from all the filter and antenna effects, and $n(t)$ is the bandlimited additive white Gaussian noise (AWGN) with double-sided power spectral density $N_0/2$ and bandwidth B. Without loss of generality, we may assume that the unknown timing offset τ in (1) is in the range of one symbol period, $\tau \in [0, T_s)$, since we know the signal is present by detection at the first step (see Section 3) and propose to find the symbol boundary before acquiring the package header (see Section 4). Then, τ can be decomposed as

$$\tau = \delta \cdot T_{\text{sam}} + \epsilon, \tag{2}$$

where $\delta = \lfloor \tau / T_{\text{sam}} \rfloor \in \{0, 1, \ldots, L_s - 1\}$ denotes the sample-level offset in the range of one symbol with $L_s = N_f P$, the symbol length in terms of number of samples, and $\epsilon \in [0, T_{\text{sam}})$ presents the fractional offset. Sample-level synchronization consists of estimating δ. The influence of ϵ will be absorbed in the data model and becomes invisible as we will show later.

Based on the received signal $r(t)$, the correlation branch of the receiver computes

$$\begin{aligned} x[n] &= \int_{(n-1)T_{\text{sam}}+D}^{nT_{\text{sam}}+D} r(t) r(t-D)\, dt \\ &= \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \{[h(t-\tau) + s_i c_j h(t-D-\tau) + n(t)] \\ &\qquad \times [h(t+D-\tau) + s_i c_j h(t-\tau) + n(t+D)]\}\, dt \\ &= s_i c_j \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h^2(t-\tau) + h(t-D-\tau) h(t+D-\tau)]\, dt \\ &\quad + \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau) h(t+D-\tau) \\ &\quad + h(t-D-\tau) h(t-\tau)]\, dt + n_1[n], \end{aligned} \tag{3}$$

where

$$\begin{aligned} n_1[n] &= n_0[n] + s_i c_j \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau) n(t) \\ &\quad + h(t-D-\tau) n(t+D)]\, dt \\ &\quad + \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau) n(t+D) \\ &\quad + h(t+D-\tau) n(t)]\, dt \end{aligned} \tag{4}$$

with

$$n_0[n] = \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} n(t) n(t+D)\, dt. \tag{5}$$

Note that $n_0[n]$ is the noise autocorrelation term, and $n_1[n]$ encompasses the signal-noise cross-correlation term and the noise autocorrelation term. Their statistics will be analyzed later. Taking ϵ into consideration, we can define the channel correlation function similarly as in [21]

$$\begin{aligned} &R(\Delta, m) \\ &\quad = \int_{(m-1)T_{\text{sam}}}^{mT_{\text{sam}}} h(t-\epsilon) h(t-\epsilon-\Delta)\, dt, \ m = 1, 2, \ldots, \end{aligned} \tag{6}$$

where $h(t) = 0$, when $t > T_h$ or $t < 0$. Therefore, the first term in (3) can be denoted as $s_i c_j \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h^2(t-\tau)\, dt = s_i c_j \int_{(n-1)T_{\text{sam}}-\delta T_{\text{sam}}}^{nT_{\text{sam}}-\delta T_{\text{sam}}} h^2(t-\epsilon)\, dt = s_i c_j R(0, n-\delta)$. Other terms in $x[n]$ can also be rewritten in a similar way, leading $x[n]$ to be

$$\begin{aligned} &x[n] \\ &\quad = \begin{cases} s_i c_j \left[R(0, n-\delta) + R\left(2D,\ n-\delta+\dfrac{D}{T_{\text{sam}}}\right)\right] \\ \quad + \left[R(D, n-\delta) + R\left(D, n-\delta+\dfrac{D}{T_{\text{sam}}}\right)\right] + n_1[n], \\ \qquad\qquad n = \delta+1, \delta+2, \ldots, \delta+P_h, \\ n_0[n], \quad \text{elsewhere}, \end{cases} \end{aligned} \tag{7}$$

where $P_h = \lceil T_h / T_{\text{sam}} \rceil$ is the channel length in terms of number of samples, and $R(0, m)$ is always nonnegative. Although $R(2D,\ m + D/T_{\text{sam}})$ is always very small compared to $R(0, m)$, we do not ignore it to make the model more accurate. We also take the two bias terms into account, which are the cause of the IPI and are independent of the data symbols and the code. Now, we can define the $P_h \times 1$ channel energy vector $\mathbf{h}$ with entries h_m as

$$h_m = R(0, m) + R\left(2D, m + \frac{D}{T_{\text{sam}}}\right), \quad m = 1, \ldots, P_h, \tag{8}$$

where $R(0, m) \geq 0$. Further, the $P_h \times 1$ bias vector $\mathbf{b}$ with entries b_m is defined as

$$b_m = R(D, m) + R\left(2D, m + \frac{D}{T_{\text{sam}}}\right), \quad m = 1, \ldots, P_h. \tag{9}$$

Note that these entries will change as a function of ϵ, although ϵ is not visible in the data model. As we stated before, sample-level synchronization is limited to the estimation of δ. Using (8) and (9), $x[n]$ can be represented as

$$\begin{aligned} &x[n] \\ &\quad = \begin{cases} s_i c_j h_{n-\delta} + b_{n-\delta} + n_1[n], \quad n = \delta+1, \delta+2, \ldots, \delta+P_h, \\ n_0[n], \quad \text{elsewhere}. \end{cases} \end{aligned} \tag{10}$$

Now we can turn to the noise analysis. A number of papers have addressed the noise analysis for TR systems [22–25]. The noise properties are summarized here, and more

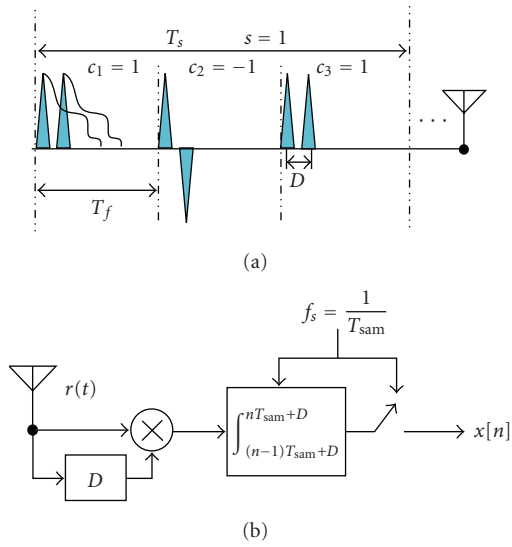

Figure 1: The transmitted UWB signal and the receiver structure.

details can be found in Appendix A. We start by making the assumptions that $D \gg 1/B$, $T_{\text{sam}} \gg 1/B$, and the time-bandwidth product $2BT_{\text{sam}}$ is large enough. Under these assumptions, the noise autocorrelation term $n_0[n]$ can be assumed to be a zero mean white Gaussian random variable with variance $\sigma_0^2 = N_0^2 BT_{\text{sam}}/2$. The other noise term $n_1[n]$ includes the signal-noise cross-correlation and the noise autocorrelation, and can be interpreted as a random disturbance of the received signal. Let us define two other $P_h \times 1$ channel energy vectors $\mathbf{h}'$ and $\mathbf{h}''$ with entries h'_m and h''_m to be used in the variance of $n_1[n]$ as follows:

$$h'_m = R(0,m) + R\left(0, m - \frac{D}{T_{\text{sam}}}\right), \quad m = 1,\ldots,P_h, \tag{11}$$

$$h''_m = R(0,m) + R\left(0, m + \frac{D}{T_{\text{sam}}}\right), \quad m = 1,\ldots,P_h. \tag{12}$$

Using those definitions and under the earlier assumptions, $n_1[n]$ can also be assumed to be a zero mean Gaussian random variable with variance $(N_0/2)(h'_{n-\delta} + h''_{n-\delta} + 2s_i c_j b_{n-\delta}) + \sigma_0^2$, $n = \delta+1, \delta+2, \ldots, \delta+P_h$. This indicates that all the noise samples are uncorrelated with each other and have a different variance depending on the data symbol, the frame code, the channel correlation coefficients, and the noise level. Note that the noise model is as complicated as the signal model.

2.2. Multiple Frames and Symbols. Now let us extend the data model to multiple frames and symbols. We assume the channel length P_h is not longer than the symbol length L_s. A single symbol with timing offset τ will then spread over at most three adjacent symbol periods. Define $\mathbf{x}_k = [x[(k-1)L_s+1], x[(k-1)L_s+2], \ldots, x[kL_s]]^T$, which is an L_s -long sample vector. By stacking $M+N-1$ such received sample vectors into an $ML_s \times N$ matrix

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_k & \mathbf{x}_{k+1} & \cdots & \mathbf{x}_{k+N-1} \\ \mathbf{x}_{k+1} & \mathbf{x}_{k+2} & \cdots & \mathbf{x}_{k+N} \\ \vdots & & \cdots & \vdots \\ \mathbf{x}_{k+M-1} & \mathbf{x}_{k+M} & \cdots & \mathbf{x}_{k+M+N-2} \end{bmatrix}, \tag{13}$$

where N indicates the number of samples in each row of $\mathbf{X}$, and M denotes the number of sample vectors in each column of $\mathbf{X}$, we obtain the following decomposition:

$$\mathbf{X} = \mathbf{C}_{\delta'}(\mathbf{I}_{M+2} \otimes \mathbf{h})\mathbf{S} + \mathbf{B}_{\delta'}\mathbf{1}_{(MN_f+2N_f)\times N} + \mathbf{N}_1, \tag{14}$$

where $\mathbf{N}_1$ is the noise matrix similarly defined as $\mathbf{X}$,

$$\mathbf{S} = \begin{bmatrix} s_{k-1} & s_k & \cdots & s_{k+N-2} \\ s_k & s_{k+1} & \cdots & s_{k+N-1} \\ \vdots & & \cdots & \vdots \\ s_{k+M} & s_{k+M+1} & \cdots & s_{k+M+N-1} \end{bmatrix}, \tag{15}$$

and the structure of the other matrices is illustrated in Figure 2. We first define a code matrix $\mathbf{C}$. It is a block Sylvester matrix of size $(L_s + P_h - P) \times P_h$, whose columns are shifted versions of the extended code vector: $[c_1, \mathbf{0}_{P-1}^T, c_2, \mathbf{0}_{P-1}^T, \ldots, c_{N_f}, \mathbf{0}_{P-1}^T]^T$. The shift step is one sample. Its structure is shown in Figure 3. The matrix $\mathbf{C}_{\delta'}$ of size $ML_s \times (MP_h + 2P_h)$ is composed of $M+2$ block columns, where $\delta = (L_s - \delta') \bmod L_s$, $\delta' \in \{0,1,\ldots,L_s-1\}$. As long as there are more than two sample vectors ($M > 2$) stacked in every column of $\mathbf{X}$, the nonzero parts of the block columns will contain $M-2$ code matrices $\mathbf{C}$. The nonzero parts of the first and last two block columns result from splitting the code matrix $\mathbf{C}$ according to δ': $\mathbf{C}'_i(2L_s - i + 1 : 2L_s, :) = \mathbf{C}(1 : i, :)$ and $\mathbf{C}''_i(1 : L_s + P_h - P - i, :) = \mathbf{C}(i+1 : L_s + P_h - P, :)$, where $\mathbf{A}(m : n, :)$ refers to column m through n of $\mathbf{A}$. The overlays between frames and symbols observed in $\mathbf{C}_{\delta'}$ indicate the existence of IFI and ISI. Then we define a bias matrix $\mathbf{B}$ which is of size $(L_s + P_h - P) \times N_f$ made up by shifted versions of the bias vector $\mathbf{b}$ with a shift step of P samples, as shown in Figure 3. The matrix $\mathbf{B}_{\delta'}$ of size $ML_s \times (MN_f + 2N_f)$ also has $M+2$ block columns, the nonzero parts of which are obtained from the bias matrix $\mathbf{B}$ in the same way as $\mathbf{C}_{\delta'}$. Since the bias is independent of the data symbols and the code, it is the same for each frame. Each column of the resulting matrix $\mathbf{B}_{\delta'}\mathbf{1}_{(MN_f+2N_f)\times N}$ is the same and has a period of P samples. Defining $\mathbf{b}_f$ to be the $P \times 1$ bias vector for one such period, we have

$$\mathbf{B}_{\delta'}\mathbf{1}_{(MN_f+2N_f)\times N} = \mathbf{1}_{MN_f \times N} \otimes \mathbf{b}_f. \tag{16}$$

Note that $\mathbf{b}_f$ is also a function of δ, but since it is independent of the code, we cannot extract the timing information from it.

Recalling the noise analysis of the previous section, the noise matrix $\mathbf{N}_1$ has zero mean and contains uncorrelated

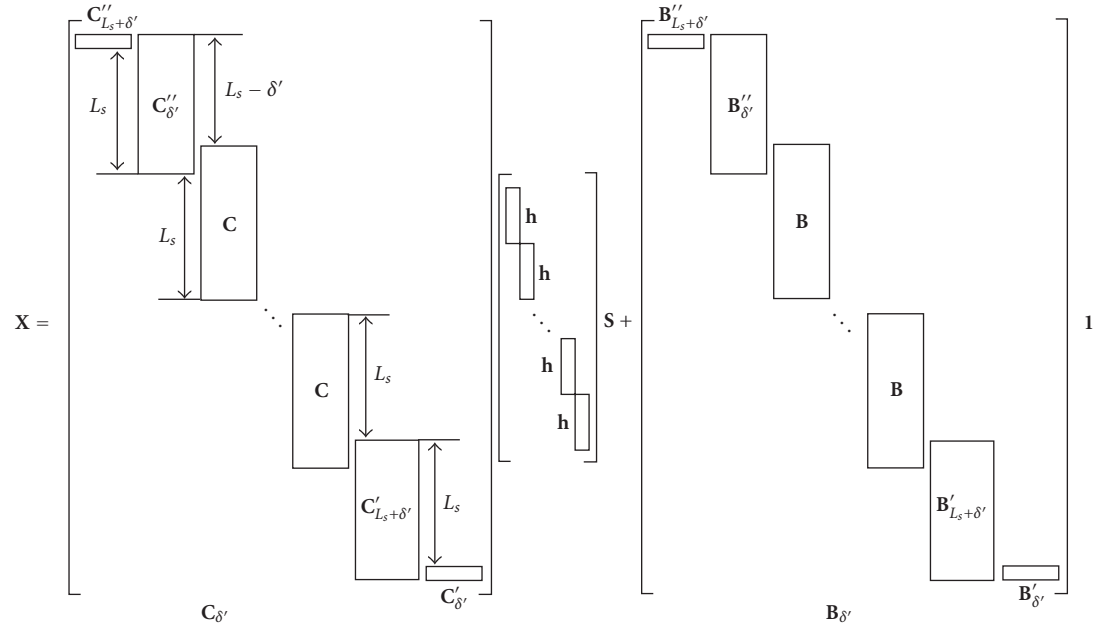

Figure 2: The data model structure of **X**.

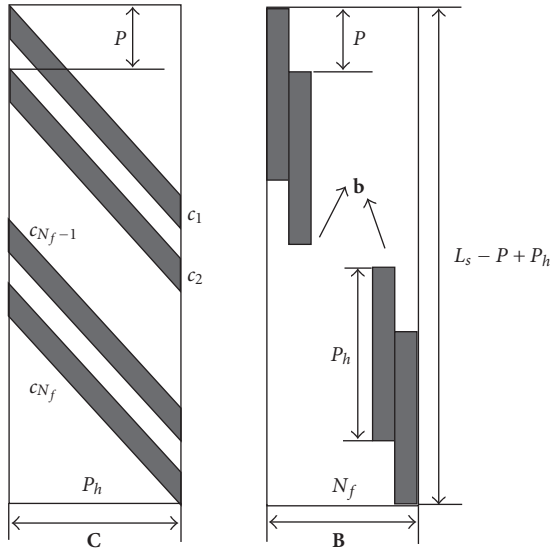

Figure 3: The structure of the code matrix **C** and the bias matrix **B**.

samples with different variances. The matrix $\mathbf{\Lambda}$, which collects the variances of each element in $\mathbf{N}_1$, is

$$\begin{aligned}\mathbf{\Lambda} &= E(\mathbf{N}_1 \odot \mathbf{N}_1) \\ &= \frac{N_0}{2}\left\{(\mathbf{H}'_{\delta'} + \mathbf{H}''_{\delta'})\mathbf{1}_{(MN_f+2N_f)\times N}\right. \\ &\qquad \left. + 2\mathbf{C}_{\delta'}(\mathbf{I}_{M+2} \otimes \mathbf{b})\mathbf{S}\right\} + \sigma_0^2 \mathbf{1}_{ML_s\times N},\end{aligned} \tag{17}$$

where $\mathbf{H}'_{\delta'}$ and $\mathbf{H}''_{\delta'}$ have exactly the same structure as $\mathbf{B}_{\delta'}$, only using $\mathbf{h}'$ and $\mathbf{h}''$ instead of $\mathbf{b}$. They all have the same periodic property, if multiplied by $\mathbf{1}$. Defining $\mathbf{h}'_f$ and $\mathbf{h}''_f$ to be the two $P \times 1$ vectors for one such period, we obtain

$$\mathbf{H}'_{\delta'}\mathbf{1}_{(MN_f+2N_f)\times N} = \mathbf{1}_{MN_f\times N} \otimes \mathbf{h}'_f, \tag{18}$$

$$\mathbf{H}''_{\delta'}\mathbf{1}_{(MN_f+2N_f)\times N} = \mathbf{1}_{MN_f\times N} \otimes \mathbf{h}''_f. \tag{19}$$

3. Detection

The first task of the receiver is to detect the existence of a signal. In order to separate the detection and the synchronization problems, we assume that the transmitted signal starts with a training sequence and assign the first segment of the training sequence to detection only. In this segment, we transmit all "+1" symbols and employ all "+1" codes. It is equivalent to sending only positive pulses for some time. This kind of training sequence bypasses the code and the symbol sequence synchronization. Therefore, we do not have to consider timing issues when we handle the detection problem. The drawback is the presence of spectral peaks as a result of the periodicity. It can be solved by employing a time hopping code for the frames. We omit this in our discussion for simplicity. It is also possible to use a signal structure other than TR signals for detection, such as a positive pulse training with an ED. Although the ED doubles the noise variance due to the squaring operation, the TR system wastes half of the energy to transmit the reference pulses. Therefore, they would have a similar detection performance for the same signal-to-noise ratio (SNR), that is, the ratio of the symbol energy to the noise power spectrum density. We keep the TR structure for detection in order to avoid additional hardware for the receiver.

In the detection process, we assume that the first training segment is $2M_1$ symbols long, and the observation window is

M_1 symbols long ($M_1 L_s = M_1 N_f P$ samples equivalently). We collect all the samples in the observation window, calculate a test statistic, and examine whether it exceeds a threshold. If not, we jump into the next successive observation window of M_1 symbols. The $2M_1$-symbol-long training segment makes sure that there will be at least one moment, at which the M_1-symbol-long observation window is full of training symbols. In this way, we speed up our search procedure by jumping M_1 symbols. Once the threshold is exceeded, we skip the next $2M_1$ symbols in order to be out of the first segment of the training sequence and we are ready to start the channel estimation and synchronization at the sample-level (see Section 4). There will be situations where the observation window only partially overlaps the signal. However, for simplicity, we will not take these cases into account, when we derive the test statistic. If these cases happen and the test statistic is larger than the threshold, we declare the existence of a signal, which is true. Otherwise, we miss the detection and shift to the next observation window, which is then full of training symbols giving us a second chance to detect the signal. Therefore, we do not have to distinguish the partially overlapped cases from the overall included case. We will derive the test statistic using only two hypotheses indicated below. But the evaluation of the detection performance will take all the cases into account.

3.1. Detection Problem Statement. Since we only have to tell whether the whole observation window contains a signal or not, the detection problem is simplified to a binary hypothesis test. We first define the $M_1 N_f P \times 1$ sample vector $\mathbf{x} = [\mathbf{x}_k^T, \mathbf{x}_{k+1}^T, \ldots, \mathbf{x}_{k+M_1-1}^T]^T$ with entries $x[n], n = (k-1)N_f P+1, (k-1)N_f P+2, \ldots, (k+M_1-1)N_f P$, which collects all the samples in the observation window. The hypotheses are as follows.

(1) $\mathcal{H}_0$: there is only noise. Under $\mathcal{H}_0$, according to the analysis from the previous section, $\mathbf{x}$ is modeled as

$$\mathbf{x} = \mathbf{n}_0, \tag{20}$$

$$\mathbf{x} \overset{a}{\sim} \mathcal{N}(0, \sigma_0^2 \mathbf{I}), \tag{21}$$

where $\mathbf{n}_0$ is the noise vector with entries $n_0[n], n = (k-1)N_f P + 1, (k-1)N_f P + 2, \ldots, (k + M_1 - 1)N_f P$, and $\overset{a}{\sim}$ indicates approximately distributed according to. The Gaussian approximation for $\mathbf{x}$ is valid based on the assumptions in the previous section.

(2) $\mathcal{H}_1$: signal with noise is occupying the whole observation window. Under $\mathcal{H}_1$, the data model (14) and the noise model (17) can be easily specified according to the all "+1" training sequence. We define $\mathbf{H}_{\delta'}$ having the same structure as $\mathbf{B}_{\delta'}$, only taking $\mathbf{h}$ instead of $\mathbf{b}$. It also has a period of P samples in each column, if multiplied by $\mathbf{1}$. Defining $\mathbf{h}_f$ to be the $P \times 1$ vector for one such period, we have

$$\mathbf{H}_{\delta'} \mathbf{1}_{(MN_f+2N_f)\times N} = \mathbf{1}_{MN_f \times N} \otimes \mathbf{h}_f. \tag{22}$$

By selecting $M = M_1$ and $N = 1$ for (14) and taking (16), (18), (19) and (22) into the model, the sample vector $\mathbf{x}$ can be decomposed as

$$\mathbf{x} = \mathbf{1}_{M_1 N_f} \otimes (\mathbf{h}_f + \mathbf{b}_f) + \mathbf{n}_1, \tag{23}$$

where the zero mean noise vector $\mathbf{n}_1$ has uncorrelated entries $n_1[n], n = (k-1)N_f P+1, (k-1)N_f P+2, \ldots, (k+M_1-1)N_f P$, and the variances of each element in $\mathbf{n}_1$ are given by

$$\begin{aligned} \boldsymbol{\lambda} &= E(\mathbf{n}_1 \odot \mathbf{n}_1) \\ &= \frac{N_0}{2} \mathbf{1}_{M_1 N_f} \otimes (\mathbf{h}'_f + \mathbf{h}''_f + 2\mathbf{b}_f) + \sigma_0^2 \mathbf{1}_{M_1 N_f P}. \end{aligned} \tag{24}$$

Due to the all "+1" training sequence, the impact of the IFI is to fold the aggregate channel response into one frame, so the frame energy remains constant. Normally, the channel correlation function is quite narrow, so $R(D, m) \ll R(0, m)$ and $R(2D, m) \ll R(0, m)$. Then, we can have the relation

$$\mathbf{h}'_f + \mathbf{h}''_f + 2\mathbf{b}_f \approx 4(\mathbf{h}_f + \mathbf{b}_f). \tag{25}$$

Defining the $P \times 1$ frame energy vector $\mathbf{z}_f = \mathbf{h}_f + \mathbf{b}_f$ with entries $z_f[i], i = 1, 2, \ldots, P$ and frame energy $\mathcal{E}_f = \mathbf{1}_P^T \mathbf{z}_f$, we can simplify $\mathbf{x}$ and $\boldsymbol{\lambda}$

$$\mathbf{x} = \mathbf{1}_{M_1 N_f} \otimes \mathbf{z}_f + \mathbf{n}_1, \tag{26}$$

$$\boldsymbol{\lambda} \approx 2N_0 \mathbf{1}_{M_1 N_f} \otimes \mathbf{z}_f + \sigma_0^2 \mathbf{1}_{M_1 N_f P}. \tag{27}$$

Based on the analysis above and the assumptions from the previous section, $\mathbf{x}$ can still be assumed as a Gaussian vector in agreement with [23]

$$\mathbf{x} \overset{a}{\sim} \mathcal{N}(\mathbf{1}_{M_1 N_f} \otimes \mathbf{z}_f,\ \mathrm{diag}(\boldsymbol{\lambda})), \tag{28}$$

where $\mathrm{diag}(\mathbf{a})$ indicates a square matrix with $\mathbf{a}$ on the main diagonal and zeros elsewhere.

3.2. Detector Derivation. The test statistic is derived using $\mathcal{H}_0$ and $\mathcal{H}_1$. It is suboptimal, since it ignores other cases. But it is still useful as we have analyzed before. The *Neyman-Pearson* (NP) detector [26] decides $\mathcal{H}_1$ if

$$L(\mathbf{x}) = \frac{p(\mathbf{x}; \mathcal{H}_1)}{p(\mathbf{x}; \mathcal{H}_0)} > \gamma, \tag{29}$$

where γ is found by making the probability of false alarm P_{FA} to satisfy

$$P_{\mathrm{FA}} = \Pr\{L(\mathbf{x}) > \gamma;\ \mathcal{H}_0\} = \alpha. \tag{30}$$

The test statistic is derived by taking the stochastic properties of $\mathbf{x}$ under the two hypotheses into $L(\mathbf{x})$ (29) and eliminating constant values. It is given by

$$T(\mathbf{x}) = \sum_{i=1}^{P} \frac{z_f[i]}{\sigma_1^2[i]} \left\{ \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} \left(x[nP+i] + \frac{N_0}{\sigma_0^2} x^2[nP+i] \right) \right\}, \tag{31}$$

where $\sigma_1^2[i] = 2N_0 z_f[i] + \sigma_0^2$. A detailed derivation is presented in Appendix B. Then the threshold γ will be found to satisfy

$$P_{\mathrm{FA}} = \Pr\{T(\mathbf{x}) > \gamma;\, \mathcal{H}_0\} = \alpha. \tag{32}$$

Hence, for each observation window, we calculate the test statistic $T(\mathbf{x})$ and compare it with the threshold γ. If the threshold is exceeded, we announce that a signal is detected.

The test statistic not only depends on the noise knowledge σ_0^2 but also on the composite channel energy profile $z_f[i]$. All data samples make a weighted contribution to the test statistic, since they have different means and variances. The larger $z_f[i]/\sigma_0^2$ is, the heavier the weighting coefficient is. If we would like to employ $T(\mathbf{x})$, we have to know σ_0^2 and $z_f[i]$ first. Note that σ_0^2 can be easily estimated, when there is no signal transmitted. However, the estimation of the composite channel energy profile $z_f[i]$ is not as easy, since it appears in both the mean and the variance of $\mathbf{x}$ under $\mathcal{H}_1$.

3.3. Detection Performance Evaluation. Until now, the optimal detector for the earlier binary hypothesis test has been derived. The performance of this detector working under real circumstances has to be evaluated by taking all the cases into account. As we have described before, there are moments where the observation window partially overlays the signal. They can be modeled as other hypotheses $\mathcal{H}_j$, $j = 2,\ldots,M_1N_fP$. Applying the same test statistic $T(\mathbf{x})$ under these hypotheses including $\mathcal{H}_1$, the probability of detection is defined as

$$P_{D,j} = \Pr\{T(\mathbf{x}) > \gamma; \mathcal{H}_j\}, \quad j = 1,\ldots,M_1N_fP. \tag{33}$$

We would obtain $P_{D,1} > P_{D,j}$, $j = 2,\ldots,M_1N_fP$. Since the observation window collects the maximum signal energy under $\mathcal{H}_1$ and the test statistic is optimized to detect $\mathcal{H}_1$, it should have the highest possibility to detect the signal. Furthermore, if we miss the detection under $\mathcal{H}_j, j = 1,\ldots,M_1N_fP$, we still have a second chance to detect the signal with a probability of $P_{D,1}$ in the next observation window, recalling that the training sequence is $2M_1$ symbols long. Therefore, the total probability of detection for this testing procedure is $P_{D,j} + (1 - P_{D,j})P_{D,1}$, $j = 1,\ldots,M_1N_fP$, which is larger than $P_{D,1}$ and not larger than $P_{D,1} + (1 - P_{D,1})P_{D,1}$. Since all hypotheses $\mathcal{H}_j, j = 1,\ldots,M_1N_fP$ have equal probability, we can obtain that the overall probability of detection P_{D_o} for the detector $T(\mathbf{x})$ is

$$P_{D_o} = \frac{1}{M_1N_fP}\sum_{j=1}^{M_1N_fP}\{P_{D,j} + (1 - P_{D,j})P_{D,1}\}, \quad j = 1,\ldots,M_1N_fP, \tag{34}$$

where $P_{D,1} < P_{D_o} < P_{D,1} + (1 - P_{D,1})P_{D,1}$. Since the analytical evaluation of P_{D_o} is very complicated, we just derive the theoretical performance of $P_{D,1}$ under $\mathcal{H}_1$. In the simulations section, we will obtain the total P_{D_o} by Monte Carlo simulations and compare it with $P_{D,1}$ and $P_{D,1} + (1 - P_{D,1})P_{D,1}$, which can be used as boundaries for P_{D_o}.

A theoretical evaluation of $P_{D,1}$ is carried out by first analyzing the stochastic properties of $T(\mathbf{x})$. As $T(\mathbf{x})$ is composed of two parts, we can define

$$T_1(\mathbf{x}) = \sum_{i=1}^{P}\frac{z_f[i]}{\sigma_1^2[i]}\sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1}x[nP+i], \tag{35}$$

$$T_2(\mathbf{x}) = \sum_{i=1}^{P}\frac{z_f[i]}{\sigma_1^2[i]}\sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1}x^2[nP+i]. \tag{36}$$

Then we have

$$T(\mathbf{x}) = T_1(\mathbf{x}) + \frac{N_0}{\sigma_0^2}T_2(\mathbf{x}). \tag{37}$$

First, we have to know the probability density function (PDF) of $T(\mathbf{x})$. However, due to the correlation between the two parts, it can only be found in an empirical way by generating enough samples of $T(\mathbf{x})$ and making a histogram to depict the relative frequencies of the sample ranges. Therefore, we simply assume that $T_1(\mathbf{x})$ and $T_2(\mathbf{x})$ are uncorrelated, and $T(\mathbf{x})$ is a Gaussian random variable. The mean (variance) of $T(\mathbf{x})$ is the sum of the weighted means (variances) of the two parts. The larger the sample number M_1N_fP is, the better the approximation is, but also the longer the detection time is. There is a tradeoff. In summary, $T(\mathbf{x})$ follows a Gaussian distribution as follows:

$$T(\mathbf{x}) \overset{a}{\sim} \mathcal{N}\Big(E(T_1(\mathbf{x})) + \frac{N_0}{\sigma_0^2}E(T_2(\mathbf{x})), \quad \mathrm{var}(T_1(\mathbf{x})) + \frac{N_0^2}{\sigma_0^4}\mathrm{var}(T_2(\mathbf{x}))\Big). \tag{38}$$

The mean and the variance of $T_1(\mathbf{x})$ can be easily obtained based on the assumption that $\mathbf{x}$ is a Gaussian vector. The stochastic properties of $T_2(\mathbf{x})$ are much more complicated. More details are discussed in Appendix C. All the performance approximations are summarized in Table 1, where the function $Q(\cdot)$ is the right-tail probability function for a Gaussian distribution.

A special case occurs when $P = 1$, which means that one sample is taken per frame ($T_{\mathrm{sam}} = T_f$). For this case, where no oversampling is used, we have constant energy $\mathcal{E}_f$ and constant noise variance $\sigma_1^2 = 2N_0\mathcal{E}_f + \sigma_0^2$ for each frame. Then the weighting parameters for each sample in the detector would be exactly the same. We can eliminate them and simplify the test statistic to

$$T_1'(\mathbf{x}) = \sum_{n=(k-1)N_f+1}^{(k+M_1-1)N_f}x[n], \tag{39}$$

$$T_2'(\mathbf{x}) = \sum_{n=(k-1)N_f+1}^{(k+M_1-1)N_f}x^2[n], \tag{40}$$

$$T'(\mathbf{x}) = T_1'(\mathbf{x}) + \frac{N_0}{\sigma_0^2}T_2'(\mathbf{x}). \tag{41}$$

Table 1: Statistical Analysis and Performance Evaluation for Different Detectors, $P > 1$, $T_{\text{sam}} = T_f/P$.

		$T_1(\mathbf{x})$	$T_2(\mathbf{x})$	$T(\mathbf{x})$
$\mathcal{H}_0$	μ	$\mu_{T_{1,0}} = 0$	$\mu_{T_{2,0}} = M_1 N_f {\sigma_0}^2 \sum_{i=1}^{P} \frac{z_f[i]}{\sigma_1^2[i]}$	$\mu_{T_0} = \mu_{T_{1,0}} + \frac{N_0}{\sigma_0^2}\mu_{T_{2,0}}$
	σ^2	$\sigma^2_{T_{1,0}} = M_1 N_f {\sigma_0}^2 \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^4[i]}$	$\sigma^2_{T_{2,0}} = 2M_1 N_f {\sigma_0}^4 \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^4[i]}$	$\sigma^2_{T_0} = \sigma^2_{T_{1,0}} + \frac{N_0^2}{\sigma_0^4}\sigma^2_{T_{2,0}}$
$\mathcal{H}_1$	μ	$\mu_{T_{1,1}} = M_1 N_f \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^2[i]}$	$\mu_{T_{2,1}} = M_1 N_f \sum_{i=1}^{P} z_f[i]\left(1 + \frac{z_f^2[i]}{\sigma_1^2[i]}\right)$	$\mu_{T_1} = \mu_{T_{1,1}} + \frac{N_0}{\sigma_0^2}\mu_{T_{2,1}}$
	σ^2	${\sigma^2}_{T_{1,1}} = M_1 N_f \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^2[i]}$	$\sigma^2_{T_{2,1}} = 2M_1 N_f \sum_{i=1}^{P} z_f^2[i]\left(1 + \frac{2z_f^2[i]}{\sigma_1^2[i]}\right)$	$\sigma^2_{T_1} = \sigma^2_{T_{1,1}} + \frac{N_0^2}{\sigma_0^4}\sigma^2_{T_{2,1}}$
P_{FA}		$Q\left(\frac{\gamma_1}{\sigma_{T_{1,0}}}\right) = \alpha$	$Q\left(\frac{\gamma - \mu_{T_{2,0}}}{\sigma_{T_{2,0}}}\right) = \alpha$	$Q\left(\frac{\gamma - \mu_{T_0}}{\sigma_{T_0}}\right) = \alpha$
γ		$\gamma_1 = \sigma_{T_{1,0}} Q^{-1}(\alpha)$	$\gamma_2 = \sigma_{T_{2,0}} Q^{-1}(\alpha) + \mu_{T_{2,0}}$	$\gamma = \sigma_{T_0} Q^{-1}(\alpha) + \mu_{T_0}$
$P_{D,1}$		$Q\left(\frac{\gamma_1 - \mu_{T_{1,1}}}{\sigma_{T_{1,1}}}\right)$	$Q\left(\frac{\gamma_2 - \mu_{T_{2,1}}}{\sigma_{T_{2,1}}}\right)$	$Q\left(\frac{\gamma - \mu_{T_1}}{\sigma_{T_1}}\right)$

Therefore, $T_2'(\mathbf{x})/\sigma_0^2$ will follow a central Chi-squared distribution under $\mathcal{H}_0$, and $T_2'(\mathbf{x})/\sigma_1^2$ will follow a noncentral Chi-squared distribution under $\mathcal{H}_1$. We calculate the threshold for $T_2'(\mathbf{x})$ as

$$\gamma_2' = {\sigma_0}^2 Q^{-1}_{\chi^2_{M_1 N_f}}(\alpha), \tag{42}$$

and the probability of detection under $\mathcal{H}_1$ as

$$P_{D,1} = Q_{\chi^2_{M_1 N_f}(M_1 N_f \mathcal{E}_f^2/\sigma_1^2)}\left(\frac{\gamma_2'}{\sigma_1^2}\right), \tag{43}$$

where the functions $Q_{\chi_v^2}(x)$ and $Q_{\chi_\lambda^2(\lambda)}(x)$ are the right-tail probability functions for a central and noncentral Chi-squared distribution, respectively. The statistics of $T_1'(\mathbf{x})$ can be obtained by taking $P = 1$, $z_f[i] = \mathcal{E}_f$, and $\sigma_1^2[i] = \sigma_1^2$ into Table 1, and multiplying the means with $\sigma_1^2/\mathcal{E}_f$ and the variances with $\sigma_1^4/\mathcal{E}_f^2$. As a result, the threshold γ_1' for $T_1'(\mathbf{x})$ is $\sqrt{M_1 N_f \sigma_0^2}\, Q^{-1}(\alpha)$, which can be easily obtained. The $P_{D,1}$ of $T'(\mathbf{x})$ could be evaluated in the same way as $T(\mathbf{x})$ in Table 1.

The theoretical contributions of $T_1'(\mathbf{x})$ and $T_2'(\mathbf{x})$ to $T'(\mathbf{x})$ are assessed in Figure 4. The simulation parameters are set to $M_1 = 8$, $N_f = 15$, $T_f = 30$ ns, $T_p = 0.2$ ns, and $B \approx 2/T_p$. For the definition of E_p/N_0, we refer to Section 5. The detector based on $T_1'(\mathbf{x})$ (dashed lines) plays a key role in the performance of the detector based on $T'(\mathbf{x})$ (solid lines) under $\mathcal{H}_1$. For low SNR, they are almost the same, since $T_1'(\mathbf{x})$ can be directly derived by ignoring the signal-noise cross-correlation term in the noise variance under $\mathcal{H}_1$. There is a small difference between them for medium SNRs. $T_2'(\mathbf{x})$ (dotted lines) has a performance loss of about 4 dB compared to $T'(\mathbf{x})$. Thanks to the ultra-wide bandwidth of the signal, the weighting parameter $N_0/{\sigma_0}^2$ greatly reduces the influence of $T_2'(\mathbf{x})$ on $T'(\mathbf{x})$. It enhances the performance of $T'(\mathbf{x})$ slightly in the medium SNR range. According to these simulation results and the impact of the weighting parameter N_0/σ_0^2, we can employ $T_1'(\mathbf{x})$ instead of $T'(\mathbf{x})$. It has a much lower calculation cost and almost the same performance as $T'(\mathbf{x})$.

Furthermore, the influence of the oversampling rate P to the $P_{D,1}$ of $T(\mathbf{x})$ can be ignored because the oversampling only affects the performance of $T_2(\mathbf{x})$, which only has a very small influence on $T(\mathbf{x})$. Therefore, the impact of the oversampling can be neglected. In Section 5, we will evaluate the $P_{D,1}$ of $T(\mathbf{x})$ using the IEEE UWB channel model by a quasi-analytical method and also by Monte Carlo simulations. Based on the simulation results in this section, we can predict that for small P ($P > 1$), the $P_{D,1}$ for $T(\mathbf{x})$ will be more or less the same as the $P_{D,1}$ for $T'(\mathbf{x})$ or $T_1'(\mathbf{x})$.

4. Channel Estimation, Synchronization, and Equalization

After successful signal detection, we can start the channel estimation and synchronization phase. The sample-level synchronization finds out the symbol boundary (estimates the unknown offset δ), and the result can later on be used for symbol-level synchronization to acquire the header. This two-stage synchronization strategy decomposes a two-dimensional search into two one-dimensional searches, reducing the complexity. The channel estimates and the timing information can be used for the equalizer construction. Finally, the demodulated symbols can be obtained.

4.1. Channel Estimation

4.1.1. Bias Estimation. As we have seen in the asynchronous data model, the bias term is undesired. It does not have any useful information, but it disturbs the signal. We will show that this bias seriously degrades the channel estimation performance later on. The second segment of the training sequence consists of "+1, −1" symbol pairs employing a random code. The total length of the second segment should be $M_1 + 2N_s$ symbols, which includes the budget for jumping $2M_1$ symbols after the detection. The "+1, −1" symbol pairs can be used for bias estimation as well as channel estimation. Since the bias is independent of the data symbols and the

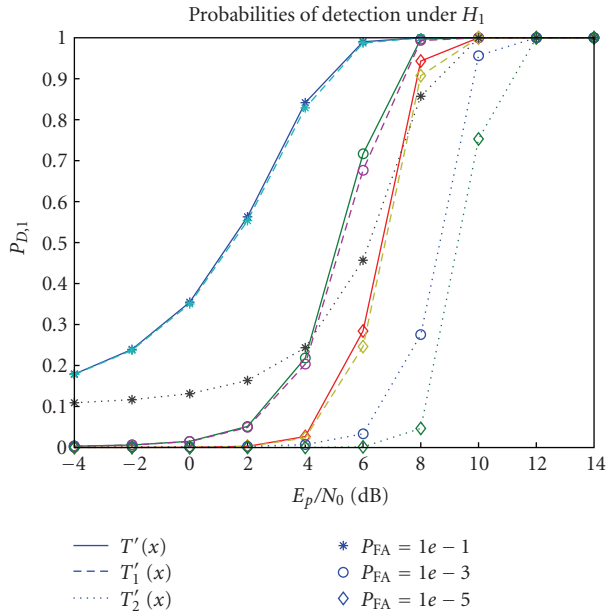

FIGURE 4: Performance comparison between $T'(\mathbf{x})$ and its components $T'_1(\mathbf{x})$ and $T'_2(\mathbf{x})$.

useful signal part has zero mean, due to the "+1, −1" training symbols, we can estimate the $L_s \times 1$ bias vector of one symbol, $\mathbf{b}_s = \mathbf{1}_{N_f} \otimes \mathbf{b}_f$, as

$$\hat{\boldsymbol{b}}_s = \frac{1}{2N_s}\left[\begin{array}{cccc} \mathbf{x}_k & \mathbf{x}_{k+1} & \cdots & \mathbf{x}_{k+2N_s-1} \end{array}\right]\mathbf{1}_{2N_s}. \tag{44}$$

4.1.2. Channel Estimation. To take advantage of the second segment of the training sequence, we stack the data samples as

$$\widetilde{\mathbf{X}} = \begin{bmatrix} \mathbf{x}_k & \mathbf{x}_{k+2} & \cdots & \mathbf{x}_{k+2N_s-2} \\ \mathbf{x}_{k+1} & \mathbf{x}_{k+3} & \cdots & \mathbf{x}_{k+2N_s-1} \end{bmatrix}, \tag{45}$$

which is equivalent to picking only odd columns of $\mathbf{X}$ in (14) with $M = 2$ and $N = 2N_s - 1$. As a result, each column depends on the same symbols, which leads to a great simplification of the decomposition in (14) as follows:

$$\begin{aligned} \widetilde{\mathbf{X}} = &\left[\left(\mathbf{C}'_{L_s+\delta'} + \mathbf{C}''_{L_s+\delta'}\right) \quad \left(\mathbf{C}'_{\delta'} + \mathbf{C}''_{\delta'}\right)\right](\mathbf{I}_2 \otimes \mathbf{h}) \\ &\times \left[\begin{array}{cc} -s_k & s_k \end{array}\right]^T \mathbf{1}_{N_s}^T + \mathbf{1}_{2\times N_s} \otimes \mathbf{b}_s + \widetilde{\mathbf{N}}_1, \end{aligned} \tag{46}$$

where $\widetilde{\mathbf{N}}_1$ is the noise matrix similarly defined as $\widetilde{\mathbf{X}}$. For simplicity, we only count the noise autocorrelation term with zero mean and variance σ_0^2 into $\widetilde{\mathbf{N}}_1$, where σ_0^2 can be easily estimated in the absence of a signal. Because we jump into this second segment of the training sequence after detecting the signal, we do not know whether the symbol s_k is "+1" or "−1". Rewriting (46) in another form leads to

$$\widetilde{\mathbf{X}} = \mathbf{C}_s \mathbf{h}_{ss\delta} \mathbf{1}_{N_s}^T + \mathbf{1}_{2\times N_s} \otimes \mathbf{b}_s + \widetilde{\mathbf{N}}_1, \tag{47}$$

where $\mathbf{C}_s$ is a known $2L_s \times 2L_s$ circulant code matrix, whose first column is $[c_1, \mathbf{0}_{P-1}^T, c_2, \mathbf{0}_{P-1}^T, \ldots, c_{N_f}, \mathbf{0}_{L_s+P-1}^T]^T$, and the vector $\mathbf{h}_{ss\delta}$ of length $2L_s$ blends the timing and the channel information, which contains two channel energy vectors with different signs, $s_k\mathbf{h}$ and $-s_k\mathbf{h}$, located according to δ as follows:

$$\begin{aligned} &\mathbf{h}_{ss\delta} \\ &= \begin{cases} \text{circshift}\left(\left[s_k\mathbf{h}^T, \mathbf{0}_{L_s-P_h}^T, -s_k\mathbf{h}^T, \mathbf{0}_{L_s-P_h}^T\right]^T, \delta\right), & \delta \neq 0, \\ \left[-s_k\mathbf{h}^T, \mathbf{0}_{L_s-P_h}^T, s_k\mathbf{h}^T, \mathbf{0}_{L_s-P_h}^T\right]^T, & \delta = 0, \end{cases} \end{aligned} \tag{48}$$

where circshift $(\mathbf{a}, n)$ circularly shifts the values in the vector $\mathbf{a}$ by $|n|$ elements (down if $n > 0$ and up if $n < 0$). According to (47) and assuming the channel energy has been normalized, the linear minimum mean square error (LMMSE) estimate of $\mathbf{h}_{ss\delta}$ then is

$$\hat{\mathbf{h}}_{ss\delta} = \mathbf{C}_s^H \left(\mathbf{C}_s\mathbf{C}_s^H + \frac{\sigma_0^2}{N_s}\mathbf{I}\right)^{-1} \frac{1}{N_s}(\widetilde{\mathbf{X}} - \mathbf{1}_{2\times N_s} \otimes \mathbf{b}_s)\mathbf{1}_{N_s}. \tag{49}$$

Defining

$$\hat{\mathbf{h}}_{s\delta} = \frac{\left[\hat{\mathbf{h}}_{ss\delta}(1:L_s) - \hat{\mathbf{h}}_{ss\delta}(L_s+1:2L_s)\right]}{2}, \tag{50}$$

where $\mathbf{a}(m:n)$ refers to element m through n of $\mathbf{a}$, we can obtain a symbol-long LMMSE channel estimate as

$$\hat{\mathbf{h}}_\delta = |\hat{\mathbf{h}}_{s\delta}|. \tag{51}$$

According to a property of circulant matrices, $\mathbf{C}_s$ can be decomposed as $\mathbf{C}_s = \mathcal{F}\boldsymbol{\Omega}\mathcal{F}^H$, where $\mathcal{F}$ is the normalized DFT matrix of size $2L_s \times 2L_s$, and $\boldsymbol{\Omega}$ is a diagonal matrix with the frequency components of the first row of $\mathbf{C}_s$ on the diagonal. Hence, the matrix inversion in (49) can be simplified dramatically. Observing that $\mathbf{C}_s^H(\mathbf{C}_s\mathbf{C}_s^H + (\sigma_0^2/N_s)\mathbf{I})^{-1}$ is a circulant matrix, the bias term actually does not have to be removed in (49), since it is implicitly removed when we calculate (50). Therefore, we do not have to estimate the bias term explicitly for channel estimation and synchronization.

When the SNR is high, $\|\mathbf{C}_s\mathbf{C}_s^H\|_F \gg \|(\sigma_0^2/N_s)\mathbf{I}\|_F$, (49) can be replaced by

$$\hat{\mathbf{h}}_{ss\delta} = \frac{1}{N_s}\mathcal{F}\boldsymbol{\Omega}^{-1}\mathcal{F}^H(\widetilde{\mathbf{X}} - \mathbf{1}_{2\times N_s} \otimes \mathbf{b}_s)\mathbf{1}_{N_s}. \tag{52}$$

It is a least squares (LS) estimator and equivalent to a deconvolution of the code sequence in the frequency domain. On the other hand, when the SNR is low, $\|\mathbf{C}_s\mathbf{C}_s^H\|_F \ll \|(\sigma_0^2/N_s)\mathbf{I}\|_F$, (49) becomes

$$\hat{\mathbf{h}}_{ss\delta} = \frac{1}{\sigma_0^2}\mathcal{F}\boldsymbol{\Omega}^H\mathcal{F}^H(\widetilde{\mathbf{X}} - \mathbf{1}_{2\times N_s} \otimes \mathbf{b}_s)\mathbf{1}_{N_s}, \tag{53}$$

which is equivalent to a matched filter (MF). The MF can also be processed in the frequency domain. The LMMSE estimator in (49), the LS estimator in (52), and the MF in (53) all have a similar computational complexity. However, for the LMMSE estimator, we have to estimate σ_0^2 and the channel energy.

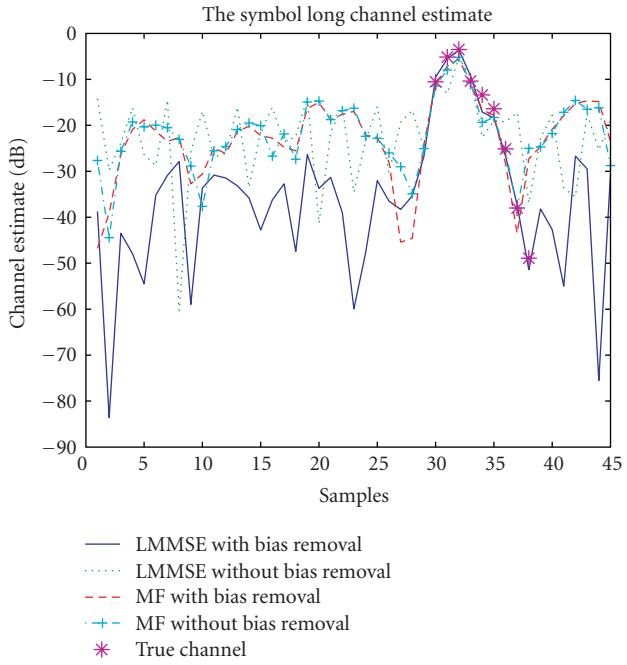

FIGURE 5: The symbol-long channel estimate $\hat{\mathbf{h}}_\delta$ with bias removal and $|\hat{\mathbf{h}}_{ss\delta}(1:L_s)|$ without bias removal, when SNR is 18 dB.

As an example, we show the performance of these channel estimates under high SNR conditions (the simulation parameters can be found in Section 5). Figure 5 indicates the symbol-long channel estimate $\hat{\mathbf{h}}_\delta$ with bias removal (implicitly obtained) and $|\hat{\mathbf{h}}_{ss\delta}(1:L_s)|$ without bias removal, where $\hat{\mathbf{h}}_{ss\delta} = \mathbf{C}_s^H(\mathbf{C}_s\mathbf{C}_s^H + (\sigma_0^2/N_s)\mathbf{I})^{-1}(1/N_s)\tilde{\mathbf{X}}\mathbf{1}_{N_s}$ for the LMMSE and $\hat{\mathbf{h}}_{ss\delta} = (1/\sigma_0^2)\mathcal{F}\boldsymbol{\Omega}^H\mathcal{F}^H\tilde{\mathbf{X}}\mathbf{1}_{N_s}$ for the MF. When the SNR is high, the LMMSE estimator is expected to have a similar performance as the LS estimator. Thus, we omit the LS estimator in Figure 5. The MF for $\hat{\mathbf{h}}_\delta$ (dashed line) has a higher noise floor than the LMMSE estimator for $\hat{\mathbf{h}}_\delta$ (solid line), since its output is the correlation of the channel energy vector with the code autocorrelation function. The bias term lifts the noise floor of the channel estimate resulting from the LMMSE estimator (dotted line) and distorts the estimation, while it does not have much influence on the MF (dashed line with + markers). The stars in the figure present the real channel parameters as a reference. The position of the highest peak for each curve in Figure 5 indicates the timing information and the area around this highest peak is the most interesting part, since it shows the estimated channel energy profile. Although the LMMSE estimator without bias suppresses the estimation errors over the whole symbol period, it has a similar performance as all the other estimators in the interesting part.

4.2. Sample-Level Synchronization. The channel estimate $\hat{\mathbf{h}}_\delta$ has a duration of one symbol. But we know that the true channel will generally be much shorter than the symbol period. We would like to detect the part that contains most of the channel energy and cut out the other part in order to be robust against noise. This basically means that we have to estimate the unknown timing δ. Define the search window length as L_w in terms of the number of samples ($L_w > 1$). The optimal length of the search window depends on the channel energy profile and the SNR. We will show the impact of different window lengths on the estimation of δ in the next section. Define $\hat{\mathbf{h}}_{w\delta} = [\hat{\mathbf{h}}_{s\delta}^T, -\hat{\mathbf{h}}_{s\delta}^T(1:L_w-1)]^T$, and define $\hat{\delta}$ as the δ estimate as follows:

$$\hat{\delta} = \underset{\delta}{\operatorname{argmax}} \left| \sum_{n=\delta+1}^{\delta+L_w} \hat{\mathbf{h}}_{w\delta}(n) \right|. \tag{54}$$

This is motivated as follows. According to the definition of $\hat{\mathbf{h}}_{s\delta}$, when $\delta > L_s - P_h$, $\hat{\mathbf{h}}_{s\delta}$ will contain channel information partially from $s_k\mathbf{h}$ and partially from $-s_k\mathbf{h}$, which have opposite signs. In order to estimate δ, we circularly shift the search window to check all the possible sample positions in $\hat{\mathbf{h}}_{s\delta}$ and find the position where the search window contains the maximum energy. If we do not adjust the signs of the two parts, the δ estimation will be incorrect when the real δ is larger than $L_s - P_h$. This is because the two parts will cancel each other, when both of them are encompassed by the search window. That is the reason why we construct $\hat{\mathbf{h}}_{w\delta}$ by inverting the sign of the first $L_w - 1$ samples in $\hat{\mathbf{h}}_{s\delta}$ and attaching them to the end of $\hat{\mathbf{h}}_{s\delta}$. Moreover, the estimator (54) benefits from averaging the noise before taking the absolute value.

4.3. Equalization and Symbol-Level Synchronization. Based on the channel estimate $\hat{\mathbf{h}}_\delta$ and the timing estimate $\hat{\delta}$, we select a part of $\hat{\mathbf{h}}_\delta$ to build three different kinds of equalizers. Since the MF equalizer cannot handle IFI and ISI, we only select the first P samples (the frame length in terms of number of samples) of circshift($\hat{\mathbf{h}}_\delta, -\hat{\delta}$) as $\hat{\mathbf{h}}_p$. The code matrix $\mathbf{C}$ is specified by assigning $P_h = P$. The estimated bias $\hat{\mathbf{b}}_s$ can be used here. We skip the first $\hat{\delta}$ data samples and collect the rest of the data samples in a matrix $\mathbf{X}_{\hat{\delta}}$ of size $L_s \times N$ as in the data model (14) but with $M = 1$. Therefore, the MF equalizer is constructed as

$$\hat{\mathbf{s}}^T = \operatorname{sign}\left\{\left(\mathbf{C}\hat{\mathbf{h}}_p\right)^T\left(\mathbf{X}_{\hat{\delta}} - \mathbf{1}_{1\times N}\otimes\hat{\mathbf{b}}_s\right)\right\}, \tag{55}$$

where $\hat{\mathbf{s}}$ is the estimated symbol vector. Moreover, we also construct a zero-forcing (ZF) equalizer and an LMMSE equalizer by replacing $\mathbf{h}$ with $\hat{\mathbf{h}}$, which collects the first $\hat{P}_h$ samples (the channel length estimate in terms of number of samples) of circshift($\hat{\mathbf{h}}_\delta, -\hat{\delta}$), and using $\hat{\delta}' = (L_s - \hat{\delta}) \bmod L_s$ in the data model (14). The channel length estimate $\hat{P}_h$ could be obtained by setting a threshold (e.g., 10% of the maximum value of $\hat{\mathbf{h}}_\delta$) and counting the number of samples beyond it in $\hat{\mathbf{h}}_\delta$. These equalizers can resolve the IFI and the ISI to achieve a better performance at the expense of a higher computational complexity. The estimated bias $\hat{\mathbf{b}}_s$ can also be used. We collect the samples in a data matrix $\mathbf{X}$ of size $2L_s \times N$ similar as the data model (14) with $M = 2$. Then the ZF equalizer gives

$$\hat{\mathbf{S}} = \operatorname{sign}\left\{\left(\mathbf{C}_{\hat{\delta}'}\left(\mathbf{I}_4\otimes\hat{\mathbf{h}}\right)\right)^\dagger\left(\mathbf{X} - \mathbf{1}_{2\times N}\otimes\hat{\mathbf{b}}_s\right)\right\}, \tag{56}$$

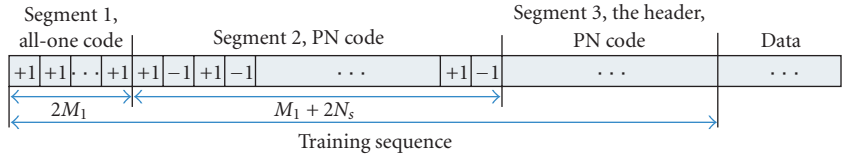

Figure 6: The signal structure for training sequence.

and the LMMSE equalizer gives

$$\hat{\mathbf{S}} = \operatorname{sign}\left\{\left(\hat{\mathbf{\Phi}}^H\hat{\mathbf{\Phi}} + \sigma_0^2\mathbf{I}_4\right)^{-1}\hat{\mathbf{\Phi}}^H\left(\mathbf{X} - \mathbf{1}_{2\times N}\otimes\hat{\mathbf{b}}_s\right)\right\}, \quad (57)$$

where $\hat{\mathbf{\Phi}} = \mathbf{C}_{\hat{\delta}'}(\mathbf{I}_4 \otimes \hat{\mathbf{h}})$. $\hat{\mathbf{S}}$ is a $4 \times N$ symbol matrix. We can choose either the second or the third row of $\hat{\mathbf{S}}$ as the demodulated symbol sequence.

Until now, the sample-level synchronization confirms the boundaries of the symbols. However, it is not able to explore the boundary of the training header, since the second segment of the training sequence just employs pairs of "+1,−1" symbols. After the sample-level synchronization, the demodulation is triggered. The third segment of the training sequence is a known training symbol pattern. Once we find the matching symbol pattern, we can distinguish the training header. Symbol-level synchronization is then accomplished. To summarize the training segments used in every stage, the overall structure of the training sequence is shown in Figure 6.

5. Simulation Results

We evaluate the performance of different detectors and the performance of different combinations of channel estimation and equalization schemes for a single user and single delay TR-UWB system. We use a Gaussian second derivative pulse, which is 0.2 ns wide. The delay interval D between two pulses in a doublet is 4 ns. The first segment of the training sequence is $2M_1 = 16$ symbols long, all of which are composed of positive pulses. Hence, the observation window includes $M_1 = 8$ symbols. The second segment of the training sequence has $M_1 + 2N_s = 38$ symbols and employs a pseudonoise (PN) code sequence. The code length N_f is 15. The frame-period T_f is 30 ns. The IEEE UWB channel model CM3 [27] is employed and truncated to 90 ns, which represents a NLOS channel. The oversampling rate P is 3, which results in $T_{\text{sam}} = 10$ ns. We define E_p/N_0 as the received aggregate pulse energy to noise ratio with $E_p = \int |h(t)|^2 dt$, where $h(t)$ represents the composite channel impulse response including pulse shaping and antenna effects as we have explained before (see Section 2.1). The system sampling rate is 50 GHz for Matlab simulations.

The test statistics $T(\mathbf{x})$ in (37) and $T_1'(\mathbf{x})$ in (39) are assessed in both a theoretical way by using the results in Table 1 and an experimental way by running Monte Carlo simulations. Figure 7 shows the probability of detection $P_{D,1}$ for the test statistics. The theoretical $P_{D,1}$ of $T(\mathbf{x})$ with $P = 3$ is evaluated in a quasianalytical method. We generate 100 IEEE CM3 channel realizations, and for each channel realization, we use Table 1 to evaluate its $P_{D,1}$ performance and average the obtained $P_{D,1}$'s. In the experimental way, we still employ 100 IEEE CM3 channel realizations. For each realization, we generate 1000 test statistics to compare with the threshold and count the probability of detection. In order to evaluate the detection performance, we divide the SNR into three ranges. For example, when $P_{\text{FA}} = 0.1$, the low SNR range is below 0 dB, the medium range is from 0 dB to 6 dB, and the high SNR range is above 6 dB. According to Figure 7, the $P_{D,1}$ of $T(\mathbf{x})$ with $P = 3$ (solid line with $*$ markers) and the $P_{D,1}$ of $T_1'(\mathbf{x})$ (dash-dotted line with $*$ markers) are similar in the low and high SNR ranges. But in the medium range, $T(\mathbf{x})$ with $P = 3$ outperforms $T_1'(\mathbf{x})$ for about 5% $\sim$ 10%. For $P_{\text{FA}} = 10^{-3}$ and $P_{\text{FA}} = 10^{-5}$, the performance differences for these test statistics are large in the SNR range from 2 dB to 8 dB. $T(\mathbf{x})$ (solid lines with $\circ$ or $\diamond$ markers) can have a detection probability as high as 20% more than $T_1'(\mathbf{x})$ (dash-dotted lines with $\circ$ or $\diamond$ markers) under $\mathcal{H}_1$. However, when the test statistic $T(\mathbf{x})$ is employed, we have to estimate the channel energy profile first. On the other hand, if we use the test statistic $T_1'(\mathbf{x})$, we only have to sum up the samples, which is easy to implement. But these results are only the detection probabilities under $\mathcal{H}_1$, which are used as boundaries for the overall performance under real circumstances.

As we have mentioned before, $P_{D,1}$ and $P_{D,1} + (1 - P_{D,1})P_{D,1}$ can be used as a lower boundary and an upper boundary for the overall P_{D_o}, respectively. We run Monte Carlo simulations to evaluate the P_{D_o} under real circumstances. For each run, the timing offset is randomly generated following a uniform distribution in the range of M_1 symbols, meanwhile the channel realization remains the same in order to exclude the channel influence in the multihypotheses case. In the detection procedure, once the first detection fails, we jump into the next observation window. When the second detection fails again, we declare a missed detection. The simulation results are shown in Figure 8. The P_{D_o}'s of $T(\mathbf{x})$ with $P = 3$ (solid lines) lie between two boundaries: the upper boundaries (dashed lines) and the lower boundaries (dotted lines), and these boundaries are getting tighter as the P_{FA}'s are getting smaller. The P_{D_o}'s of $T_1'(\mathbf{x})$ (dash-dotted lines) are a bit higher than the P_{D_o}'s of $T(\mathbf{x})$. Especially for $P_{\text{FA}} = 10^{-3}$, around SNR = 6 dB, the P_{D_o} of $T_1'(\mathbf{x})$ (dash-dotted line with $\circ$ markers) is 5% larger than the P_{D_o} of $T(\mathbf{x})$ (solid line with $\circ$ markers). That is because $T(\mathbf{x})$ weights each sample only based on two hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$. The weighting coefficients are not optimal for other hypotheses.

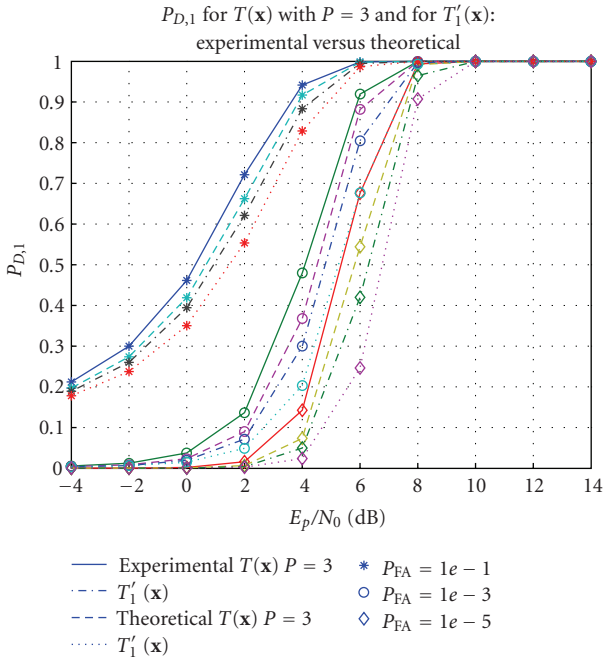

FIGURE 7: Experimental and theoretical $P_{D,1}$ performance comparison for $T(\mathbf{x})$ with $P = 3$ and $T_1'(\mathbf{x})$.

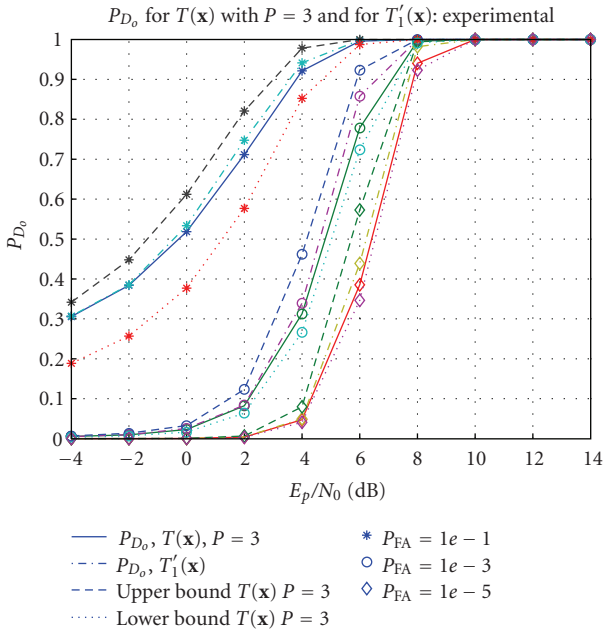

FIGURE 8: Experimental P_{D_o} for $T(\mathbf{x})$ with $P = 3$ and $T_1'(\mathbf{x})$.

The noise samples may be mistakenly weighted heavily under real circumstances. On the other hand, $T_1'(\mathbf{x})$ accumulates all the frame samples in the observation window, which is equivalent to equally weighting. According to these results, we can employ $T_1'(\mathbf{x})$ because of its simplicity and similar performance as $T(\mathbf{x})$.

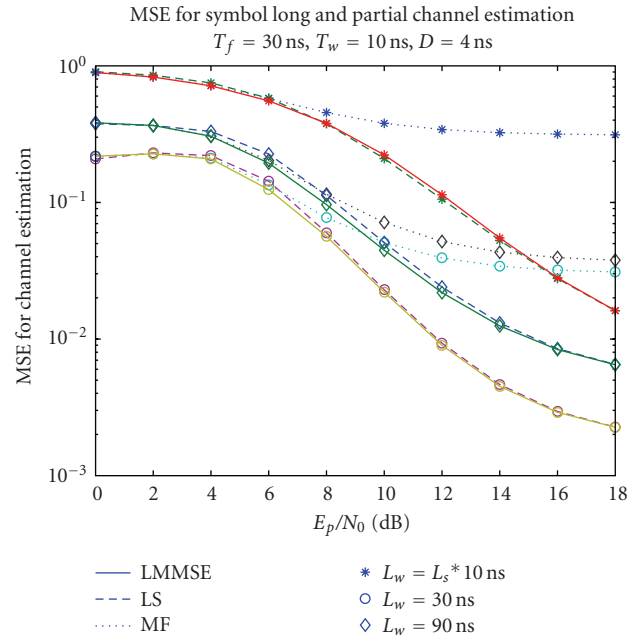

FIGURE 9: MSE performance for channel estimation with different lengths.

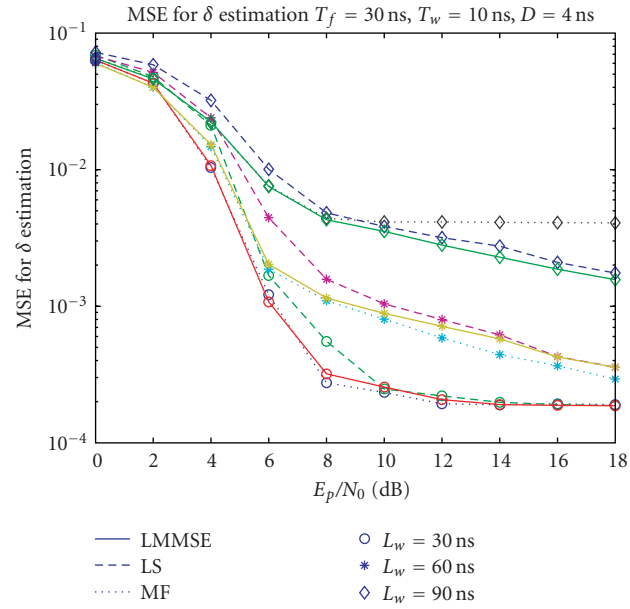

FIGURE 10: MSE performance for δ estimation with various L_w's.

500 Monte Carlo runs are used to evaluate the mean squared error (MSE) of $\hat{\mathbf{h}}_\delta$ versus SNR. In each run, the timing offset and the channel are randomly generated. The results for the symbol-long estimates and the L_w-long estimates assuming perfect timing are shown in Figure 9. The MF curves (dotted lines) always have the highest noise floor, since the MF output is the convolution of the channel energy vector with the code autocorrelation function.

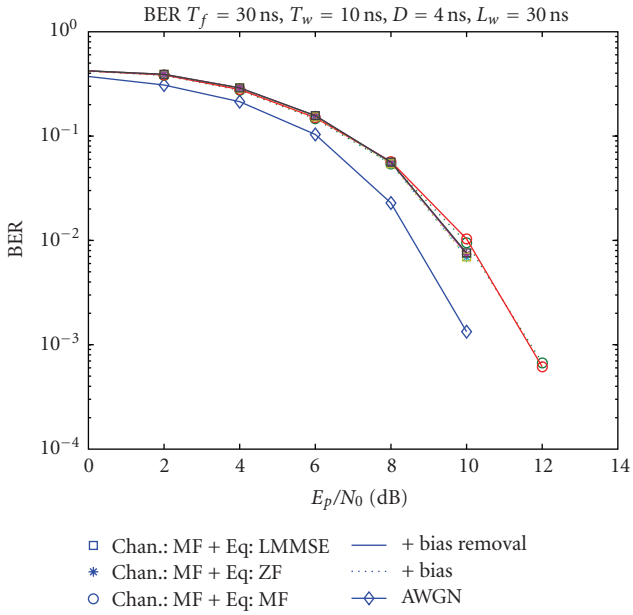

Figure 11: BER performance for CM3.

The performance gap for symbol-long estimates between the LS/LMMSE (dashed lines/solid lines) estimator and the MF is large. When we concentrate on the channel estimates in a limited range, such as 30 ns (lines with ∘ markers) and 90 ns (lines with ◊ markers), the gap between the MF and the LS/LMMSE estimator is smaller. The normalized MSE $E[|(\hat{\delta} - \delta)/L_s|^2]$ for δ estimation is also assessed with different values of L_w based on different channel estimators. From Figure 10, we see that the δ estimates based on MF (dotted lines), LS (dashed lines), and LMMSE (solid lines) channel estimates with the same L_w have similar performance, and $L_w = 30$ ns is the best choice among all. The MSE for δ with $L_w = 30$ ns (lines with ∘ markers) is saturated after the SNR reaches 10 dB. This is because we use NLOS channels, where the first path may not be the strongest and there is always remaining a fractional timing offset ϵ. Meanwhile the differences of the MSE for channel estimation with a 90-nanosecond range based on different methods (lines with ◊ markers) are quite small around 10 dB in Figure 9, which will be employed to construct the equalizer. As a result, we choose the MF as the channel estimator.

Furthermore, combinations of the MF channel estimator with different equalizers are investigated. We employ $L_w = 30$ ns for synchronization. Figure 11 shows the BER performance. The BER performance for the MF equalizer (lines with ∘ markers) approaches 0 after 12 dB, while the performances for the ZF (lines with ∗ markers) and the LMMSE equalizers (lines with □ markers) approach 0 after 10 dB. Hence, the MF equalizer is 2 dB worse than the ZF and the LMMSE equalizer, and all of them employ 90 ns long channel estimates. The curves of the ZF equalizer and the LMMSE equalizer overlay each other. The bias does not have much impact on them. They have almost the same performance. As a result, the optimal combination considering cost and performance would be an MF channel estimator with a ZF equalizer. According to the results above, we can remark that the IFI after the integrate-and-dump is not so serious in our simulation setup, since the channel energy attenuates exponentially and one frame contains most of the energy. The performance differences of different equalizers are not so obvious. However, the LMMSE estimator has the potential to handle more serious IFI and ISI. The effects of the bias on the BER performance can be ignored, but they have to be taken into account for the channel estimation (done implicitly, see Section 4.1). When we want to shorten the frame length to achieve a higher data rate, more interference will be generated. We then need a more accurate data model to handle this interference.

6. Conclusions

We have proposed a complete solution for signal detection, channel estimation, synchronization, and equalization in a TR-UWB system. The scheme is based on a data model, which takes IPI, IFI, and ISI into account and releases the frame time requirements to allow for higher data rate communications. Several detectors based on a specific training scheme are derived and assessed. We find that the simple detector, which sums up all the samples in the observation window and compares the result with a threshold, gives a good balance between performance and cost. Moreover, the joint channel and timing estimation is achieved in three different ways. The property of the circulant matrix in the data model is exploited to reduce the complexity of the algorithms. Then a two-stage synchronization strategy is proposed to first achieve sample-level synchronization and later to achieve symbol-level synchronization. Last but not least, three kinds of equalizers are derived. We evaluate different combinations of channel estimation and equalization schemes using the IEEE UWB channel model CM3, which shows that the TR-UWB system can be implemented with low cost and achieves moderate data rate communications.

Appendices

A. Noise Analysis

The noise autocorrelation term $n_0[n]$ is

$$n_0[n] = \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} n(t)n(t+D)dt, \tag{A.1}$$

where $n(t)$ is band limited AWGN, and its autocorrelation function is $R_n(\tau) = E[n(t)n(t-\tau)] = N_0 B\text{sinc}(2B\tau)$. Therefore, $n_0[n]$ has approximately zero mean, as a result of $R_n(D) \approx 0$ based on the assumption $D \gg 1/B$. According to

the Gaussian joint variable theorem [28, 29], its variance can be derived as

$$\begin{aligned}
&\operatorname{var}(n_0[n]) \\
&\quad \approx E\left[n_0^2[n]\right] \\
&\quad \approx \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \Big[R_n^2(t-u) + R_n(t-u-D) \\
&\qquad\qquad \times R_n(t+D-u)\Big]\, dt\, du.
\end{aligned} \tag{A.2}$$

The second term is the product of two sinc functions offset by $2D$, which is approximately zero by using the property of sinc functions saying that $\operatorname{sinc}(2B\tau)\operatorname{sinc}(2B(\tau+\Delta)) \approx \operatorname{sinc}^2(2B\tau)\delta(\Delta)$, where $\delta(\Delta)$ is the Kronecker delta. Recalling $R_n(D) \approx 0$ and $T_{\text{sam}} \gg 1/B$ and applying Parseval's theorem, we derive the variance of $n_0[n]$ as (also see [30])

$$\begin{aligned}
\operatorname{var}(n_0[n]) &\approx \frac{N_0^2}{4} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \\
&\quad \times \left[4B^2 \operatorname{sinc}^2(2B(t-u))\right] dt\, du \\
&\approx \frac{N_0^2}{4} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \left[\int_{-B}^{B} 1\, df\right] dt \\
&= \frac{N_0^2 B T_{\text{sam}}}{2}.
\end{aligned} \tag{A.3}$$

In summary, $n_0[n]$ is approximately zero mean and white with variance $N_0^2 B T_{\text{sam}}/2$. These noise autocorrelation samples are uncorrelated with each other, due to the assumption $T_{\text{sam}} \gg 1/B$.

Furthermore, the aggregate noise term $n_1[n]$ is

$$\begin{aligned}
n_1[n] &= n_0[n] + s_i c_j \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau)n(t) \\
&\quad + h(t-D-\tau)n(t+D)]\, dt \\
&\quad + \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau)n(t+D) \\
&\quad + h(t+D-\tau)n(t)]\, dt.
\end{aligned} \tag{A.4}$$

Defining

$$\begin{aligned}
&\gamma'[n] \\
&\quad = s_i c_j \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau)n(t) + h(t-D-\tau)n(t+D)]\, dt,
\end{aligned} \tag{A.5}$$

$$\begin{aligned}
&\gamma''[n] \\
&\quad = \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} [h(t-\tau)n(t+D) + h(t+D-\tau)n(t)]\, dt,
\end{aligned} \tag{A.6}$$

we obtain

$$n_1[n] = \gamma'[n] + \gamma''[n] + n_0[n], \tag{A.7}$$

where $\gamma'[n]$ and $\gamma''[n]$ are random variables, resulting from the cross-correlation between the signal and the noise.

Now we will derive the statistical properties of these two random variables. Both $\gamma'[n]$ and $\gamma''[n]$ have zero mean. The variance of $\gamma'[n]$ is calculated as follows:

$$\begin{aligned}
&\operatorname{var}\left(\gamma'[n]\right) \\
&\quad = E\left[\left|\gamma'[n]\right|^2\right] \\
&\quad = \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \big[h(t-\tau)h(u-\tau)R_n(t-u) \\
&\qquad\qquad + h(t-D-\tau)h(u-D-\tau) \\
&\qquad\qquad \times R_n(t-u)\big]\, dt\, du.
\end{aligned} \tag{A.8}$$

Let us insert $R_n(\tau)$ into the first term (also see [30]) as follows:

$$\begin{aligned}
&\int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau)h(u-\tau)R_n(t-u)\, dt\, du \\
&\quad = \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau)h(u-\tau) \\
&\qquad \times N_0 B\, \operatorname{sinc}(2B(t-u))\, dt\, du \\
&\quad = \frac{N_0}{2} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau) \\
&\qquad \times h(u-\tau) \int_{-B}^{B} e^{j2\pi f(t-u)}\, df\, dt\, du \\
&\quad = \frac{N_0}{2} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau) \int_{-B}^{B} e^{j2\pi f(t-\tau)}\, df\, dt \\
&\qquad \times \int_{(n-1)T_{\text{sam}}-\tau}^{nT_{\text{sam}}-\tau} h(u-\tau) e^{-j2\pi f(u-\tau)}\, d(u-\tau) \\
&\quad = \frac{N_0}{2} \int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau) \left(\int_{-B}^{B} H(f) e^{j2\pi f(t-\tau)}\, df\right) dt,
\end{aligned} \tag{A.9}$$

where $H(f)$ is the Fourier transform of $h(u-\tau)$, $u \in [(n-1)T_{\text{sam}}, nT_{\text{sam}}]$, which is a segment of the aggregate channel. Since the bandwidth B of $n(t)$ is assumed much larger than the bandwidth of $h(u-\tau)$, $u \in [(n-1)T_{\text{sam}}, nT_{\text{sam}}]$, we obtain $\int_{-B}^{B} H(f)e^{j2\pi f(t-\tau)}df \approx h(t-\tau)$, $t \in [(n-1)T_{\text{sam}}, nT_{\text{sam}}]$. As a result, we obtain similar results as in [24, 25, 30] as follows:

$$\begin{aligned}
&\int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}}\int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau)h(u-\tau)R_n(t-u)dt\,du \\
&\quad\approx \frac{N_0}{2}\int_{(n-1)T_{\text{sam}}}^{nT_{\text{sam}}} h(t-\tau)h(t-\tau)dt \\
&\quad= \frac{N_0}{2}R(0,n-\delta).
\end{aligned} \tag{A.10}$$

In a similar way, the other term of $\text{var}(\gamma'[n])$ can be deduced. The same method is applied to $\text{var}(\gamma''[n])$ and $E[\gamma'[n]\gamma''[n]]$. All the derivations are based on the assumption that $R_n(D) \approx 0$ and $T_{\text{sam}} \gg 1/B$. The results are summarized as follows:

$$\begin{aligned}
&\text{var}(\gamma'[n]) \\
&\quad\approx \begin{cases} \dfrac{N_0}{2}\left(R(0,n-\delta)+R\left(0,n-\delta-\dfrac{D}{T_{\text{sam}}}\right)\right), \\ \qquad\qquad n=\delta+1,\ \delta+2,\ldots,\delta+P_h, \\ 0, \quad \text{elsewhere,} \end{cases}
\end{aligned} \tag{A.11}$$

$$\begin{aligned}
&\text{var}(\gamma''[n]) \\
&\quad= E[|\gamma''[n]|^2] \\
&\quad\approx \begin{cases} \dfrac{N_0}{2}\left(R(0,n-\delta)+R\left(0,n-\delta+\dfrac{D}{T_{\text{sam}}}\right)\right), \\ \qquad\qquad n=\delta+1,\ \delta+2,\ldots,\delta+P_h, \\ 0, \quad \text{elsewhere,} \end{cases}
\end{aligned} \tag{A.12}$$

$$\begin{aligned}
&E[\gamma'[n]\gamma''[n]] \\
&\quad\approx \begin{cases} \dfrac{N_0}{2}s_ic_j\left(R(D,n-\delta)+R\left(D,n-\delta+\dfrac{D}{T_{\text{sam}}}\right)\right), \\ \qquad\qquad n=\delta+1,\ \delta+2,\ldots,\delta+P_h, \\ 0, \quad \text{elsewhere,} \end{cases}
\end{aligned} \tag{A.13}$$

$$E[\gamma'[n]n_0[n]] = E[\gamma''[n]n_0[n]] = 0. \tag{A.14}$$

In summary, the stochastic properties of $n_1[n]$ are

$$\begin{aligned}
&E[n_1[n]] \approx 0, \\
&\text{var}(n_1[n]) \\
&\quad\approx \begin{cases} \dfrac{N_0}{2}\left\{2R(0,n-\delta)+R\left(0,n-\delta-\dfrac{D}{T_{\text{sam}}}\right)\right. \\ \qquad +R\left(0,n-\delta+\dfrac{D}{T_{\text{sam}}}\right) \\ \qquad \left.+s_ic_j\left(2R(D,n-\delta)+2R\left(D,n-\delta+\dfrac{D}{T_{\text{sam}}}\right)\right)\right\} \\ \qquad +\sigma_0^2, \quad n=\delta+1,\ \delta+2,\ldots,\delta+P_h, \\ 0, \quad \text{elsewhere,} \end{cases}
\end{aligned} \tag{A.15}$$

where $\sigma_0^2 = N_0^2BT_{\text{sam}}/2$. These aggregate noise samples are uncorrelated with each other, recalling that $T_{\text{sam}} \gg 1/B$. This assumption has usually been satisfied by UWB signals (e.g., in our case $T_{\text{sam}} = 10$ ns, $B \approx 2/T_p = 10$ GHz, then $2BT_{\text{sam}} = 200$). Also $n_0[n]$ and $n_1[n]$ can be assumed as Gaussian random variables by invoking the sampling theorem and the central limit theorem [28].

B. Detector Derivation

In summary, the statistics of $\mathbf{x}$ in (31) are

$$\mathcal{H}_0\text{: } \mathbf{x} \overset{a}{\sim} \mathcal{N}(0, \sigma_0^2\mathbf{I}), \tag{B.1}$$

$$\mathcal{H}_1\text{: } \mathbf{x} \overset{a}{\sim} \mathcal{N}(\mathbf{1}_{M_1N_f} \otimes \mathbf{z}_f, \text{diag}(\boldsymbol{\lambda})). \tag{B.2}$$

The *Neyman-Pearson* detector decides $\mathcal{H}_1$ if

$$L(\mathbf{x}) = \frac{p(\mathbf{x};\mathcal{H}_1)}{p(\mathbf{x};\mathcal{H}_0)} > \gamma, \tag{B.3}$$

where γ is found by making the probability of false alarm P_{FA} to satisfy

$$P_{\text{FA}} = Pr\{L(\mathbf{x}) > \gamma; \mathcal{H}_0\} = \alpha. \tag{B.4}$$

$L(\mathbf{x})$ can be expressed as

$$L(\mathbf{x}) = \frac{\displaystyle\prod_{i=1}^{P} \frac{1}{(2\pi(2N_0z_f[i]+\sigma_0^2))^{(M_1N_f/2)}} \exp\left[-\frac{1}{2(2N_0z_f[i]+\sigma_0^2)}\sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1}(x[nP+i]-z_f[i])^2\right]}{\dfrac{1}{(2\pi\sigma_0^2)^{M_1N_fP/2}} \exp\left[-\dfrac{1}{2\sigma_0^2}\displaystyle\sum_{n=(k-1)N_fP+1}^{(k+M_1-1)N_fP} x^2[n]\right]}. \tag{B.5}$$

Defining $\sigma_1^2[i] = 2N_0 z_f[i] + \sigma_0^2$, inserting it into $\ln L(\mathbf{x})$, and eliminating the constants leads to

$$\begin{aligned}
&\ln L(\mathbf{x}) \\
&= \sum_{i=1}^{P} \left\{ \frac{1}{2\sigma_0^2} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x^2[nP+i] \right. \\
&\qquad \left. - \frac{1}{2\sigma_1^2[i]} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} \left(x[nP+i] - z_f[i]\right)^2 \right\} \\
&= \sum_{i=1}^{P} \left\{ \frac{2z_f[i]}{2\sigma_1^2[i]} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x[nP+i] \right. \\
&\qquad \left. + \left(\frac{1}{2\sigma_0^2} - \frac{1}{2\sigma_1^2[i]} \right) \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x^2[nP+i] \right\} \\
&= \sum_{i=1}^{P} \left\{ \frac{z_f[i]}{\sigma_1^2[i]} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x[nP+i] \right. \\
&\qquad \left. + \frac{N_0 z_f[i]}{\sigma_0^2 \sigma_1^2[i]} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x^2[nP+i] \right\} \\
&= \sum_{i=1}^{P} \frac{z_f[i]}{\sigma_1^2[i]} \left\{ \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x[nP+i] \right. \\
&\qquad \left. + \frac{N_0}{\sigma_0^2} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x^2[nP+i] \right\}.
\end{aligned} \tag{B.6}$$

Then, the test statistic is

$$\begin{aligned}
T(\mathbf{x}) = \sum_{i=1}^{P} \frac{z_f[i]}{\sigma_1^2[i]} &\left\{ \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x[nP+i] \right. \\
&\left. + \frac{N_0}{\sigma_0^2} \sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x^2[nP+i] \right\}.
\end{aligned} \tag{B.7}$$

C. Statistic of the Detectors

C.1. Detector $T_1(\mathbf{x})$. Since $\mathbf{x}$ is assumed to be a Gaussian vector, $T_1(\mathbf{x})$ also follows a Gaussian distribution:

$$\begin{aligned}
&\mathcal{H}_0\colon\ T_1(\mathbf{x}) \overset{a}{\sim} \mathcal{N}\left(0, M_1 N_f {\sigma_0}^2 \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^4[i]}\right), \\
&\mathcal{H}_1\colon\ T_1(\mathbf{x}) \overset{a}{\sim} \mathcal{N}\left(M_1 N_f \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^2[i]},\ M_1 N_f \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^2[i]}\right).
\end{aligned} \tag{C.1}$$

Actually, if the condition $z_f[i]/N_0 \ll BT_{\text{sam}}/4$ is satisfied, which means the signal-to-noise ratio (SNR) is low, the term $2N_0 z_f[i]$ can be ignored in the variance of $\mathbf{x}$ under $\mathcal{H}_1$, and then $T_1(\mathbf{x})$ can be derived directly.

C.2. Detector $T_2(\mathbf{x})$. Since the different entries of $\mathbf{x}$ have different weighting factors in $T_2(\mathbf{x})$, we collect the data samples bearing the same weighting factor into the same group. Therefore, there are P groups of data samples, and they are assumed to be uncorrelated. Each group $\sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} x^2[nP+i]$ follows a Chi-squared distribution. However, $T_2(\mathbf{x})$ is still assumed to be a Gaussian variable, as it is the sum of the weighted groups. Then, we can obtain

$$\begin{aligned}
&\mathcal{H}_0: \\
&\sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} \frac{x^2[nP+i]}{\sigma_0^2} \overset{a}{\sim} \chi^2_{M_1 N_f}, \\
&T_2(\mathbf{x}) \overset{a}{\sim} \mathcal{N}\left(M_1 N_f {\sigma_0}^2 \sum_{i=1}^{P} \frac{z_f[i]}{\sigma_1^2[i]}, 2M_1 N_f {\sigma_0}^4 \sum_{i=1}^{P} \frac{z_f^2[i]}{\sigma_1^4[i]}\right), \\
&\mathcal{H}_1: \\
&\sum_{n=(k-1)N_f}^{(k+M_1-1)N_f-1} \frac{x^2[nP+i]}{\sigma_1^2[i]} \overset{a}{\sim} \chi^2_{M_1 N_f}\left(\frac{M_1 N_f E_f^2[i]}{\sigma_1^2[i]}\right), \\
&T_2(\mathbf{x}) \overset{a}{\sim} \mathcal{N}\left(M_1 N_f \sum_{i=1}^{P} z_f[i]\left(1 + \frac{z_f^2[i]}{\sigma_1^2[i]}\right),\right. \\
&\qquad \left. 2M_1 N_f \sum_{i=1}^{P} z_f^2[i]\left(1 + \frac{2z_f^2[i]}{\sigma_1^2[i]}\right)\right),
\end{aligned} \tag{C.2}$$

where χ_ν^2 is the central Chi-squared pdf with ν degrees of freedom, which has mean ν and variance 2ν. Meanwhile, $\chi_\nu^2(\lambda)$ is the noncentral Chi-squared pdf with ν degrees of freedom and noncentrality parameter λ. Hence, it has mean $\nu + \lambda$ and variance $2\nu + 4\lambda$.

Acknowledgments

This work was supported in part by STW under the Green and Smart Process Technologies Program (Project 7976) and by NWO-STW under the VICI programme (DTC. 5893). Parts of this paper were presented in [17].

References

[1] L. Yang and G. B. Giannakis, "Ultra-wideband communications: an idea whose time has come," *IEEE Signal Processing Magazine*, vol. 21, no. 6, pp. 26–54, 2004.

[2] Z. Tian and G. B. Giannakis, "BER sensitivity to mistiming in ultra-wideband impulse radios—part II: fading channels," *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1897–1907, 2005.

[3] R. Blazquez, P. Newaskar, and A. Chandrakasan, "Coarse acquisition for ultra wideband digital receivers," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '03)*, vol. 4, pp. 137–140, Hong Kong, April 2003.

[4] V. Lottici, A. D'Andrea, and U. Mengali, "Channel estimation for ultra-wideband communications," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 9, pp. 1638–1645, 2002.

[5] S. R. Aedudodla, S. Vijayakumaran, and T. F. Wong, "Timing acquisition in ultra-wideband communication systems," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 5, pp. 1570–1583, 2005.

[6] Z. Tian and G. B. Giannakis, "A GLRT approach to data-aided timing acquisition in UWB radios—part I: algorithms," *IEEE Transactions on Wireless Communications*, vol. 4, no. 6, pp. 2956–2967, 2005.

[7] J. Kusuma, I. Maravić, and M. Vetterli, "Sampling with finite rate of innovation: channel and timing estimation for UWB and GPS," in *Proceedings of IEEE International Conference on Communications (ICC '03)*, vol. 5, pp. 3540–3544, Anchorage, Alaska, USA, May 2003.

[8] L. Yang and G. B. Giannakis, "Timing ultra-wideband signals with dirty templates," *IEEE Transactions on Communications*, vol. 53, no. 11, pp. 1952–1963, 2005.

[9] I. Guvenc, Z. Sahinoglu, and P. V. Orlik, "TOA estimation for IR-UWB systems with different transceiver types," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 4, pp. 1876–1886, 2006.

[10] R. Hoctor and H. Tomlinson, "Delay-hopped transmitted-reference RF communications," in *Proceedings of IEEE Conference on Ultra Wideband Systems and Technologies (UWBST '02)*, pp. 265–269, Baltimore, Md, USA, May 2002.

[11] M. Ho, V. S. Somayazulu, J. Foerster, and S. Roy, "A differential detector for an ultra-wideband communications system," in *Proceedings of the 55th IEEE Vehicular Technology Conference (VTC '02)*, vol. 4, pp. 1896–1900, Birmingham, Ala, USA, May 2002.

[12] Z. Tian and B. M. Sadler, "Weighted energy detection of ultra-wideband signals," in *Proceedings of the 6th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC '05)*, pp. 1068–1072, New York, NY, USA, June 2005.

[13] Y. Vanderperren, G. Leus, and W. Dehaene, "A reconfigurable pulsed UWB receiver sampling below nyquist rate," in *Proceeedings of the IEEE International Conference on Ultra-Wideband (ICUWB '08)*, vol. 2, pp. 145–148, Hannover, Germany, September 2008.

[14] J. Kim, S. Yang, and Y. Shin, "A two-step search scheme for rapid and reliable UWB signal acquisition in multipath channels," in *Proceedings of IEEE International Conference on Ultra-Wideband (ICU '05)*, pp. 355–360, Zurich, Switzerland, September 2005.

[15] S. Gezici, Z. Sahinoglu, A. F. Molisch, H. Kobayashi, and H. V. Poor, "Two-step time of arrival estimation for pulse-based ultra-wideband systems," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 529134, 11 pages, 2008.

[16] M. R. Casu and G. Durisi, "Implementation aspects of a transmitted-reference UWB receiver," *Wireless Communications and Mobile Computing*, vol. 5, no. 5, pp. 537–549, 2005.

[17] Y. Wang, G. Leus, and A.-J. van der Veen, "On digital receiver design for transmitted reference UWB," in *Proceedings of IEEE International Conference on Ultra-Wideband (ICUWB '08)*, vol. 3, pp. 35–38, Hannover, Germany, September 2008.

[18] S. Bagga, L. Zhang, W. A. Serdijn, J. R. Long, and E. B. Busking, "A quantized analog delay for an IR-UWB quadrature downconversion autocorrelation receiver," in *Proceedings of IEEE International Conference on Ultra-Wideband (ICU '05)*, pp. 328–332, Zurich, Switzerland, September 2005.

[19] R. C. Qiu, H. Liu, and X. Shen, "Ultra-wideband for multiple access communications," *IEEE Communications Magazine*, vol. 43, no. 2, pp. 80–87, 2005.

[20] R. Djapic, G. Leus, and A.-J. van der Veen, "Blind synchronization in asynchronous UWB networks based on the transmit-reference scheme," in *Proceedings of the 38th Asilomar Conference on Signals, Systems and Computers (ACSSC '04)*, vol. 2, pp. 1506–1510, Pacific Grove, Calif, USA, November 2004.

[21] Q. H. Dang and A.-J. van der Veen, "A decorrelating multiuser receiver for transmit-reference UWB systems," *IEEE Journal on Selected Topics in Signal Processing*, vol. 1, no. 3, pp. 431–442, 2007.

[22] J. D. Choi and W. E. Stark, "Performance of ultra-wideband communications with suboptimal receivers in multipath channels," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 9, pp. 1754–1766, 2002.

[23] K. Witrisal, G. Leus, M. Pausini, and C. Krall, "Equivalent system model and equalization of differential impulse radio UWB systems," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 9, pp. 1851–1862, 2005.

[24] T. Q. S. Quek and M. Z. Win, "Analysis of UWB transmitted-reference communication systems in dense multipath channels," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 9, pp. 1863–1874, 2005.

[25] T. Q. S. Quek, M. Z. Win, and D. Dardari, "Unified analysis of UWB transmitted-reference schemes in the presence of narrowband interference," *IEEE Transactions on Wireless Communications*, vol. 6, no. 6, pp. 2126–2139, 2007.

[26] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume 1: Estimation Theory*, Prentice-Hall, Upper Sadle River, NJ, USA, 1993.

[27] J. R. Foerster, "Channel modeling sub-committee report final," Tech. Rep. IEEE P802.15-02/368r5-SG3a, IEEE P802.15 Working Group for WPANs, November 2002.

[28] H. Stark and J. W. Woods, *Probability, Random Processes, and Estimation Theory for Engineers*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1994.

[29] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communication Techniques*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.

[30] L. Yang and G. B. Giannakis, "Optimal pilot waveform assisted modulation for ultrawideband communications," *IEEE Transactions on Wireless Communications*, vol. 3, no. 4, pp. 1236–1249, 2004.