Synchronization for Impulse-Radio UWB With Energy-Detection and Multi-User Interference: Algorithms and Application to IEEE 802.15.4a

Manuel Flury, Ruben Merz, *Member, IEEE*, and Jean-Yves Le Boudec, *Fellow, IEEE*

Abstract—Energy-detection (ED) receivers can take advantage of the ranging and multipath resistance capabilities of impulse-radio ultra-wideband (IR-UWB) physical layers at a much lower complexity than coherent receivers. However, ED receivers are extremely vulnerable to multi-user interference (MUI). Therefore, the design of IR-UWB ED architectures must take MUI into account. In this paper, we present the design and evaluation of two complementary algorithms for reliable and robust synchronization of IR-UWB ED receivers in the presence of MUI: 1) power-independent detection and preamble code interference cancellation (PICNIC) and 2) detection of start-frame-delimiter through sequential ratio tests (DESSERT). PICNIC addresses packet detection and timing acquisition while DESSERT focuses on start-frame-delimiter (SFD) detection. Both algorithms are evaluated with the IEEE 802.15.4a IR-UWB physical layer, standardized for low data-rate networks. The performance evaluation with extensive simulations show that our algorithms outperform nonrobust synchronization algorithms by up to two orders of magnitude in the presence of MUI.

Index Terms—IEEE 802.15 standards, interference cancellation, multiple-access interference, ultra-wideband communication.

I. INTRODUCTION

THE IEEE 802.15.4 standard targets low data-rate wireless networks with extensive battery life and low complexity[1] [2]. IEEE 802.15.4a [3], [4] is an amendment that specifies an impulse-radio ultra-wideband (IR-UWB) physical layer

(PHY) [5]. It allows for several independent networks to operate simultaneously in the same frequency band without coordination. In addition, the corresponding mandatory IEEE 802.15.4a medium access control (MAC) protocol uses random access with no carrier sensing [3, Sec. 7]. Therefore, IEEE 802.15.4a receivers must cope with *uncontrolled* multi-user interference (MUI) including near–far scenarios.

IEEE 802.15.4a can support both coherent and noncoherent receiver architectures. Coherent receivers are resilient to MUI. The wide bandwidth provides ample multipath that can be used as a spatial signature to separate transmitters when coherently combined. However, they require a high sampling rate [6] and stringent timing requirements [7]. Because of their low complexity and low requirements in terms of sampling rate and channel estimation and their robustness to timing impairments (see [8] and [9]), the alternative provided by noncoherent receivers using energy-detection (ED) is often preferred for IEEE 802.15.4 networks.

However, MUI is much less mitigated by a noncoherent receiver than by a coherent one [10]. And not accounting for MUI with an ED architecture can greatly degrade system performance, even at low data-rates [11], [12]. One option to prevent MUI is to coordinate medium access, for instance, with carrier sensing. However, the low power emitted, the intermittent characteristics of IR-UWB signals and the possible absence of a carrier make it hardly feasible to reliably perform carrier sensing or clear channel assessment (CCA) with a reasonable complexity.

In this paper, we present a successful alternative to mitigate MUI. Our approach combats MUI by taking advantage of its non-Gaussian structure in low data-rate IR-UWB systems [13]–[15]. The present study focuses on synchronization and develops algorithms for ED receivers that remain operational, even in strong near–far scenarios (Section V). We combine the low-complexity characteristics of ED receivers with robustness to MUI. We address exclusively MUI generated by similar IR-UWB transceivers. Coexistence with narrow-band systems is out of scope. Detailed information on this topic can be found in [9] and the references therein.

1) Summary of Our Contributions: To the best of our knowledge, our work is the first to design and evaluate a synchronization solution that 1) mitigates MUI for IR-UWB energy-detection receivers and 2) covers *all* steps of synchronization, namely packet detection, timing acquisition and start frame delimiter (SFD) detection (see Section II-A).

[1]The Zigbee specification is currently based on the IEEE 802.15.4 standard.

Our solution comprises two complementary algorithms: 1) power-independent detection and preamble code interference cancellation (PICNIC) and 2) detection of SFD through sequential ratio tests (DESSERT).

- PICNIC addresses packet detection and timing acquisition. For packet detection, PICNIC uses a correlation-based packet detection algorithm as in [11] and [16]. However, the novelty of PICNIC and its robustness to interference rely on 1) a preprocessing prior to the correlation of the received signal using thresholding and 2) a postprocessing of the correlation output with an interference detection and cancellation mechanism. For timing acquisition, a prior jump-back and search-forward algorithm [17] is enhanced with a simple but efficient heuristic.

- DESSERT addresses SFD detection. DESSERT uses a likelihood-based approach as in the recent work [18], [19]. However, DESSERT innovates in three different aspects: 1) detection is online, which greatly simplifies implementation and reduces memory requirements, 2) detection uses soft-decision, and 3) a postprocessing on the received signal is used to provide robustness against MUI.

The major breakthrough of the proposed solution is that PICNIC and DESSERT allow a given transmitter to successfully synchronize in the presence of several other interfering transmitters, even in severe near–far scenarios (Section V). Our algorithms are compliant with the IEEE 802.15.4a standard. They can also be used with a classic IR-UWB PHY [5].

2) Organization of the Paper: In the remainder of this paper, we introduce our system model, the IEEE 802.15.4a PHY, and assumptions made throughout the paper in Section II. We then describe PICNIC and DESSERT in Sections III and IV, and evaluate their performance in Section V. Related work is discussed in Section VI and we conclude in Section VII.

II. SYSTEM MODEL AND ASSUMPTIONS

We consider a packet-based system of IR-UWB [5] radios that follow the IEEE 802.15.4a specification [3].

A. Synchronization in Packet-Based Networks

In packet-based systems with no global clock, such as IEEE 802.11 or IEEE 802.15.4a, synchronization is achieved on a per-packet basis. In our terminology, synchronization corresponds to packet detection, timing acquisition and SFD detection. Packet detection, followed by timing acquisition are the first steps towards the correct reception of a packet. A destination detects any packet intended for itself and finds the time reference of the transmitter. Only then, can the destination begin to look for a SFD. The SFD is a special data sequence that marks the end of the preamble and the beginning of the payload. Once it is detected, the destination can recover the payload.

B. IEEE 802.15.4a IR-UWB Physical Layer: Packet and Signal Formats

1) Packet Format and Terminology: IEEE 802.15.4a packets consist of a preamble, known by receivers, followed by a payload. The preamble comprises 1) a synchronization part (the
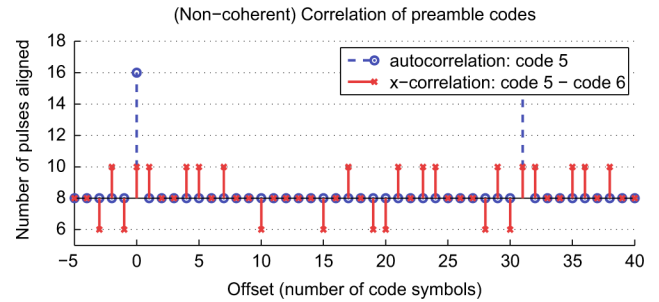

Fig. 1. Auto- and cross-correlation of the two IEEE 802.15.4a preamble codes 5 and 6 (length $C = 31$) with noncoherent reception. A cross-correlation period shows ten peaks that may cause false alarms. Note the minimum abscissa (six) is positive because of noncoherent reception.

SYNC part) for packet detection, timing acquisition and channel estimation, and 2) an SFD to mark the beginning of the payload. The basic time unit is a *chip* of duration T_c seconds. A *preamble symbol* is obtained by modulating C consecutive pulses, sent periodically every Lth chip, according to a ternary *preamble code* of length C. The SYNC part is assembled by repeating N_{sync} preamble symbols. The SFD is obtained by spreading a ternary *SFD code* of length N_{sfd} with a preamble symbol. The full preamble corresponds to $N_{\text{pre}} = N_{\text{sync}} + N_{\text{sfd}}$ preamble symbols.

2) IEEE 802.15.4a Preamble Codes: There are two possible preamble codes allocated per frequency band. Although they have perfect autocorrelation, the same is not true of their cross-correlation if noncoherent reception is used [11]. Fig. 1 shows the auto- and cross-correlation for the preamble codes 5 and 6 (band 3, length $C = 31$). We observe that code 5 has a periodic autocorrelation with only one peak per period but its cross-correlation with code 6 shows $C_{\text{cross-peak}} = 10$ peaks per period. Note that the minimum abscissa (equal to 6) is positive because of noncoherent operation. The codes employed in IEEE 802.15.4a are based on m-sequences and have the minimal three-valued cross-correlation that can be achieved by such sequences [20], [21]. Hence, no such sequences with better cross-correlation properties can be found. Consequently, these cross-correlation peaks are not specific to IEEE 802.15.4a but affect any system using multiple such preamble codes with perfect periodic autocorrelation. These cross-correlation peaks can generate false alarms (FA) or missed detections (MD) and the performance loss entailed is significant (see Section V).

3) Received Signal Modeling: The received preamble signal is given by

$$r(t) = x(t - \nu_0) + v(t)$$
$$= \sum_{i=0}^{N_{\text{pre}}-1} s_i \sum_{j=0}^{C-1} c_j \cdot h(t - (j + iC)LT_c - \nu_0) + v(t) \quad (1)$$

where $x(t)$ is the contribution of the user of interest (UOI), ν_0 is the propagation delay and $v(t)$ accounts for thermal noise and MUI. We model thermal noise as a zero-mean additive white Gaussian noise (AWGN) process with power spectral density (PSD) $\frac{N_0}{2}$. Because it is generated by the receiver circuitry, we assume that an external calibration procedure provides an estimate of N_0. A sensitivity analysis of the thermal noise estimation is outside the scope of this

article. MUI is created by interfering IEEE 802.15.4a devices using the same physical layer as the UOI. Furthermore, $c_j \in \{-1, 0, +1\}, j \in \{0, 1, \ldots, C-1\}$ is the jth *code symbol* of the ternary preamble code. Every preamble symbol is further modulated by s_i where

$$s_i = \begin{cases} 1 & \text{if } i \in \{0, 1, \ldots, N_{\text{sync}} - 1\} \\ s_{i-N_{\text{sync}}}^{(\text{sfd})} & \text{if } i \in \{N_{\text{sync}}, \ldots, N_{\text{pre}} - 1\} \end{cases} \quad (2)$$

and $s_i^{(\text{sfd})} \in \{-1, 0, +1\}, i \in \{0, 1, \ldots, N_{\text{sfd}} - 1\}$ is the ternary SFD code. Finally, $h(t)$ denotes the unknown channel response, which is assumed invariant for the duration of one packet. It includes the transmitted waveform, the response of the multipath channel and any transmit or receive filters.

In contrast to the preamble signal, each symbol of an IEEE 802.15.4a payload is composed of a short *burst* of L_b pulses with pseudo-random polarity and time-hopping [3]. Hence, the payload signal from an interfering transmitter contains roughly L_b times more energy than an interfering preamble pulse [11].

C. Noncoherent Energy-Detection Receiver Model

Reception is noncoherent with energy-detection. Our receiver model comprises an antenna, a bandpass filter of bandwidth B, a squaring device, and an integrator sampled at rate $\frac{1}{T}$. We assume that the *integration time* is $T = \frac{L}{M} T_c$ where M is an integer divisor of L. With the minimum inter-pulse spacing $L T_c$, we get M discrete samples per pulse and the discrete time signal after sampling is

$$y_n = \int_{nT}^{(n+1)T} [r(t)]^2 \, dt = \int_{nT}^{(n+1)T} \left[\sum_{i=0}^{N_{\text{pre}}-1} s_i \sum_{j=0}^{C-1} c_j \right.$$
$$\left. \cdot h(t - (j + iC) L T_c - \nu_0) + v(t) \right]^2 dt. \quad (3)$$

D. Statistics of the Receiver Output Without MUI

If $x(t) = 0$ and there is no MUI, then $y_n = \int_{nT}^{(n+1)T} [v(t)]^2 \, dt$ and $v(t)$ is AWGN. In this case, the distribution of y_n can be approximated with a chi-square distribution with $2BT$ degrees of freedom [22]. We denote the cumulative distribution function (CDF) of a chi-square random variable with κ degrees of freedom by $F_{\chi^2}(z|\kappa)$.

If $x(t) \neq 0$ and there is no MUI, then the distribution of every sample y_n can be approximated by a scaled noncentral chi-square distribution with $2BT$ degrees of freedom and noncentrality parameter $\frac{p_n}{\frac{N_0}{2}}$, where

$$p_n = \int_{nT}^{(n+1)T} [x(t)]^2 \, dt = s_{\lfloor \frac{n}{CM} \rfloor}^2 \cdot c_{\lfloor \frac{n}{M} \rfloor \bmod C}^2 \cdot q_{n \bmod M}. \quad (4)$$

In the second equality we used (1), assumed no inter-pulse interference ($h(t) = 0, \forall t < 0$ and $\forall t > L T_c$) and perfect synchronization ($\nu_0 = 0$). Furthermore

$$q_m = \int_{mT}^{(m+1)T} [h(t)]^2 dt, \quad m \in \{0, 1 \ldots, M-1\} \quad (5)$$

where q_m represents the power-delay profile (PDP) of the channel. It can be estimated from the preamble [18], [19], [23]. Algorithms for robust estimation of q_m in the presence of MUI are available in [12], [24]. We denote the CDF of a noncentral chi-square random variable with κ degrees of freedom and noncentrality parameter ζ by $F_{\text{NC}\chi^2}(z|\kappa, \zeta)$.

III. PACKET DETECTION AND TIMING ACQUISITION ALGORITHMS WITH INTEGRATED MUI MITIGATION MECHANISMS

This section presents the PICNIC algorithm, which comprises three parts: packet detection yielding initial and coarse timing information generally corresponding to the location of strongest multipath component (Section III-A); timing acquisition to locate the first received component because the strongest is not always the first in time (Section III-B); and an interference cancellation (IC) module (Section III-C). The IC module addresses the performance anomaly created by the imperfect cross-correlation of preamble codes.

A. Correlation-Based Power-Independent Packet Detection With Thresholding Preprocessing

1) Correlation Structure: Similar to earlier work [11], [16], the core of the packet detection procedure relies on a correlation. However, the correlation operation

$$z_n = \sum_{l=0}^{N_T-1} t_l \cdot \mathbf{1}_{[y_{n-(N_T-1)+l} > \eta_{\text{pid}}]} \quad (6)$$

is performed between a template t_l and a preprocessed receiver output y_n, where $\mathbf{1}_{[\cdot]}$ denotes the indicator function, η_{pid} is a threshold and z_n is the correlation output. Samples above η_{pid} are set to 1, samples below η_{pid} are set to 0. This thresholding operation addresses two issues of correlation-based algorithms when the received power from an interferer is higher than the received power of the UOI:

1) in case the interferer is using a different preamble sequence, high cross-correlation values can cause packet detection with the UOI to fail (in case of concurrent transmission), or create false alarms;

2) if the interferer uses the same preamble sequence, the packet detection algorithm can be constantly biased towards the packets with the stronger power.

The template t_l has length $N_T = N_G \cdot C \cdot M$ where N_G is a processing gain parameter. The template is derived from the known preamble code c_j of the UOI:

$$t_l = \sum_{k=0}^{N_G-1} \sum_{j=0}^{C-1} c_j^2 \cdot \delta_{l-(j+kC)M} \quad (7)$$

where δ_n denotes the Kronecker delta and c_j is the preamble code of length C.

2) Detection and Verification: The correlation output in (6) yields a discrete sequence z_n. If a UOI signal is present, i.e., $x(t) \neq 0$, z_n is $(M \cdot C)$-periodic. Therefore, the algorithm
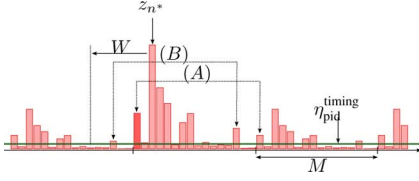
Fig. 2. Illustration of the jump-back and search-forward algorithm. Starting from the coarse-timing acquisition z_{n^*} obtained with packet detection, a window of length W is searched. While both (A) and (B) are above the threshold $\eta_{\mathrm{pid}}^{\mathrm{timing}}$, only (A) fulfills the second condition given by (10) and is selected as first path.

processes z_n by blocks of MC consecutive samples. We define the ith block $(i = 0, 1, \ldots)$ as

$$\mathbf{z_i} = \left[z_{iMC}, z_{iMC+1}, \ldots, z_{(i+1)MC-1} \right]. \tag{8}$$

Let N_V denote the length of the verification phase. A packet is detected when the maxima in N_V consecutive blocks are well aligned, i.e., when they exhibit the expected $(M \cdot C)$-periodicity. Specifically, for every block $\mathbf{z_{i+k}}$ $(k = 1, \ldots, N_V)$ the index of its maximum $z_{(i+k)MC+j_{i+k}^{\max}}$ must not differ by more than M (the minimum inter-pulse distance) from the index of the maximum $z_{(i+k)MC+j_{i+k-1}^{\max}}$ in block $\mathbf{z_{i+k-1}}$. Note that $j_{i+k}^{\max} \in \{0, \ldots, CM-1\}$. This condition ensures that both maxima stem from the same preamble pulse. If this condition is not verified for N_V consecutive blocks, packet detection keeps running.

3) Calculation of the Threshold η_{pid}: The goal of the preprocessing using η_{pid} is to ensure that, with high probability, y_n is set to 1 if a UOI signal or MUI is present and set to 0 if there is only AWGN. We derive η_{pid} from the statistics of the receiver output (Section II-D) according to

$$\eta_{\mathrm{pid}} = \frac{N_0}{2} F_{\chi^2}^{-1}(1 - P_{\mathrm{AWGN}}^{\mathrm{FA,PID}} | 2BT) \tag{9}$$

where the parameter $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$ denotes the desired (false alarm) probability that a received signal corrupted only by AWGN exceeds the threshold.

B. Timing Acquisition

Packet detection yields a coarse timing acquisition that generally corresponds to the strongest multipath component rather than the first in time. Timing acquisition tries to locate the first sample that i) belongs to the same pulse and ii) is above the noise-floor. Our timing acquisition algorithm uses a jump-back and search-forward (JBSF) procedure as in [17] (see Fig. 2), but with two major differences: 1) the JBSF is applied on the correlation output z_n and not on the received signal y_n, and 2) a different detection criterion is applied (see (10)). Using the correlation output enables to benefit from processing gain.

1) JBSF Procedure: We limit the search-back window to $W = \frac{M}{2}$ samples because we assume that the first and strongest paths are separated by less than half the minimum inter-pulse distance. If packet detection yields the sample z_{n^*} with index $n^* = (i + N_V)MC + j_{i+N_V}^{\max}$, the set of candidates for timing acquisition is $\{z_{n^*-W+1}, \ldots, z_{n^*-2}, z_{n^*-1}, z_{n^*}\}$. A candidate is

considered if it is larger than a given threshold $\eta_{\mathrm{pid}}^{\mathrm{timing}}$ (see calculation hereafter). However, this detection criterion [17] is *not* sufficient. Indeed, with noncoherent reception, the minimum of the autocorrelation of each preamble code is nonzero and equals half its maximum value (see Fig. 1 where the autocorrelation value is 16 and its minimum is 8). Even if the template is misaligned, it is still aligned with half of the pulses of the preamble code and the correlation is likely to exceed the threshold because of noise and interference. Hence, an additional condition to ensure that we do not select samples from the preceding pulse is $z_{n^*-k} \geq z_{n^*-k+M}$. The intuition is that a correlation peak from a fully aligned template will always be higher than a secondary peak that occurs with a misaligned template (see Fig. 2). The complete detection criterion is

$$z_{n^*-k} \geq \eta_{\mathrm{pid}}^{\mathrm{timing}} \quad \text{and} \quad z_{n^*-k} \geq z_{n^*-k+M}. \tag{10}$$

2) Calculation of the Threshold $\eta_{\mathrm{pid}}^{\mathrm{timing}}$: The goal of $\eta_{\mathrm{pid}}^{\mathrm{timing}}$ is to decide whether a sample is above the noise floor. Because of η_{pid} in (6), the output of the correlation is distributed according to a binomial distribution with parameters $C_{\mathrm{NZ}} \cdot N_G$ and $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$, where C_{NZ} is the number of nonzero code symbols in the preamble code. The threshold for timing acquisition is then obtained with

$$\eta_{\mathrm{pid}}^{\mathrm{timing}} = F_{\mathrm{BIN}}^{-1} \left(1 - P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}} | C_{\mathrm{NZ}} N_G, P_{\mathrm{AWGN}}^{\mathrm{FA,PID}} \right) \tag{11}$$

where $F_{\mathrm{BIN}}^{-1}(z|N, p)$ is the inverse of the cumulative distribution function (CDF) of a binomial distribution with parameters N and p. The parameter $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$ denotes the desired probability that a signal with only AWGN exceeds $\eta_{\mathrm{pid}}^{\mathrm{timing}}$.

C. Preamble Code Interference Cancellation Algorithm

As explained in Section II-B-2), all pairs of IEEE 802.15.4a preamble codes used per frequency band do not have a perfect cross-correlation with ED receivers. These cross-correlation peaks can generate FA or MD inducing a significant performance loss (see Section V). Now, for a specific combination of preamble codes, the location of the cross-correlation peaks is known *a priori* and corresponds to a well-defined pattern. Because IEEE 802.15.4a allows only two codes per frequency band, there is only a single cross-correlation per band.

The IC algorithm exploits the *a priori* known location of the cross-correlation peaks to remove their effects. The IC algorithm proceeds in two steps: 1) interference detection and 2) if necessary, interference cancellation. The IC algorithm processes the correlation output blocks ($\mathbf{z_i}$ defined in (6)) prior to detection and verification. We detail both steps in the following. For notational simplicity, we drop the index i ($\mathbf{z_i}$ becomes $\mathbf{z}$) in the remainder of this section. We also denote by $C_{\mathrm{cross-peak}}$, the number of peaks in the cross-correlation (the peaks correspond to a cross-correlation value of 10 in Fig. 1), by C_{trough} the number of troughs (the throughs correspond to a cross-correlation value of 6 in Fig. 1), and by $C_{\mathrm{mid}} = C - C_{\mathrm{cross-peak}} - C_{\mathrm{trough}}$ the number of remaining values. A subblock corresponds to M consecutive samples within a block. There are C subblocks per block.
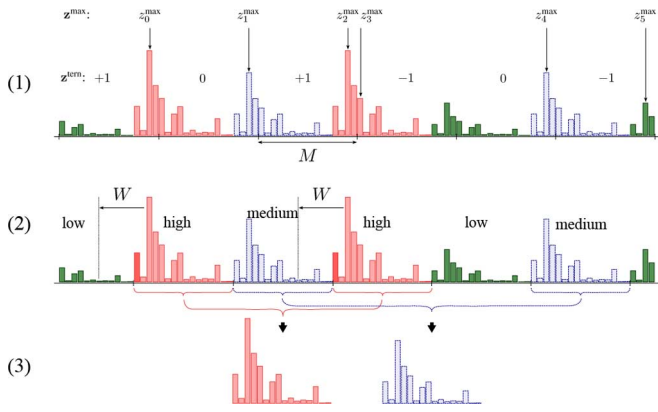
Fig. 3. Three steps of the PICNIC algorithm to cancel the effect of interfering code: (1) Interference is detected by matching the cross-correlation pattern to high-, mid- and low-energy blocks in the correlation output. (2) Time-base is aligned on the interferer to find beginning of blocks via a search-back algorithm. (3) Channel power-delay profile (PDP) to subtract is calculated separately for high-, mid- and low-energy blocks via a *robust* method based on order statistics.

1) Detection of an Interfering Preamble Code: If there is MUI, the cross-correlation peaks will likely generate $C_{\text{cross-peak}}$ subblocks with high energy within a block $\mathbf{z}$. Similarly, the cross-correlation troughs will generate C_{trough} subblocks with low energy and the remaining C_{mid} subblocks will have a medium energy.

To detect MUI, IC begins with the construction of a ternary vector $\mathbf{z}^{\text{tern}}$ that corresponds to the approximate location of each type of subblock within $\mathbf{z}$ (step 1 in Fig. 3). To obtain $\mathbf{z}^{\text{tern}}$, IC takes the maximum of consecutive groups of M samples in $\mathbf{z}$ to obtain an intermediate vector $\mathbf{z}^{\text{max}}$ of length C. For $j = 0, \ldots, C - 1$, the jth element of $\mathbf{z}^{\text{max}}$ is $z_j^{\text{max}} = \max\left[z_{Mj}, z_{Mj+1}, \ldots, z_{Mj+(M-1)}\right]$. This vector $\mathbf{z}^{\text{max}}$ is converted to $\mathbf{z}^{\text{tern}}$ by replacing its $C_{\text{cross-peak}}$ highest values with $+1$, its C_{trough} lowest values with -1 and the rest with 0. IC then also maps the cross-correlation (in Fig. 1 for codes 5 and 6) to a ternary vector $\mathbf{x}_{\text{cross}}^{\text{tern}}$ by replacing its $C_{\text{cross-peak}}$ highest values with $+1$, its C_{trough} lowest values with -1 and the rest with 0.

The vectors $\mathbf{z}^{\text{tern}}$ and $\mathbf{x}_{\text{cross}}^{\text{tern}}$ are now correlated and the maximum of the correlation is compared with an interference detection threshold $\eta_{\text{picnic}} = \left\lfloor \frac{C_{\text{cross-peak}} + C_{\text{trough}}}{2} \right\rfloor + 1$. This operation verifies that more than half of the peaks and troughs of the cross-correlation correspond to the peaks and troughs of $\mathbf{z}^{\text{tern}}$. If the maximum is above η_{picnic}, IC assumes MUI is present and continues its execution. Otherwise, timing acquisition proceeds.

2) Interference Cancellation by Subtraction of the Estimated Interference Power-Delay Profile: If MUI is present, the IC algorithm will attempt to subtract interference from $\mathbf{z}$. For each group of subblocks, the power-delay profile (PDP) of the interference will be estimated and then removed. To perform the estimation, the boundaries of the subblocks must be located. For this purpose, a JBSF procedure, similar to the one in Section III-B, is performed on each high-energy subblock:

1) Let us consider the set

$$\mathcal{C}_{\text{cp}} = \left\{ z_j^{\text{max}} \middle| z_j^{\text{max}} \in \mathbf{z}^{\text{max}} \text{ s.t. } z_j^{\text{tern}} = 1 \right\}$$

of samples of $\mathbf{z}^{\text{max}}$ corresponding to the $C_{\text{cross-peak}}$ high-energy subblocks (corresponding to the value 1 in $\mathbf{z}^{\text{tern}}$). Note that $|\mathcal{C}_{\text{cp}}| = C_{\text{cross-peak}}$. Now, for each element of $\mathcal{C}_{\text{cp}}$, we take its index in $\mathbf{z}$ and apply a modulo M operation to create the vector $\mathbf{k} = [k_0, k_1, \ldots, k_{C_{\text{cross-peak}}-1}]$. Hence, $k_j \in \{0, 1, \ldots, M-1\}$. A majority vote on $\mathbf{k}$ yields $\bar{k}$ and the set of indices $\mathcal{K}_{\text{cp}} = \left\{ jM + \bar{k} \mid \forall j \text{ s.t. } z_j^{\text{tern}} = 1 \right\}$.

2) From each element of $\mathcal{K}_{\text{cp}}$, i.e., for *each* high-energy subblocks, IC searches in a window of length $\frac{M}{2}$ the first sample above the noise threshold

$$\eta_{\text{picnic}}^{\text{jump}} = F_{\text{BIN}}^{-1}\left(1 - P_{\text{AWGN}}^{\text{FA,PICNIC}} | C_{\text{NZ}} N_{\text{G}}, P_{\text{AWGN}}^{\text{FA,PID}}\right) \quad (12)$$

where the parameter $P_{\text{AWGN}}^{\text{FA,PICNIC}}$ is the desired probability that AWGN can exceed the threshold. Each search procedure returns the index (in $\{0, 1, \ldots, \frac{M}{2}-1\}$) of the selected sample in the search window.

3) Let l be the index reported by more than half of these $C_{\text{cross-peak}}$ search procedures. The set $\left\{ jM + \bar{k} - \frac{M}{2} + l \mid \forall j \text{ s.t. } z_j^{\text{tern}} = 1 \right\}$ comprises the indices of the beginning of the high-energy subblocks.

The block $\mathbf{z}$ is now split at the exact boundaries into $C_{\text{cross-peak}}$ high-energy subblocks, C_{trough} low-energy subblocks and C_{mid} medium-energy subblocks. The signal is wrapped around if needed (see Fig. 3 for the first low-energy subblock).

To cancel interference, the following procedure processes the three groups of subblocks. Let's take high-energy subblocks as an example. An interference PDP $\mathbf{q}^{\text{high}} = \{q_0^{\text{high}}, q_1^{\text{high}}, \ldots, q_{M-1}^{\text{high}}\}$ is estimated for the high-energy subblocks with a *robust* estimation procedure [12]. Let us denote the mth sample of the jth high-energy subblock by $z_{j,m}^{\text{high}}$. We find $\mathbf{q}^{\text{high}}$ according to

$$q_m^{\text{high}} = \text{median}\left\{z_{j,m}^{\text{high}} : j \in \{0, 1, \ldots, C_{\text{cross-peak}}-1\}\right\} - \bar{v}_{\text{AWGN}}^{\text{PICNIC}}$$
$$(13)$$

where $\bar{v}_{\text{AWGN}}^{\text{PICNIC}} = C_{\text{NZ}} \cdot N_{\text{G}} \cdot P_{\text{AWGN}}^{\text{FA,PID}}$ is the expected noise level at the output of the correlation in (6). The median in (13) is used for robustness to outliers [12], typically, the UOI signal that we do *not* want to subtract. Further details on the robust estimation procedure are available in [12], [24]. To cancel interference, $\mathbf{q}^{\text{high}}$ is simply subtracted from all the high-energy subblocks in $\mathbf{z}$. The procedure is replicated for the medium- and low-energy subblocks.

IV. ONLINE ALGORITHMS FOR START FRAME DELIMITER DETECTION ROBUST TO MUI

After packet detection and timing acquisition, the receiver knows the beginning of a preamble symbol. It performs channel estimation, but does not know yet when the payload precisely begins. Hence, the receiver looks for an SFD: a known sequence that marks the end of the preamble and the beginning of the payload. Assuming that packet detection and timing acquisition were performed perfectly, the propagation delay is of the form $\nu_0 = N_\Delta \cdot CLT_c$, i.e., the uncertainty on the number of preamble symbols N_Δ used during timing acquisition remains. To simplify notation, we make the equivalent assumption that

$\nu_0 = 0$ but that the number of preamble symbols N_{sync} sent in the SYNC part of the preamble is unknown.

Earlier work [11] developed SFD detection algorithms using a correlation-based approach. However, the approach we follow, as in [19], treats SFD detection as a decoding problem: The receiver decodes N_{sfd} consecutive received preamble symbols and tries to determine whether they correspond to the squared SFD sequence $\mathbf{s}^{2(\text{sfd})} = [s_0^{2(\text{sfd})}, \ldots, s_{N_{\text{sfd}}-1}^{2(\text{sfd})}]$. Squaring occurs because of noncoherent operation. As simulation results show in Section V-B, this approach is superior to a correlation-based one. Novelty with respect to [19] in our approach is threefold: 1) detection is online, which greatly simplifies implementation and reduces memory requirements, 2) detection uses soft-decision, and 3) a postprocessing on the received signal is used to provide robustness against MUI.

A. DESSERT: Detection of SFD Through Sequential Ratio Tests

The online property of DESSERT stems from a sequential processing of blocks of N_{sfd} preamble symbols where N_{sfd} is the length of an SFD code (see Section II-B-3)). Hence a block corresponds to $N_{\text{sfd}} \cdot C \cdot M$ consecutive samples. The kth block is denoted by $\mathbf{y_k}$ where

$$\mathbf{y_k} = \big[y_{kCM}, y_{kCM+1}, \ldots, y_{(k+1)CM-1},$$
$$y_{(k+1)CM}, y_{(k+1)CM+1}, \ldots, y_{(k+N_{\text{sfd}})CM-1}\big]. \quad (14)$$

Thanks to the structure of the preamble in (2), there exists only $N_{\text{sfd}} + 1$ sequences $\mathbf{s_k^2}$ than can be observed, namely

$$\mathbf{s_k^2} = \begin{cases} \mathbf{s}^{2(\text{sfd})} = \\ [s_0^{2(\text{sfd})}, s_1^{2(\text{sfd})}, \ldots, s_{N_{\text{sfd}}-1}^{2(\text{sfd})}] & \text{if } k = N_{\text{sync}} \\ [1, s_0^{2(\text{sfd})}, \ldots, s_{N_{\text{sfd}}-2}^{2(\text{sfd})}] & \text{if } k = N_{\text{sync}} - 1 \\ [1, 1, s_0^{2(\text{sfd})}, \ldots, s_{N_{\text{sfd}}-3}^{2(\text{sfd})}] & \text{if } k = N_{\text{sync}} - 2 \\ \cdots & \cdots \\ [1, 1, \ldots, 1, s_0^{2(\text{sfd})}] & \text{if } k = N_{\text{sync}} - N_{\text{sfd}} + 1 \\ [1, 1, \ldots, 1] & \text{otherwise.} \end{cases} \quad (15)$$

Note that N_{sfd} is usually small ($N_{\text{sfd}} = 8$ for the mandatory mode of the IEEE 802.15.4a). Hence, for each new received preamble symbol, DESSERT calculates N_{sfd} log-likelihood ratio (LLR) tests (one for each possible sequence $\mathbf{s_k^2}$). Calculation of the likelihood relies on the assumption of independence between the samples y_n (as in [19] and [23]).

1) Derivation of the Likelihood Function of a Block $\mathbf{y}_k$: We know from Section II that the distribution of the energy-detection receiver output can be approximated by a scaled noncentral chi-square distribution. Hence the likelihood function for the kth block $\mathbf{y_k}$ given the parameters $\boldsymbol{\Theta} = (\frac{N_0}{2}, BT, q_m)$ and the sequence $\mathbf{s_k^2}$ is

$$f(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s_k^2}) = \prod_{i,j,m} \frac{1}{\frac{N_0}{2}} f_{\text{NC}\chi^2}$$
$$\times \left(\frac{y_{(k+i)CM+jM+m}}{\frac{N_0}{2}}\Big|2BT, \frac{s_{k+i}^2 c_j^2 q_m}{\frac{N_0}{2}}\right) \quad (16)$$

where $i \in \{0, \ldots, N_{\text{sfd}} - 1\}$, $j \in \{0, \ldots, C - 1\}$ and $m \in \{0, \ldots, M - 1\}$. All parameters are either known (B and T) or can be estimated robustly (N_0 and q_m).

2) Soft-Decision Decoding Likelihood-Ratio Test for SFD Detection: Therefore, for a block $\mathbf{y_k}$, DESSERT will declare presence of the SFD if $\mathbf{s}^{2(\text{sfd})} = \arg \max_{\mathbf{s_k^2}} f(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s_{k^2}})$. This is equivalent to $\ln(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s}^{2(\text{sfd})})) \geq \ln(f(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s_k^2})), \forall \mathbf{s_k^2}$ and can be expressed as N_{sfd} log-likelihood ratio (LLR) tests $\text{LLR}_{\text{sfd}}(\mathbf{y}_k|\boldsymbol{\Theta}, \mathbf{s}_k^2) \geq 0, \forall \mathbf{s_k^2}$ where

$$\text{LLR}_{\text{sfd}}(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s_k^2})$$
$$= \ln\big(f(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s}^{2(\text{sfd})})\big) - \ln\big(f(\mathbf{y_k}|\boldsymbol{\Theta}, \mathbf{s_k^2})\big)$$
$$= \sum_{\substack{i \\ s_i^{2(\text{sfd})} \neq s_{k+i}^2}} (2s_i^{2(\text{sfd})} - 1)$$
$$\times \sum_{j,m} \text{LLR}\left(\frac{y_{(k+i)CM+jM+m}}{\frac{N_0}{2}}\Big|2BT, \frac{c_j^2 q_m}{\frac{N_0}{2}}\right) \quad (17)$$

with

$$\text{LLR}\left(\frac{y_{\tilde{n}+m}}{\frac{N_0}{2}}\Big|2BT, \frac{q_m}{\frac{N_0}{2}}\right) = \ln\left[\frac{f_{\text{NC}\chi^2}\left(\frac{y_{\tilde{n}+m}}{\frac{N_0}{2}}\Big|2BT, \frac{q_m}{\frac{N_0}{2}}\right)}{f_{\chi^2}\left(\frac{y_{\tilde{n}+m}}{\frac{N_0}{2}}\Big|2BT\right)}\right]$$
$$= \ln\left[{}_0F_1\left(;BT;\frac{q_m y_{\tilde{n}+m}}{N_0^2}\right)\right]$$
$$- \frac{q_m}{N_0} \quad (18)$$

where $\tilde{n}$ is an integer multiple of M and ${}_0F_1(;a;x)$ denotes the confluent hypergeometric limit function.

For implementation purpose, the function $\ln\left[{}_0F_1\left(;BT;\frac{q_m y_{\tilde{n}+m}}{N_0^2}\right)\right]$ can be tabulated. Further, with the assumption of independent samples y_n, the contribution of every sample to the likelihood (16) can be computed individually, sample by sample.

If the SFD is declared present, the algorithm stops and payload decoding can begin. Otherwise the algorithm continues with the next preamble symbol. If the SFD is not found during a maximum of N_{sync} preamble symbols, the algorithm has likely missed the SFD. Reception of the packet is aborted and the receiver goes back to packet detection.

3) Hard-Decision Decoding Likelihood-Ratio Test for SFD Detection: A hard-decision decoding version of the above likelihood-ratio is also feasible. In this case, every preamble symbol is decoded individually with

$$\sum_{j,m} \text{LLR}\left(\frac{y_{kCM+jM+m}}{\frac{N_0}{2}}\Big|2BT, \frac{c_j^2 q_m}{\frac{N_0}{2}}\right) \overset{\hat{s}_k^2=1}{\underset{\hat{s}_k^2=0}{\gtrless}} 0 \quad (19)$$

to yield a sequence $\hat{s}_k^2 = (\hat{s}_k^2, \ldots, \hat{s}_{k+N_{\text{sfd}}-1}^2)$ of decoded preamble symbols $\hat{s}_k^2$. DESSERT then calculates the Hamming distance between $\hat{s}_k^2$ and each of the possible sequences given by (15). If the sequence closest to $\hat{s}_k^2$ is $\mathbf{s}^{2(\text{sfd})}$ and no other sequence is equally close, detection of the SFD is declared and the algorithm stops.

B. Improving Robustness to MUI of SFD Detection Algorithms

For robustness against MUI, the PDP q_m needs to be estimated in a robust fashion, for instance according to the method in [12] which we use in this paper.

In addition, an adaptive thresholding scheme to reject or limit strong interference terms is employed during SFD detection. The receiver calculates the adaptive threshold

$$\eta_m^{\text{sfd}} = \frac{N_0}{2} F_{\text{NC}\chi^2}^{-1} \left(1 - P_{\text{MUI}}^{\text{FA,SFD}} | 2BT, \frac{q_m}{\frac{N_0}{2}} \right) \qquad (20)$$

with parameter $P_{\text{MUI}}^{\text{FA,SFD}}$ to govern the nonlinearity

$$g\left(y_{\tilde{n}+m} | q_m, \frac{N_0}{2} \right) = \begin{cases} y_{\tilde{n}+m} & \forall m : y_{\tilde{n}+m} \leq \eta_m^{\text{sfd}} \\ q_m & \forall m : y_{\tilde{n}+m} > \eta_m^{\text{sfd}}. \end{cases} \qquad (21)$$

The nonlinearity (21) is then used in place of $y_{\tilde{n}+m}$ in (18).

V. PERFORMANCE EVALUATION

We use a IEEE 802.15.4a packet-based simulator, implemented with Matlab. The *simulation* accuracy is 10 picoseconds. We simulate one receiver and N_{u} transmitters (the UOI and $N_{\text{u}} - 1$ interferers). All transmitters generate packets according to a Poisson process with rate $R = 100$ packets/s. At the maximum allowed packet size, with a payload of 1016 bits, and with the IEEE 802.15.4a backoff procedure [3], this corresponds to roughly half the peak rate [11]. We simulate two types of interference scenarios: a near–far scenario with $N_{\text{u}} - 1$ interferers with power levels 10 dB higher than the UOI, or an equal-power scenario with $N_{\text{u}} - 1$ interferers power levels equal to the UOI.

We consider the mandatory LPRF mode [3] of IEEE 802.15.4a , band number 3 and corresponding preamble codes 5 and 6. With the HPRF mode, or other bands, all our findings are similar and are therefore omitted. For the preamble, we use default values: $N_{\text{sync}} = 64$ and $N_{\text{sfd}} = 8$. The integration time T is equal to $T_c = 2$ ns. Our algorithms allow for longer integration times.

Propagation is modeled according to the IEEE 802.15.4a residential non line-of-sight (NLOS, CM2) or the office line-of-sight (LOS, CM3) channel models [25]. However, the results for NLOS and LOS are very similar and we concentrate on NLOS results. Finally, we define the signal-to-noise ratio (SNR) as $\text{SNR} = \frac{E_p}{N_0}$ where E_p is the received energy per pulse (after the convolution of the pulse with the impulse response of the channel). Confidence intervals are at the 95% level.

Furthermore, we compare our algorithms with a *conventional* algorithm that performs a correlation between the received samples and the template (defined by (7)) but does not apply the threshold η_{pid}. A maximum selection and threshold crossing criterion [26] is applied to the correlation output for packet detection and initial timing acquisition. The detection threshold is based on the AWGN noise distribution at the correlator output and given by

$$\eta_{\text{detect}} = \frac{N_0}{2} F_{\chi^2}^{-1} (1 - P_{\text{AWGN}}^{\text{FA}} | 2BT \cdot C_{\text{NZ}} \cdot N_{\text{G}}),$$

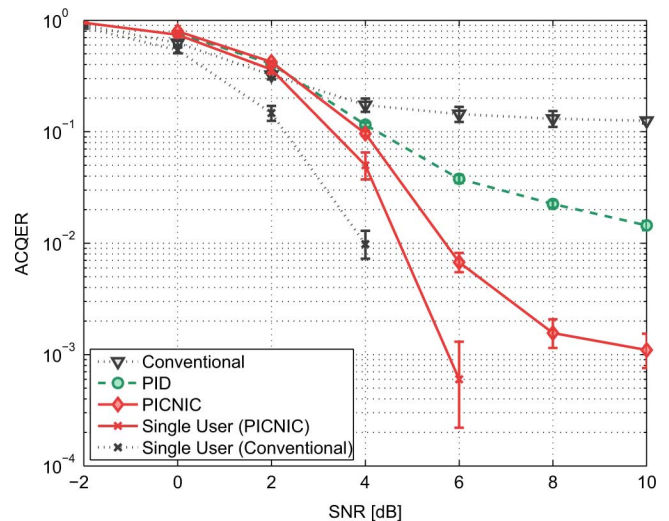$$\text{with} \quad P_{\text{AWGN}}^{\text{FA}} = 1e^{-3}. \qquad (22)$$

Fig. 4. ACQER for the different algorithms with two interferers in the near–far scenario. The preamble codes of the interferers differ from the one of the UOI. The conventional algorithm is not robust. PID is able to reduce strong interference. Interference due to imperfect cross-correlation is only reduced by the PICNIC method, yielding a gain of up to two orders of magnitude.

As in PICNIC, any initial synchronization point is verified with N_{V} consecutive blocks. Timing acquisition is performed using the JBSF procedure in Section III-B with the only difference that the threshold (11) is replaced by η_{detect}.

For system results that present packet error rate (PER) results using PICNIC and DESSERT, the reader can consult [24] and the references therein.

A. Performance of Packet Detection and Timing Acquisition

For packet detection and timing acquisition, the receiver uses a template with $N_{\text{G}} = 10$ and performs $N_{\text{V}} = 16$ verification steps. These parameters, found by simulations, keep timing acquisition fast enough to manage to perform channel estimation on the preamble. Channel estimation is performed according to [12] and lasts $N_{\text{CH}} = 16$ preamble symbols.

The default parameters used to set the sensitivity of the various thresholds during timing acquisition are $P_{\text{AWGN}}^{\text{FA,PID}} = 0.2$, $P_{\text{AWGN}}^{\text{FA,PID,fine}} = 1e^{-4}$, $P_{\text{AWGN}}^{\text{FA,PICNIC}} = 1e^{-3}$. Extensive simulations show that, as long as $P_{\text{AWGN}}^{\text{FA,PID}}$ is not set too low, a wide range of values give a similar performance (see Section V-A-3)).

Our main performance metric is the packet acquisition error rate (ACQER) that includes FA *and* MD: A packet is correctly acquired if the receiver synchronizes on a multipath component of the correct preamble code symbol. For comparison, we also show the performance of the PICNIC algorithm without the IC algorithm, which is referred to as the power independent detection (PID) algorithm.

1) Packet Detection and Initial Timing Acquisition Performance: Fig. 4 shows the performance of the different algorithms in the near–far scenario with two interferers using preamble code 6. The UOI uses code 5. The conventional algorithm is not robust, more than 10% of the packets are lost due to interference. The PID method reduces the effect of large interference and the ACQER is improved by roughly one order of magnitude at high SNR. Still, even with the PID and the
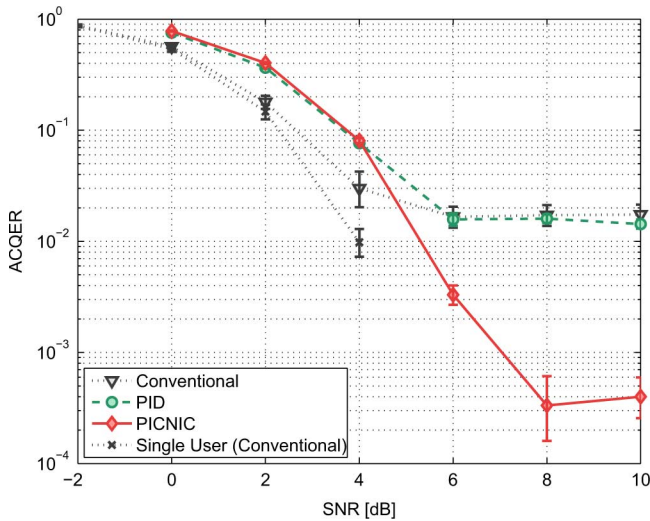
Fig. 5. ACQER for the different algorithms with two interferers in the equal-power scenario. The preamble codes of the interferers differ from the one of the UOI. The PID and conventional algorithm perform identically because all of the errors are caused by interference due to imperfect cross-correlation. This type of interference is only reduced by the PICNIC method, yielding again a significant improvement.

use of different preamble codes, FAs occur due to the imperfect cross-correlation. PICNIC addresses this issue and gains another order of magnitude. For reference purpose, we show the single-user performance of both the conventional algorithm and PICNIC in Fig. 4. The single-user performance of PID is identical to PICNIC. PICNIC performs slightly worse than the conventional algorithm because the threshold η_{pid} removes some signal information.

Fig. 5 shows the performance in the equal-power scenario for two interferers with different codes. In the interference limited SNR regions, PID and the conventional algorithms have equal performance and exhibit an error floor due to the imperfect cross-correlation. In this case, interference is not high enough to dominate the output of the correlation. PID does not improve the performance because all the acquisition errors are due to the imperfect cross-correlation. PICNIC again significantly reduces this type of interference.

If all transmitters use the same preamble codes in the previous scenario, many FAs occur because the receiver cannot distinguish an interfering signal from the signal of the UOI. Independently of the algorithms used, the ACQER is consequently very high (simulations yield 15%). Hence, a more meaningful metric is needed in order to allow for the quantification of the capture effect capabilities of the receiver. We define the capture error rate (CER) as the probability that a UOI packet is lost *and* the receiver does not correctly acquire an interfering packet either. Our simulation results show that if two packets arrive at the receiver at about the same time, the one with higher power is usually acquired. The worst-case scenario is when these two packets have similar power levels. The verification phase may then never succeed because the receiver switches back and forth between the two packets. Hence, we evaluate this scenario by simulating an equal-power interferer that is always present and always starts at about the same time as the UOI. The results are
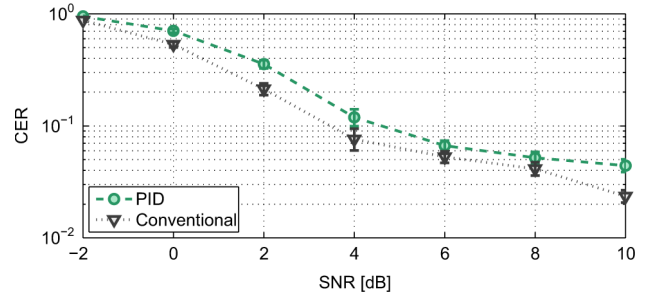

Fig. 6. Worst case scenario to assess the capture effect: One equal-power interferer using the same code than the UOI. Furthermore, the interferer is always present and starts its transmission at about the same time as the UOI. Still, the probability that we acquire neither of the two remains below 5%.
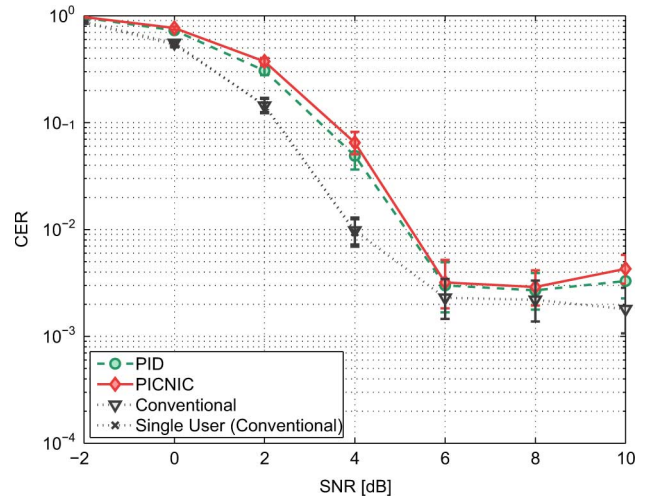

Fig. 7. Probability of neither acquiring the UOI nor an interferer for two interferers with same code and equal-power level as the UOI. All algorithms show a good capture effect.

shown in Fig. 6. For both the conventional algorithm and PID (PICNIC is not shown because it coincides with PID if identical preamble codes are used), capture is above 95% at high SNR. In fact, even though the two users have the same power level, the received energies are distributed differently because of the different propagation channels. Further, we see that the conventional algorithm performs even a bit better than PID. PID, to some extent, levels out different power levels through the thresholding operation on the correlation input. Fig. 7 shows similar results with the equal-power scenario with identical preamble codes but with three users that generate packets according to a Poisson process and use the IEEE 802.15.4a Aloha back-off procedure. All algorithms show a good capture effect.

2) Timing Acquisition Performance: Figs. 8 and 9 show the empirical probability density function (PDF) of the timing acquisition error after the JBSF algorithm for, respectively, the conventional algorithm and PICNIC. A near–far scenario with different codes and SNR of 10 dB is used. The PDFs are obtained with packets that are correctly acquired during the packet detection and initial timing acquisition. The search-back window, fixed to $W = \frac{M}{2}$ (64 ns) in our simulations, creates the tail at the left of the distribution. We can clearly see in Fig. 8 that the conventional algorithm is vulnerable to FAs
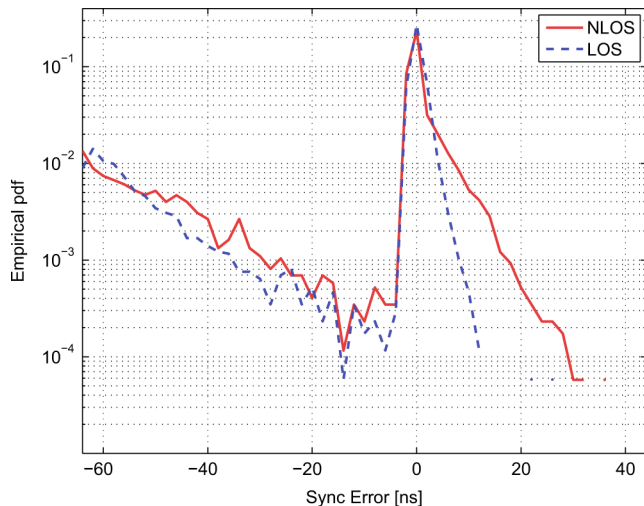
Fig. 8. Empirical PDF of the synchronization error after timing acquisition with the conventional algorithm. Results shown are for the near–far scenario at SNR = 10 dB. FAs occur at the beginning of the search-back window of size $W = \frac{M}{2}$ corresponding to a duration of 64 ns.
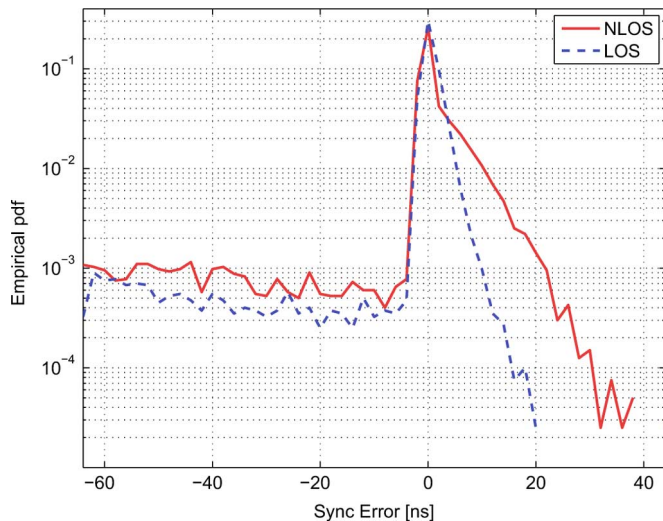

Fig. 9. Empirical PDF of the synchronization error after timing acquisition with the PICNIC algorithm (near–far scenario at $SNR = 10$ dB). PICNIC prevents FAs, resulting in more accurate synchronization.

during timing acquisition. These FAs are caused by high interference terms and lead the receiver to synchronize too early. Fig. 9 shows that PICNIC definitely limits FAs. As expected, synchronization accuracy is better with the LOS channel.

Table I shows that despite our algorithms are optimized for communication rather than ranging, especially the choice of W, we obtain an excellent synchronization accuracy, even with severe interference. Results are obtained with the near–far scenario with different codes and NLOS. All values are only calculated over correctly acquired packets. The conventional algorithm performs poorly because FAs occur due to large interfering terms exceeding the threshold. PID has a slightly better accuracy, because PICNIC acquires more packets with interference.

3) PICNIC Sensitivity Analysis for $P_{\mathrm{Awgn}}^{\mathrm{Fa,pid}}$, $P_{\mathrm{Awgn}}^{\mathrm{Fa,pid,fine}}$, and $P_{\mathrm{Awgn}}^{\mathrm{Fa,picnic}}$: This section presents performance evaluation results of PICNIC for various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$, $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$

TABLE I
PRECISION OF THE SYNCHRONIZATION FOR NLOS AND TWO NEAR INTERFERERS

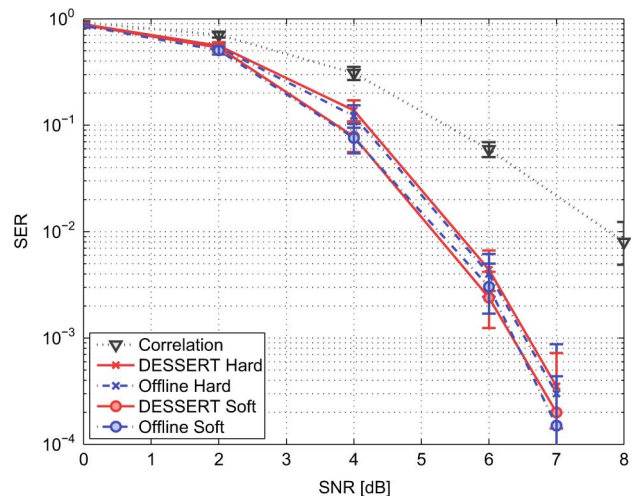| Algorithm	RMSE	Mean	50%	75%	90%
Baseline	22.8ns	11.0ns	< 1.1ns	< 5.5ns	< 52.6ns
PID	9.2ns	3.7ns	< 0.9ns	< 2.9ns	< 9.1ns
PICNIC	10.0ns	4.0ns	< 0.9ns	< 2.9ns	< 9.5ns

Fig. 10. Single-user performance of DESSERT with soft- and hard-decision decoding and comparison with the offline algorithms from [19]. DESSERT performs very close to the offline algorithms.

and $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}}$. Remember that default values used in the remainder of Section V are $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}} = 0.2$, $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}} = 1e^{-4}$, and $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}} = 1e^{-3}$.

Figs. 11, 12, and 13 show the ACQER with $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}} = \{0.10, 0.15, 0.20, 0.25, 0.30, 0.35\}$ for the single-user scenario, the near–far scenario and the equal-power scenario respectively. Default values are used for $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$ and $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}}$. Figs. 12 and 13 also show the single-user performance for reference purpose ($P_{\mathrm{AWGN}}^{\mathrm{FA,PID}} = 0.2$). The performance of PICNIC remains stable with all values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$. The default value of 0.2 exhibits the best performance over a wide range of SNR values.

Figs. 16 and 17 shows the ACQER for $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}} = \{1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}\}$ for the near–far scenario and the equal-power scenario respectively. Default values are used for $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$ and $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$. We also show the single-user performance for reference purpose (with $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}} = 1e^{-3}$). The performance of PICNIC remains stable with all values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}}$. The default value of $1e^{-3}$ exhibits the best performance over a wide range of SNR values. Fig. 14 shows the empirical PDF of the timing acquisition error for the near–far scenario with two interferers at SNR = 10 dB for $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}} = \{1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}\}$. The best performance is achieved with the default value of $1e^{-4}$ and decreases with increasing values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$. Fig. 15 addresses the equal-power scenario. The conclusion is similar than for the near–far scenario.
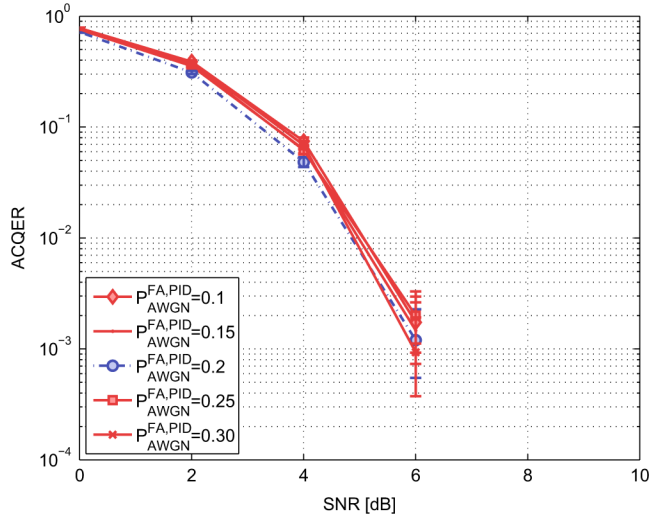
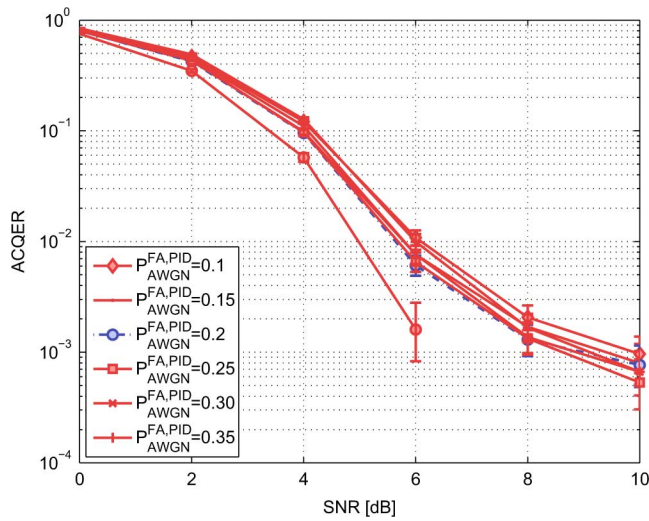Fig. 11. ACQER for single-user scenario with various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$.

Fig. 12. ACQER for near–far scenario with two interferers and various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$.
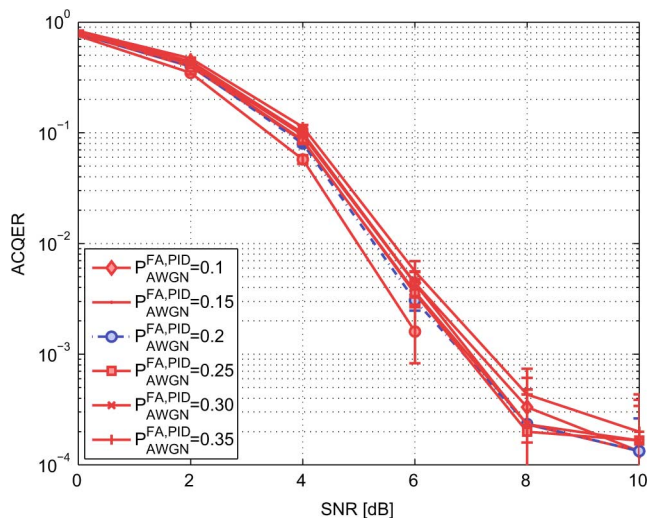

Fig. 13. ACQER for equal-power scenario with two interferers and various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID}}$.

Fig. 14. Empirical PDF of the synchronization error after timing acquisition with the PICNIC algorithm (near–far scenario at SNR = 10 dB) for various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$.
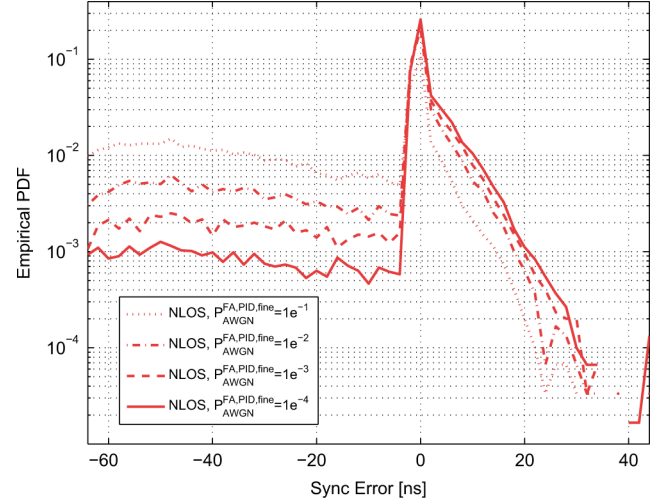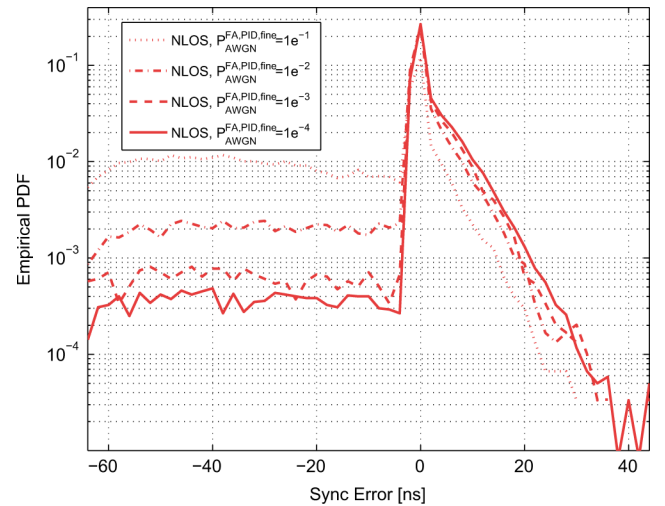

Fig. 15. Empirical PDF of the synchronization error after timing acquisition with the PICNIC algorithm (equal-power scenario at SNR = 10 dB) for various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PID,fine}}$.

B. SFD Detection and Overall Synchronization Performance

We evaluate the performance of DESSERT for SFD detection. All of the following results include packet detection and timing acquisition, performed either with the conventional algorithm or PICNIC. Therefore, the results show the overall synchronization performance, which we measure with the synchronization error rate (SER): the percentage of packets that were missed because of synchronization errors. It includes both FAs and MDs. We also simulate i) the offline SFD detection algorithm with hard-decision from [19], and ii) a soft-decision version of [19] by borrowing the soft metric (17), yielding

$$\mathrm{LLR}_{\mathrm{sfd}}^{\mathrm{off}}(\mathbf{y_k}|\boldsymbol{\Theta}) = \sum_{i,j,m} \left(2s_i^{2(\mathrm{sfd})} - 1\right)$$

$$\cdot \mathrm{LLR}\left(\frac{y_{(k+i)CM+jM+m}}{\frac{N_0}{2}}\bigg|2BT, \frac{c_j^2 q_m}{\frac{N_0}{2}}\right). \quad (23)$$

With this soft metric, SFD detection is decided on the index k for which (23) is maximal. An offline algorithm does not detect the SFD in a sequential fashion. Rather, it stores N_{mem} de-
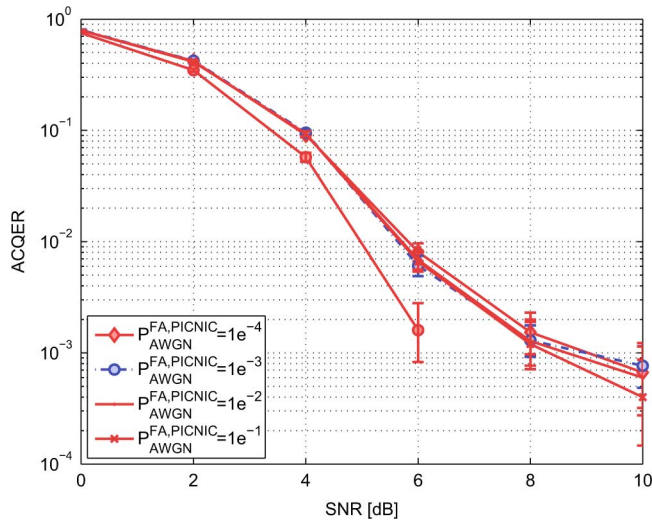
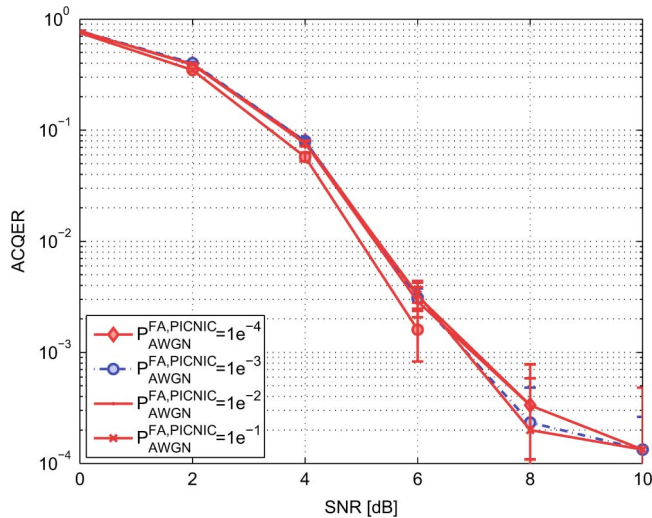Fig. 16. ACQER for near–far scenario with two interferers and various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}}$.

Fig. 18. Performance of DESSERT online algorithms and the offline algorithms from [19] in a near–far scenario with three users. None of the algorithms uses thresholding to mitigate high interference terms. The only algorithm that is robust to MUI is the offline algorithm with soft metric.

Fig. 17. ACQER for equal-power scenario with two interferers and various values of $P_{\mathrm{AWGN}}^{\mathrm{FA,PICNIC}}$.

coded preamble symbols $\hat{s}_k^2$ in memory. N_{mem} has to be chosen *large* enough to ensure that the stored samples contain the entire SFD. This makes the search window, within which the algorithm looks for the SFD, extends into the payload of the packet. Consequently, the SFD is not detected instantly but only after an important part of the payload has been received, leading to a more complex receiver that has to perform decoding and SFD detection partly in parallel.

1) Single-User Performance: Fig. 10 shows the performance of the different SFD detection algorithms without MUI. Timing acquisition is performed with the conventional algorithm because it has the best single-user performance.

The performance of DESSERT is almost undistinguishable from the offline algorithm. For both DESSERT and the offline algorithms, a soft metric performs negligibly better than a hard metric.

We also report the results of the correlation-based algorithm of [11]. We observe that it is outperformed by likelihood-based
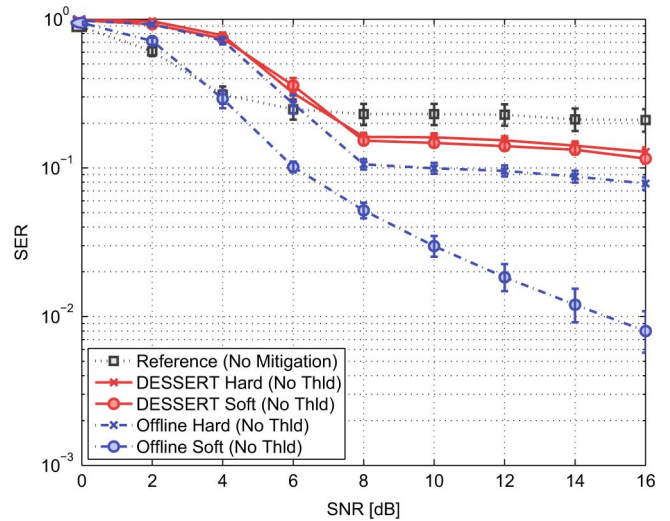
algorithms. Because there is already a considerable performance difference of roughly 2.5 dB in the single-user case, we will not show further results with correlation-based algorithms.

2) SFD Detection With MUI But No Interference Mitigation: To assess the performance with MUI, we consider a near–far scenario with $N_u = 3$. For reference, we also show the performance of a receiver that does not perform any form of interference mitigation. This receiver uses the conventional algorithm for timing acquisition, estimates the channel parameters with an algorithm [23] not robust to MUI and performs SFD detection with the DESSERT algorithm with soft-decision and no thresholding.

Fig. 18 shows the SER for the four algorithms in the near–far scenario if no thresholding is used to reject interference during SFD detection. PICNIC is used for timing acquisition and channel estimation is performed in a robust fashion. Both online algorithms perform similarly and they are not robust to MUI. Robust channel estimation alone gives only a minor performance improvement compared to the reference receiver, making any gain achieved during the timing acquisition void. The offline algorithm with hard-decision shows a slight gain but is not very robust to MUI either. The offline algorithm from [19], augmented with soft-decision, outperforms all others. It already shows a decent robustness against MUI. At high SNR, it achieves a SER improvement of more than one order of magnitude with respect to the reference receiver. At low SNRs it outperforms the other algorithms by 2 dB. The improvement comes with an increased complexity and memory requirement because of offline operation.

3) SFD Detection With MUI and Interference Mitigation: Fig. 19 shows again the same near–far scenario with all receivers using thresholding to reject high interference terms during SFD detection (except the reference one). The parameter $P_{\mathrm{MUI}}^{\mathrm{FA,SFD}}$ is set to $= 0.01$. This value was found through simulations and shows a good performance over various interference scenarios.

TABLE II
COMPLEXITY AND MEMORY REQUIREMENTS OF THE CONVENTIONAL ALGORITHM, PICNIC AND SFD DETECTION

|	Memory (bits)	Operations (per sample)		
		add/subtract	multiply/divide	compare
Conventional				
Packet Detection	$N_{G}MCK$	$C_{NZ} + N_{G} - 2$	-	1
PICNIC				
Packet Detection	$N_{G}MCK$	$C_{NZ} + N_{G} - 2$	-	2
Preamble Code Interference Cancellation				
Interference Detection	$(M+1)CK$	-	-	$\frac{(M-1+\log_2 C)C}{MC} \approx 1$
First Path Selection	-	-	-	$\frac{M(C_{\text{cross-peak}}+1)}{2MC} \approx \frac{1}{6}$
Cancellation	$3MK$	1	-	$O(3)$
SFD Detection				
DESSERT soft	$(N_{\text{sfd}}+2M+K)K$	$\frac{C_{NZ}}{C} + \frac{N_{\text{sfd}}^2}{MC} \approx \frac{1}{2}$	$\frac{2C_{NZ}}{C} \approx 1$	$\frac{N_{\text{sfd}}}{MC} \approx 0$
DESSERT soft, robust	$(N_{\text{sfd}}+2M+K)K$	$\frac{C_{NZ}}{C} + \frac{N_{\text{sfd}}^2}{MC} \approx \frac{1}{2}$	$\frac{2C_{NZ}}{C} \approx 1$	$\frac{C_{NZ}}{C} + \frac{N_{\text{sfd}}}{MC} \approx \frac{1}{2}$
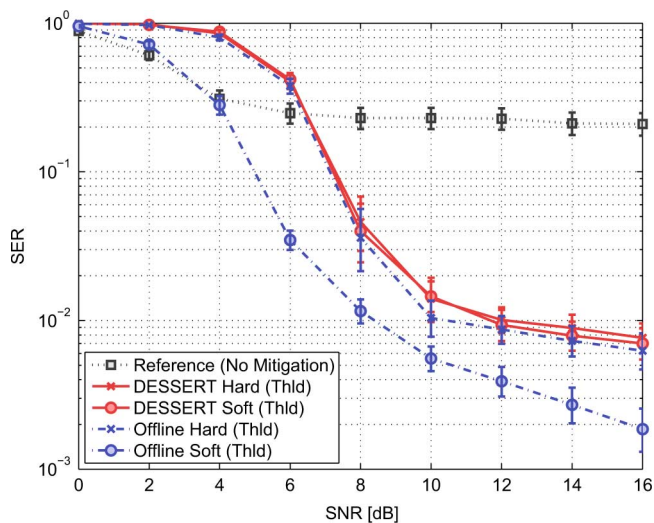
Fig. 19. Scenario of Fig. 18 but now all of the algorithms apply a threshold to mitigate interference. They all show a good robustness against MUI but the soft offline algorithm again has a performance advantage throughout the whole SNR range.

DESSERT profits the most from thresholding, performing now over one order of magnitude better than the reference receiver. Offline algorithms also improve with thresholding: hard-decision performs similarly to the online algorithms; soft decision is now two orders of magnitude better than the reference receiver in the interference limited SNR region.

We do not display results for an equal-power scenario because none of the algorithms is severely affected by this weaker interference. All have a similar performance, independently of whether a threshold is used during SFD detection. The resulting SER is close to the corresponding ACQER of PICNIC in Fig. 5. In the equal-power scenario, there are hardly any errors due to SFD detection; most of the remaining errors are due to timing acquisition.

Other nonlinear operations than (21) are of course also possible. We considered erasing samples that exceed the threshold such that they do not contribute to the decision. However, we could not find any noticeable performance difference between different schemes.

4) Robustness to MUI: What Is the Best SFD Detection Algorithm?: First, for robustness to MUI, PICNIC must be used for initial packet detection and timing acquisition. Then, all four algorithms can provide robustness against MUI. However, using the offline algorithm with hard-decision is not advisable; it performs equivalently to DESSERT but with higher complexity and memory requirements. Offline with soft-decision performs best but complexity and memory requirements are high. Finally, DESSERT with soft- or hard-decision perform similarly, both in terms of complexity and performance. They remain within 2 dB of an offline solution with soft-decision and offer the best trade-off between performance and complexity.

C. Complexity and Memory Requirements Analysis of PICNIC and DESSERT

Table II contains a summary of the complexity and memory requirements of PICNIC and DESSERT with soft-decision decoding. We also provide comparative results for a conventional algorithm. Details of the analysis are given in Appendix A.

With our simulation parameters and assuming similar complexity for comparators and adders, PICNIC requires roughly 20% to 30% more operations and 40% more memory than a conventional algorithm when MUI is detected. If no MUI is detected, the overhead is 8% more operations and 10% more memory. Compared to packet detection, SFD detection requires much less complexity and memory requirements.

VI. RELATED WORK

The usage of thresholding can be traced back to the late seventies in [27]. Work exploring the design of IR-UWB receiver structures for detection (but not synchronization) under non-Gaussian interference can be found in [12] and [28]–[31] and the survey [32]. Because we concentrate on synchronization and their architectures are coherent (except for [12]), their results are not applicable to our setting.

In the case of synchronization, ranging or time-of-arrival (TOA) estimation with MUI, there are several related

papers. A thresholding mechanism to mitigate MUI is used for packet detection with an analog and coherent receiver in [33]. This mechanism is not applicable to our setting since we use an ED architecture. In addition, optimal thresholds in [33] are found through exhaustive simulations only and it remains unanswered how to adapt these thresholds in practice. Our work solves the problem of threshold adaptation and additionally considers SFD detection. To suppress MUI during TOA estimation, [34] and [35] use nonlinear image filtering techniques. They assume packet detection and timing acquisition is already achieved. Both works are based on energy-detection and are complementary to our work. Algorithms for ranging for IEEE 802.15.4a are discussed in [17], [36], and [37]. In Section V-B, we already referred to [18] and [19], where offline SFD detection with hard-decision is considered in addition to ranging. However, MUI is not considered in [17]–[19], [36], and [37]. As in [17] and [19], we use a JBSF procedure. The major difference is that a novel detection criterion is applied (see (10)). We previously evaluated the performance of correlation-based SFD detection algorithms [11]. As shown in Section V-B, their performance is poor compared to our present work. To the best of our knowledge, there is no other work than [11], [18], and [19] that addresses SFD detection for IR-UWB.

In comparison with the references cited in this Section, our work encompasses *all* components of synchronization, namely packet detection, timing acquisition and SFD detection. Furthermore, our algorithms are designed to operate even in the presence of strong MUI.

VII. SUMMARY AND CONCLUSION

This paper presents PICNIC and DESSERT, two complementary low-complexity algorithms for reliable and robust synchronization with an IR-UWB energy-detection receiver in the presence of MUI. PICNIC addresses packet-detection and fine-timing acquisition and DESSERT addresses SFD detection. PICNIC novelty and robustness to interference stems from 1) a preprocessing prior to the correlation of the received signal using thresholding and 2) a postprocessing of the correlation output with an interference detection and cancellation mechanism. In addition, PICNIC exhibits a near perfect capture property, i.e., one signal is acquired with practically no false detection even if several transmitters compete for timing acquisition at a receiver. Furthermore, PICNIC solves a particular performance anomaly of timing acquisition with energy-detection; the anomaly is created by the nonperfect cross-correlation properties of the preamble codes when using energy-detection receivers. DESSERT uses a likelihood-based approach for SFD detection. Novelty in DESSERT is threefold: 1) detection is online, which greatly simplifies implementation and reduces memory requirements, 2) detection uses soft-decision, and 3) a postprocessing on the received signal is used to provide robustness against MUI. Our performance evaluation with extensive and detailed simulations shows that, at the cost of a negligible performance reduction in single-user scenarios, our algorithms outperform nonrobust synchronization algorithms by up to two orders of magnitude if MUI is present.

APPENDIX
COMPLEXITY AND MEMORY CONSUMPTION ANALYSIS OF PICNIC AND DESSERT

Table II contains a summary of our findings. With $\frac{1}{T}$ samples per second at the analog-to-digital converter (ADC) output, with a resolution of K bits per sample, we obtain $\frac{K}{T}$ bits/s. The thresholds involved in any of the algorithms do not depend on the ADC output samples and are computed at most once per packet. Their related complexity and memory requirements are negligible.

A. Complexity and Memory Requirements for a Conventional Correlation-Based Algorithm

1) Packet Detection: With $C_{\mathrm{NZ}} \approx \frac{C}{2}$, computation of the correlation requires $(C_{\mathrm{NZ}} - 1) + (N_{\mathrm{G}} - 1)$ K-bit additions per sample. The memory requirement of the correlation is $N_{\mathrm{G}} M C$ K-bit samples to store MC past received samples and $(N_{\mathrm{G}} - 1) MC$ *already computed* correlation values. Packet detection requires one additional K-bit comparison per sample with negligible memory requirements.

2) Timing Acquisition: Because it is performed once per packet for a short duration (once verification succeeds), its complexity and memory requirements are negligible.

B. Complexity and Memory Requirements for the PICNIC Algorithm

1) Packet Detection: Complexity and memory requirements of PICNIC for packet detection are similar to the conventional method. An additional comparison is required for every sample (see (6)). But the additional cost is offset because the PICNIC correlation can be partly implemented with counters, reducing both memory consumption and complexity.

2) Timing Acquisition: It is identical to the conventional method and assumed negligible.

3) Preamble Code Interference Cancellation: Only the first part of the IC algorithm runs during the entire packet detection and timing acquisition phase. The second part runs only when MUI is detected. All algorithms act on blocks of MC samples. The total memory requirement is $(M + 1)CK$ bits. The complexity of MUI detection is dominated by the K-bit comparisons needed to build $\mathbf{z}^{\mathrm{max}}$ and adds roughly one comparison per sample. Remaining operations are negligible. Complexity of the first-path selection is dominated by the JBSF procedure. In the worst case, it requires $\frac{M}{2}(C_{\mathrm{cross\text{-}peak}} + 1)$ comparisons. Memory requirements are negligible. Memory requirements of MUI cancellation are negligible. Complexity requirements depend on median computation complexity. See [38] and references therein.

C. Complexity and Memory Requirements for SFD Detection With DESSERT

SFD detection runs only once timing is acquired. DESSERT stores the LLRs of N_{sfd} blocks of MC samples. Consequently, the only operations per sample are 1) scaling the samples with $\frac{q_m}{N_0^2}$ (precomputed after channel estimation, requiring MK bits of memory), 2) table lookup of the logarithm of the confluent

hypergeometric limit function, and 3) adding the result to the stored LLR of the ith block. Then, LLR_{sfd} is evaluated $N_{\text{sfd}}+1$ times after each block. Each evaluation requires one comparison and N_{sfd} additions in the worst case (if $\mathbf{s}_{\mathbf{k}}^{2}$ differs from the SFD in every position). The threshold in Section IV-B adds one comparison for every sample. SFD detection with hard-decision decoding according to (19) has a similar complexity. Discussion of the complexity of offline algorithms is omitted. As stated earlier, the operation of an offline algorithm is similar to its online counterpart but over a much larger number of samples. It is safe to assume its complexity to be several times higher than an online version.

REFERENCES

[1] M. Flury, R. Merz, and J. Y. Le Boudec, "Robust non-coherent timing acquisition in IEEE 802.15.4a IR-UWB networks," *Proc. IEEE Int. Symp. Personal, Indoor, Mobile Radio Commun. (PIMRC)*, pp. 1642–1646, 2009.

[2] *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Std 802.15.4-2006 (Rev. of IEEE Std 802.15.4-2003), 2006.

[3] *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs): Amendment to add alternate PHY*, IEEE Std 802.15.4a-2007 (Amd. to IEEE Std 802.15.4-2006), 2007.

[4] J. Zhang, P. Orlik, Z. Sahinoglu, A. Molisch, and P. Kinney, "UWB systems for wireless sensor networks," *Proc. IEEE*, vol. 97, no. 2, pp. 313–331, Feb. 2009.

[5] M. Z. Win and R. A. Scholtz, "Impulse radio: How it works," *IEEE Commun. Lett.*, vol. 2, no. 2, pp. 36–38, 1998.

[6] V. Lottici, A. D'Andrea, and U. Mengali, "Channel estimation for ultra-wideband communications," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 9, pp. 1638–1645, 2002.

[7] W. M. Lovelace and J. K. Townsend, "The effects of timing jitter and tracking on the performance of impulse radio," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 9, pp. 1646–1651, 2002.

[8] S. Dubouloz, A. Rabbachin, S. de Rivaz, B. Denis, and L. Ouvry, "Performance analysis of low complexity solutions for UWB low data rate impulse radio," *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, p. 2440, May 2006.

[9] K. Witrisal, G. Leus, G. Janssen, M. Pausini, F. Troesch, T. Zasowski, and J. Romme, "Noncoherent ultra-wideband systems," *IEEE Signal Process. Mag.*, vol. 26, no. 4, pp. 48–66, Jul. 2009.

[10] R. Tesi, M. Hämäläinen, J. Iinatti, J. Oppermann, and V. Hovinen, "On the multi-user interference study for ultra wideband communication systems in AWGN and modified Saleh–Valenzuela channel," in *Proc. IEEE Joint Conf. Ultra Wideband Syst. Technol./Int. Workshop Ultra Wideband Syst.*, 2004, pp. 91–95.

[11] M. Flury, R. Merz, J.-Y. le Boudec, and J. Zory, "Performance evaluation of an IEEE 802.15.4a physical layer with energy detection and multi-user interference," *Proc. IEEE Int. Conf. Ultra-Wideband (ICUWB)*, pp. 663–668, 2007.

[12] M. Flury, R. Merz, and J.-Y. Le Boudec, "An energy detection receiver robust to multi-user interference for IEEE 802.15.4a networks," *Proc. IEEE Int. Conf. Ultra-Wideband (ICUWB)*, pp. 149–152, 2008.

[13] G. Durisi and G. Romano, "On the validity of Gaussian approximation to characterize the multiuser capacity of UWB TH PPM," *Proc. IEEE Conf. Ultra Wideband Sys. Technol. (UWBST) Dig. of Papers*, pp. 157–161, 2002.

[14] A. Forouzan, M. Nasiri-Kenari, and J. Salehi, "Performance analysis of time-hopping spread-spectrum multiple-access systems: Uncoded and coded schemes," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 671–681, 2002.

[15] Y. Dhibi and T. Kaiser, "On the impulsiveness of multiuser interferences in TH-PPM-UWB systems," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2853–2857, 2006.

[16] J. Ibrahim and R. Buehrer, "Two-stage acquisition for UWB in dense multipath," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, pp. 801–807, Apr. 2006.

[17] I. Guvenc, Z. Sahinoglu, P. Orlik, and H. Arslan, "Searchback algorithms for TOA estimation in non-coherent low-rate IR-UWB systems," *Wireless Personal Commun.*, vol. 48, no. 4, pp. 585–603, 2009.

[18] A. D'Amico, U. Mengali, and L. Taponecco, "Ranging algorithm for the IEEE 802.15.4a standard," in *Proc. IEEE Int. Conf. Ultra-Wideband (ICUWB)*, Sep. 2009, pp. 285–289.

[19] A. A. D'Amico, U. Mengali, and L. Taponecco, "TOA estimation with the IEEE 802.15.4a standard," *IEEE Trans. Wireless Commun.*, vol. 9, no. 7, pp. 2238–2247, 2010.

[20] F. C. Y.-S. Kwok and X. Peng, "Ranging mechanism, preamble generation, and performance with IEEE 802.15.4a low-rate low-power UWB systems," *Proc. IEEE 9th Int. Symp. Spread Spectrum Tech. Appl. (ISSSTA)*, pp. 430–434, 2006.

[21] T. Helleseth, "Some results about the cross-correlation function between two maximal linear sequences," *Discrete Math.*, vol. 16, no. 3, pp. 209–232, 1976.

[22] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE*, vol. 55, no. 4, pp. 523–531, 1967.

[23] A. A. D'Amico, U. Mengali, and E. Arias-De-Reyna, "Energy-detection UWB receivers with multiple energy measurements," *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2652–2659, 2007.

[24] M. Flury, "Interference robustness and security of impulse-radio ultra-wide band networks," Ph.D. thesis, EPFL, Lausanne, Switzerland, 2010.

[25] "IEEE 802.15.4a Channel Model—Final Report," 2004.

[26] K. Yu and I. Oppermann, "Timing acquisition for IR-UWB systems," in *Proc. 8th Int. Symp. Signal Process. Its Appl.*, 2005, vol. 1, pp. 287–290.

[27] A. Spaulding and D. Middleton, "Optimum reception in an impulsive interference environment-part I: Coherent detection," *IEEE Trans. Commun.*, vol. 25, no. 9, pp. 910–923, 1977.

[28] W. M. Lovelace and J. K. Townsend, "Threshold discrimination and blanking for large near-far power ratios in UWB networks," *IEEE Trans. Commun.*, vol. 53, no. 9, pp. 1447–1450, 2005.

[29] M. Flury and J.-Y. Le Boudec, "Interference mitigation by statistical interference modeling in an impulse radio UWB receiver," in *Proc. IEEE Int. Conf. Ultra-Wideband*, Sep. 2006, pp. 393–398.

[30] T. Erseghe, V. Cellini, and G. Dona, "On UWB impulse radio receivers derived by modeling MAI as a Gaussian mixture process," *IEEE Trans. Wireless Commun.*, vol. 7, no. 6, pp. 2388–2396, Jun. 2008.

[31] J. Mitra and L. Lampe, "Design and analysis of robust detectors for TH IR-UWB systems with multiuser interference," *IEEE Trans. Commun.*, vol. 57, no. 8, pp. 2210–2214, Aug. 2009.

[32] N. Beaulieu and D. Young, "Designing time-hopping ultrawide bandwidth receivers for multiuser interference environments," *Proc. IEEE*, vol. 97, no. 2, pp. 255–284, Feb. 2009.

[33] A. El Fawal and J.-Y. Le Boudec, "A robust signal detection method for ultra wide band (UWB) networks with uncontrolled interference," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 4, pp. 1769–1781, Jun. 2006.

[34] Z. Sahinoglu and I. Guvenc, "Multiuser interference mitigation in noncoherent UWB ranging via nonlinear filtering," *EURASIP J. Wireless Commun. Netw.*, pp. 1–10, 2006.

[35] D. Dardari, A. Giorgetti, and M. Z. Win, "Time-of-arrival estimation of UWB signals in the presence of narrowband and wideband interference," *Proc. IEEE Int. Conf. Ultra-Wideband (ICUWB)*, pp. 71–76, 2007.

[36] Z. Sahinoglu and S. Gezici, "Ranging in the IEEE 802.15.4a standard," in *Proc. IEEE Annu. Wireless Microw. Technol. Conf. (WAMICON)*, 2006, pp. 1–5.

[37] D. Dardari, A. Conti, U. Ferner, A. Giorgetti, and M. Win, "Ranging with ultrawide bandwidth signals in multipath environments," *Proc. IEEE*, vol. 97, no. 2, pp. 404–426, Feb. 2009.

[38] C. Lee and C.-W. Jen, "Bit-sliced median filter design based on majority gate," *Proc. Inst. Electr. Eng.—Circuits, Devices, Syst.*, vol. 139, no. 1, pp. 63–71, Feb. 1992.

Manuel Flury received the Master's of Science degree in communication systems engineering from the Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, in 2005.

During his studies, he completed internships at Nokia Research Center, Helsinki, Finland, and at Qualcomm Inc., San Diego, CA. In 2005, he joined the Laboratory for Computer Communication and Applications (LCA) at EPFL, School of Computer and Communication Sciences, and began working on his Ph.D. dissertation. There, he participated to the National Center of Competence in Research on Mobile Information and Communication Systems (NCCR-MICS) and received the Ph.D. degree in 2010. His research interests are in wireless communication and computer networks.

Ruben Merz (S'05–M'08) received the M.Sc. and Ph.D. degrees in communication systems from Ecole Polytechnique Fédérale de Lausanne (EPFL), School of Computer and Communication Systems, Lausanne, Switzerland, in 2003 and 2008, respectively.

He is a Senior Research Scientist at Deutsche Telekom Laboratories, Berlin, Germany. His research interests focus on the design, modeling, and performance evaluation of wireless communication systems and networks.

Dr. Merz is a recipient of a Best Student Paper award at the 2005 IEEE International Conference on Ultra-Wideband (ICU). He is a member of the ACM.
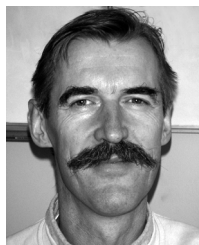
Jean-Yves Le Boudec (M'89–SM'01–F'04) graduated from Ecole Normale Superieure de Saint-Cloud, Paris, France, where he received the Agregation in mathematics in 1980 (rank 4) and the Ph.D. degree from the University of Rennes, France, in 1984.

From 1984 to 1987, he was with INSA/IRISA, Rennes, France. In 1987, he joined Bell Northern Research, Ottawa, Canada, as a Member of Scientific Staff in the Network and Product Traffic Design Department. In 1988, he joined the IBM Zurich Research Laboratory, where he was manager of the Customer Premises Network Department. In 1994, he joined Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, as an Associate Professor. He is currently a Full Professor at EPFL. His interests are in the performance and architecture of communication systems. In 1984, he developed analytical models of multiprocessor, multiple bus computers. In 1990, he invented the concept called "MAC emulation," which later became the ATM forum LAN emulation project and developed the first ATM control point based on OSPF. He also launched public domain software for the interworking of ATM and TCP/IP under Linux. In 1998, he proposed the first solution to the failure propagation that arises from common infrastructures in the Internet. He contributed to network calculus a recent set of developments that forms a foundation to many traffic control concepts in the Internet, and co-authored a book on this topic. He is also the author of the book *Performance Evaluation* (EPFL Press, 2010).

Dr. Le Boudec received the IEEE Millenium Medal, the Infocom 2005 Best Paper Award, the CommSoc 2008 William R. Bennett Prize, and the 2009 ACM Sigmetrics Best Paper Award. He is or has been on the program committee or editorial board of many conferences and journals, including SIGCOMM, Sigmetrics, INFOCOM, Performance Evaluation, and the ACM/IEEE TRANSACTIONS ON NETWORKING.