A practical synchronization algorithm for IEEE 802.15.4a UWB receivers

Tan Nghia Duong, Minh Tu Hoang, Manh Hoang Tran and Quang Hieu Dang
Hanoi University of Science and Technology
Email: hieu.dangquang@hust.edu.vn

***Abstract*—This paper proposes a two-step synchronization algorithm for UWB receivers in accordance with IEEE 802.15.4a standard. A low data rate, low complexity receiver is developed using a typical energy detection UWB non-coherent analog front-end, which consists of a square-law device and an integrate-and-dump ADC with sub-Nyquist sampling rates. A coarse synchronization algorithm, often neglected in literature, is introduced by continuously correlating received samples' segments with a self-generated template until a simple threshold condition is satisfied. A proposed fine synchronization algorithm exploits the low sampling rates of the system and does not need to measure noise power, which helps reduce the receiver complexity (and power consumption). Finally, we assess, by simulation, the impact of different sampling rates and the synchronization lengths on bit-error-rate (BER) performance of the system.**

I. Introduction

It has been several years since ultra-wideband (UWB) radio was approved for commercial use by FCC [1] in 2001, and since the first UWB standard, IEEE 802.15.4a [2], was introduced in 2007, and almost no UWB chip is successfully entered in the market so far, to which the transceiver's cost is one of the main reasons. Non-coherent receivers have been potential candidates for low data-rate applications [3], [4], because Nyquist sampling rate (could be at Gsamples/s) is not required and channel estimation, which may consumes intensive signal processing power due to densely multipath channels, is not needed either.

There are two popular classes of non-coherent receivers: transmitted reference (TR) [5] and energy detection (ED) [6]. TR receiver uses a reference pulse (spread by the channel) as a noisy template to detect the data-bearing pulse. This causes at least 3dB loss in performance and a delay line longer than channel delay spread is difficult to implement at sub-nanosecond precision [7]. Meanwhile, ED receiver computes the signal energy, and detects the data symbol based on the energy presence of pulses in certain positions within a symbol period. Despite its inferior noise performance (because noise is also squared), energy detection is the most feasible receiver structure as proposed in IEEE 802.15.4a standard [2]. This paper, therefore, uses ED receiver.

While data detection is pretty straightforward in ED receiver, the synchronization algorithms are often complex and suffer from the fact that UWB multipath channels are dense, in which the first LOS path may not always be the strongest path [8], [9].

A threshold-based synchronization algorithm was proposed in [10] using maximum energy search and a backward search. However, the threshold depends on SNR, which is not always easily measured instantly at receiver. In [11], a synchronization algorithm was proposed to estimate the position of the ranging marker, e.g. the first pulse of the first symbol in the physical-layer header in IEEE 802.15.4a standard, by performing a two-dimension search for the highest peak and a backward search for the leading path. One important assumption of this algorithm is that the noise power is known, which is not always the case in practical situation, and the noise measurement can increase the receiver complexity greatly. Another assumption is that the received signal segment always starts in SYNC part, while in fact, it can be anywhere, e.g. in data part.

In our proposed paper, aiming at low data rate and low complexity receivers, both assumptions are relaxed, i.e. received signal can start anywhere, and noise measurement is not needed. As a result, two-step algorithm is implemented: the coarse synchronization algorithm estimates the position of the SYNC part within received signal segment, and the fine synchronization algorithm estimates the position of the ranging marker. As discussed later, our algorithm is not only more practical but also superior in certain scenarios.

II. Signal model and receiver structure

As specified in IEEE 802.15.4a standard, signal is transmitted in frames, each of which consists of a synchronization header (SHR) preamble, a physical-layer header (PHR) and a data field (PSDU). The SHR preamble is then divided into a synchronization preamble (SYNC) and a start of frame delimiter (SFD), as follows

$$s(t) = \sum_{t=0}^{N_{shr}-1} a_i \psi(t - iT_{psym})$$

where N_{shr} is the total number of preamble symbols in SHR; T_{psym} is the preamble symbol duration; the symbols $a_i = 1$ in the SYNC part and $a_i \in \{-1, 0, +1\}$ in the SFD part; and $\psi(t)$ is defined as

$$\psi(t) \triangleq \sum_{k=0}^{K_{pbs}-1} c_k p(t - kT_{pr})$$

where $\{c_k\}_{k=0}^{K_{pbs}-1}$ is a *perfectly balanced sequence*, $c_k \in \{-1, 0, +1\}$; $p(t)$ is the UWB monocycle and $T_{pr} \triangleq T_{psym}/K_{pbs}$ is the pulse repetition period.

The PHR and PSDU are modulated using BPM-BPSK, in which the number of bursts per data symbol is $N_{burst} = 32$ as provided by the standard [2].

978-1-4799-1089-2/13/$31.00 ©2013 IEEE

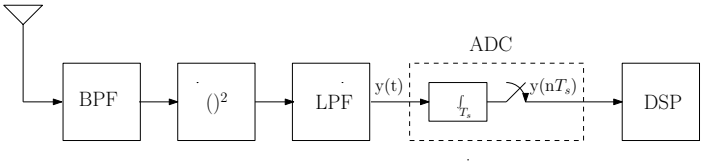

Fig. 1. Block diagram of the ED receiver

The structure of the non-coherent ED receiver is shown in Fig. 1. It can be easily proved that, given a transmitted pulse $p(t)$, after upconverting to center frequency f_0 (specified by the standard), spread by the multipath channel $h(t)$, the output of the LPF (after passing through BPF and a square-law device) at receiver (ignoring noise) is

$$q(t) = [p(t) * h(t)]^2$$

Here, $q(t)$ can be regarded as a non-negative pulse of duration $(T_p + T_h)$, where T_p is the pulse duration and T_h is the channel length. Here we assume $T_{pr} > T_p + T_h$ so that there is no inter-chip interference.

Therefore, during the SHR interval, the ADC's input is

$$y(t) = \sum_{i=0}^{N_{shr}-1} a_i \sum_{k=0}^{K_{pbs}-1} c_k^2 q(t - kT_{pr} - iT_{psym} - \tau_0) + n_y(t) \quad (1)$$

where $n_y(t)$ represents thermal noise after passing the LPF.

Finally, at ADC, signal $y(t)$ is sampled using *integrate and dump* with sampling period T_s. The resulting samples $y(nT_s)$ will be used by our following synchronization algorithms.

III. Synchronization algorithms

Assuming the transmitter starts sending signal at time instant 0, and the receiver is turned on randomly, and capture signal from time instant t_0 as illustrated in Fig. 2. The objective of the synchronization operation is, given t_0 and the signal structure of SHR (SYNC + SFD), estimate the time of arrival t_{PHR} of the first non-negative pulse $q(t)$ in the first PHR symbol (the ranging marker).

Since t_0 may be anywhere in the signal frame: SHR preamble, PHR or PSDU data field, to determine the ranging marker, we propose a two-step synchronization algorithm:

- Step 1 - Coarse synchronization: Find the location of SYNC in the received signal, i.e. estimate t_1.
- Step 2 - Fine synchronization: Remove SYNC and SFD symbols to obtain the first symbol in the PHR field, i.e. estimate t_{PHR}.

A. Coarse synchronization

The coarse synchronization algorithm aims at, given t_0, estimating the position of the SYNC field within the received signal. As in [2], SYNC consists of N_{sync} repetitions of the preamble symbol constructed from the sequence $\{c_k\}$. In this paper, we utilize the structure of SYNC to develop a SYNC detection algorithm.

First, we divide the received samples $y(nT_s)$ into segments of length L_s.

$$\mathbf{y}_i = [y_{i0}, y_{i1}, \ldots, y_{i(L_s-1)}]$$

A template $\mathbf{s}_c$ is created by repeating each element of the perfectly balanced sequence L_s/K_{pbs} times (Fig. 3), where $L_s = T_{psym}/T_s$ is the number of samples in a SYNC symbol.

Each segment $\mathbf{y}_i$ is consecutively correlated with the template $\mathbf{s}_c$ as follows (Fig. 4)

$$g_{ik} = \sum_{j=kN+1}^{(k+1)N} \mathbf{s}_c(j) \cdot y_{ij}, \quad k = 0, 1, \ldots, (Q-1)$$

where $N = \lceil L_s/P \rceil$ is the integration length, P is an integer varying from 1 to K_{pbs}.

Collecting all g_{ik}, we obtain a Q-element array $\mathbf{g}_i = [g_{i0}, g_{i1}, \ldots, g_{i(Q-1)}]$ with $Q = \lceil K_{pbs}/P \rceil$.

It is observed that, when the sequence $\{c_k\}_{k=0}^{K_{pbs}-1}$ is perfectly balanced, we can always choose a value of P so that there's only one maximum element in each array $\mathbf{g}_i$. With this value of P, we look for the position of the maximum element $g_{i_{max}}$ in each array $\mathbf{g}_i$. As SYNC includes the repeated preamble symbols, if there are K ($K \leq N_{sync}$) consecutive maximum elements which have the same position, it is possible to claim that these K segments are all in SYNC. The beginning of the $(\lceil K/2 \rceil)$-th segment is now considered as t_1.

When implementing this algorithm in hardware, the processes of correlation and finding maximum elements can be performed instantly on every input sample $y(nT_s)$. Moreover, both coarse and fine synchronization algorithms, which will be discussed in the next section, can also be performed simultaneously. Consequently, the receiver does not require extra buffers to save the samples $y(nT_s)$, which helps reduce the resource usage and complexity while increase the processing speed of the receiver.

B. Fine synchronization

After successfully locating SYNC in the received signal, the objective now is to determine the ranging marker t_{phr} (Fig. 2). Suppose t_1 is in the symbol period $(N_\tau - 1)$ of SYNC, our fine synchronization algorithm is implemented in the following order:

1) Estimate the interval $\tau \in [0, T_{psym})$ between t_1 and the beginning of symbol N_τ,
2) Remove all the preamble symbols from $t_1 + \tau$ to t_{phr} to obtain the first symbol of the PHR.

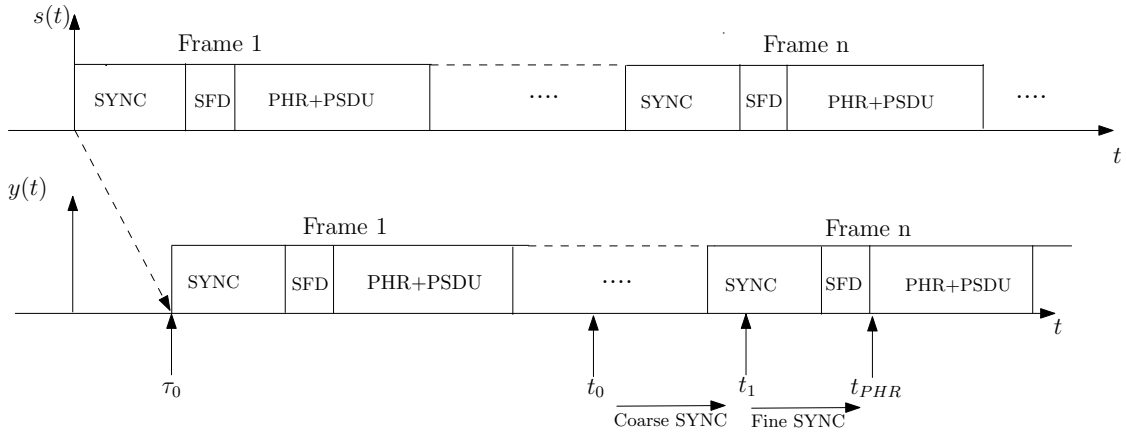

Fig. 2. Format of the IEEE 802.15.4a signal frames.

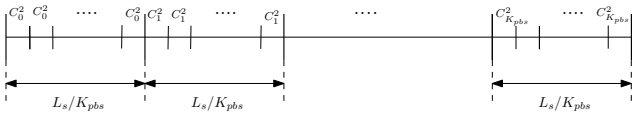

Fig. 3. Structure of the template $\mathbf{s}_c$.

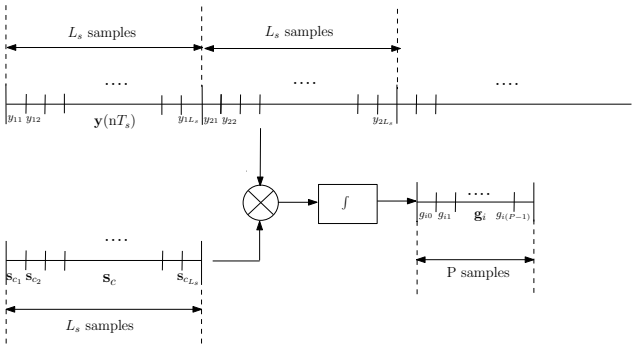

Fig. 4. Correlate the template $\mathbf{s}_c$ with the samples of signal $y(t)$

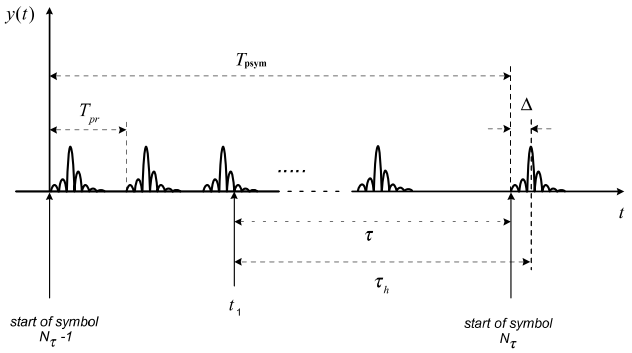

Fig. 5. The shape of a T_{psym}-long segment of $y(t)$ in the absence of noise

1) Estimation of τ: As we can see from Fig. 5, τ is calculated as $\tau = \tau_h - \Delta$.

Firstly, we measure the interval τ_h from t_1 to the highest peak of the first pulse $q(t)$ in symbol $(N_\tau - 1)$. Assuming T_s a sub-multiple of T_{pr}, say $T_{pr}/T_s = N_s$, and write $\tau_h = mT_{pr} + \varepsilon$ with $0 \le m \le K_{pbs}$ and $0 \le \varepsilon < T_{pr}$. According to [11], the values of m and ε are computed by considering the sum

$$S[\tilde{m}, \tilde{n}_\varepsilon] \triangleq \frac{1}{M} \sum_{i=0}^{M-1} \sum_{k=0}^{K_{pbs}-1} c^2_{|k-\tilde{m}|_{K_{pbs}}} y(t_{\tilde{n}_\varepsilon + (k+iK_{pbs})N_s}) \tag{2}$$

where $t_l \triangleq t_1 + lT_s$ $(l = 0, 1, 2, ...)$; $|a|_b \triangleq a$ modulo b and $n_\varepsilon = \varepsilon/T_s$. Here M is chosen so that $S[\tilde{m}, \tilde{n}_\varepsilon]$ only includes the SYNC symbols, and m and ε can be obtained as

$$\begin{aligned} \hat{m}, \hat{n}_\epsilon = \quad & arg\,max \quad \{S[\tilde{m}, \tilde{n}_\epsilon]\} \\ & 0 \le \tilde{m} \le K_{pbs} \\ & 0 \le \tilde{n}_\epsilon \le N_s - 1 \end{aligned} \tag{3}$$

Secondly, the difference Δ between the leading path and the highest peak of the first $q(t)$-pulse in symbol N_τ is measured by monitoring the threshold crossing of the sum

$$S'[m, \tilde{n}_\varepsilon] \triangleq \frac{1}{M} \sum_{i=0}^{M-1} \sum_{k \in \Gamma(m)} c^2_{|k-m|_{K_{pbs}}} y(t_{\tilde{n}_\varepsilon + (k+iK_{pbs})N_s}) \tag{4}$$

where $\Gamma(m)$ is the set of indices k so that $c^2_{|k-m|_{K_{pbs}}} = 1$ and $c^2_{|k-m-1|_{K_{pbs}}} = 0$. If a crossing occurs, it indicates that the leading path of the $q(t)$-pulse has been reached.

In [11], threshold λ is computed as a function of the noise level, which requires a noise power measurement in receiver. With this calculation of λ, the estimation of τ gives us a precise result even when the sampling period is very small (T_s varies from 1ns to 4ns in [11]). The accuracy of this algorithm is about one sampling period, hence the ranging error are on the order of 20cm. However, the use of this threshold λ increases not only the complexity computation but also the time to process signal of the receiver.

As mentioned in Section I, our purpose is to design a low-data-rate and low-power-consumption UWB IEEE 802.15.4a system. To successfully decode the transmitted data symbols

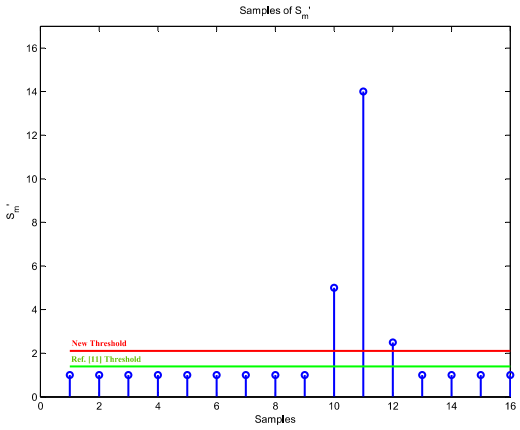

Fig. 6. Comparison between threshold λ' with $S'[m, \tilde{n}_\varepsilon]$ in case $T_s = 16$ [ns]

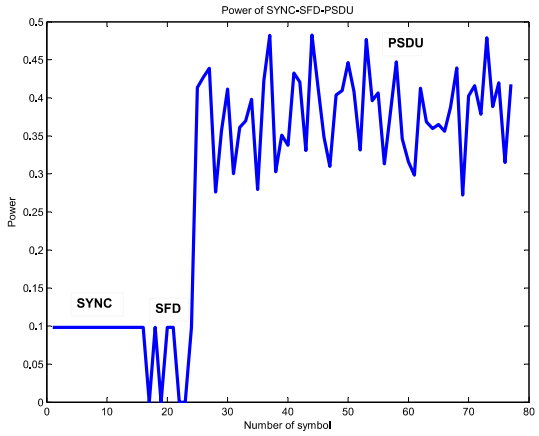

Fig. 7. Power comparison among three parts SYNC, SFD and PHR/PSDU in the absence of noise

in this system, the synchronization process only needs to locate t_{phr} with the accuracy of $1/2$ burst duration in a data symbol. Therefore, it is possible to replace the GHz sampling rate ADC in [11] with a lower and cheaper rate one without compromising the data decoding performance.

With a lower sampling rate, the difference between the amplitudes of the samples of $S'[m, \tilde{n}_\varepsilon]$ in the noise region and the $q(t)$-pulse becomes rather large. This is because of the fact that all the samples of the $q(t)$-pulse are non-negative, which will accumulate more if the integration length increases.

Fig. 6 illustrates the samples of $S'[m, \tilde{n}_\varepsilon]$ in case $T_s =$ 16ns.

The threshold λ computed from the noise level is now unnecessary. From the above observation, we propose a new threshold λ' without the knowledge of noise. Specifically, λ' is simply computed as an average sum of $S'[m, \tilde{n}_\varepsilon]$.

$$\lambda' = \sum_{\tilde{n}_\varepsilon=0}^{N_s-1} S'[m, \tilde{n}_\varepsilon]$$

As can be seen in Fig. 6, in case $T_s = 16$ns the new threshold λ' is approximately equal to threshold λ in [11], so it can be used to differentiate between the noise region and the $q(t)-$pulse in $S'[m, \tilde{n}_\varepsilon]$. When T_s is smaller, the amplitudes of the samples corresponding to the $q(t)$-pulse is not much greater than the noise region. Therefore, the use of the average of $S'[m, \tilde{n}_\varepsilon]$ as a threshold is not effective to determine the value of Δ. The performance of our algorithm for various sampling periods will be shown in Section IV.

Finally, after measuring the values of τ_h and Δ, we have $\tau = \tau_h - \Delta$, or equivalently, the start of each individual preamble symbol now could be identified.

2) PHR detection: The final objective now is to locate the ranging marker t_{phr}, i.e. the start of the first PHR symbol. Fig. 7 shows the power comparison among three parts of a UWB IEEE 802.15.4a frame with the parameters: $K_{pbs} = 31$, $N_{burst} = 32$ and $N_{cpb} = 16$ (also be used in Section IV).

It can be seen that the average power during the data field is always much greater than the SHR. Specifically, the average power of each SYNC symbol (ignoring noise) is

$$P_{sync} = \frac{K_{-1,+1} P_q}{K_{pbs}} \tag{5}$$

where $K_{-1,+1}$ is the number of non-zero elements in sequence $\{c_k\}$ and P_q is the power of a $q(t)$-pulse. Similarly, the average power of a T_{psym}-long segment in the PHR/PSDU data field is

$$P_{data} = \frac{N_{cpb} P_q}{N_{burst}} . \frac{T_{psym}}{T_{dsym}} \tag{6}$$

where N_{cpb} and T_{dsym} are the number of chips per burst and the duration of a PHR/PSDU data symbol, respectively. From (5) and (6), we have

$$\frac{P_{data}}{P_{sync}} = \frac{K_{pbs}}{K_{-1,+1}} . \frac{N_{cpb}}{N_{burst}} . \frac{T_{psym}}{T_{dsym}} \tag{7}$$

Using the parameters mentioned above, the ratio in (7) is always approximately 3.75, which agrees with Fig. 7. Based on this observation, the first PHR symbol in the received signal can be detected by:

- Setting the threshold $\alpha = f * P_{sync}$ where P_{sync} is the average power of symbol N_τ and $1 < f < 3.75$ is called the comparing factor,
- Comparing the average power each T_{psym}-long segment one after another starting from $t_1 + \tau + T_{psym}$ with threshold α; when the threshold crossing occurs, the start of the being considered segment is just the ranging marker t_{phr}.

IV. Simulation Results

We simulate a baseband UWB IEEE 802.15.4a system to verify the proposed synchronization algorithm and compare its performance with the one developed in [11] which measures noise power under various conditions. The UWB pulse is the second derivative of a Gaussian monocycle of duration 2 [ns] (the effect of the transmitted and received antennas

TABLE I. SIMULATION PARAMETERS

Parameters	K_{pbs}	L	N_{sync}	N_{sfd}	N_{burst}	N_{cpb}
Values	31	64	16	8	32	16

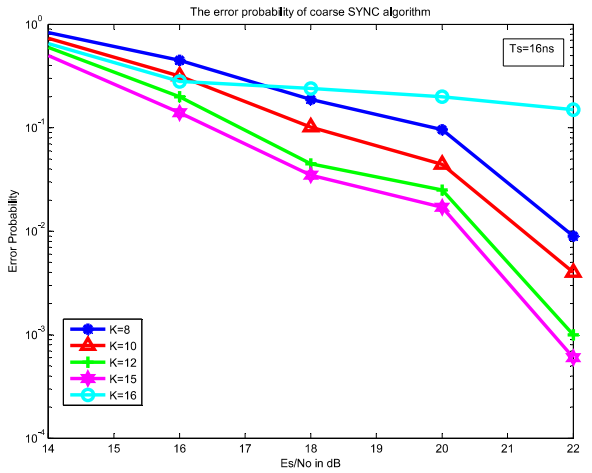

Fig. 8. The error probabilities of the coarse synchronization algorithm for varying K

on the signal is approximated as differentiation). Due to limited number of pages, we only use channel model CM1 as provided in IEEE 802.15.4a standard. 5000 Monte Carlo runs are executed to obtain the plots of error probability vs E_s/N_0 for all the algorithm presented in this paper. Here, E_s/N_0 is defined as the energy of a $q(t)$-pulse over noise power spectral density, and error probability is defined as the ratio of the number of unsuccessful synchronizations (when the timing error is so large that the data decoding gives completely incorrect results) to the total number of Monte Carlo runs.

The chosen parameters of a IEEE 802.15.4a signal frame are listed in Table I. To satisfy the condition $T_{pr} > T_p + T_h$ assumed in Section II, the perfectly balanced sequence must have 31 elements and the length of delta function must be $L = 64$. According to [2], the chip duration T_c is approximately 2ns, so the pulse repetition is $T_{pr} = LT_c \approx 128$ns, which largely exceeds the delay spread of channel CM1 in [9]. Additionally, in our simulation N_{cpb} is set to 16, so the data symbol duration is $T_{dsym} = N_{burst}N_{cpb}T_c \approx 1025.64$ns, which is much shorter than the preamble symbol duration $T_{psym} = K_{pbs}T_{pr} \approx 3974.36$ns.

A. Coarse synchronization

Fig. 8 shows the error probability of the coarse synchronization process. We perform the simulation with various values of K to verify the accuracy of the algorithm. The conclusion is that the bigger K is, the more accurate the algorithm achieves. However, if the maximum possible value of K, i.e. $K = N_{sync}$, is chosen, the performance of the algorithm breaks down. The reason is that, in this case, mostly the first and the last segment will include the samples of PSDU and SFD, so we rarely have the K consecutive similar maximum positions of $\mathbf{g}_i$. Therefore, the optimal value of K is $N_{sync} - 1$.

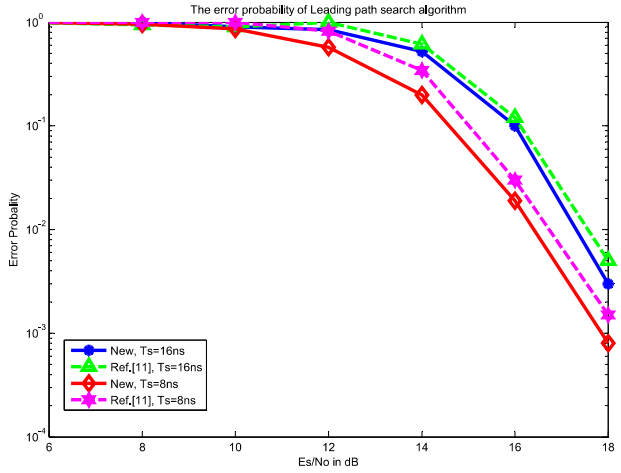

Fig. 9. Error probability of the estimation of τ for $T_s \in \{16, 8\}$ [ns]

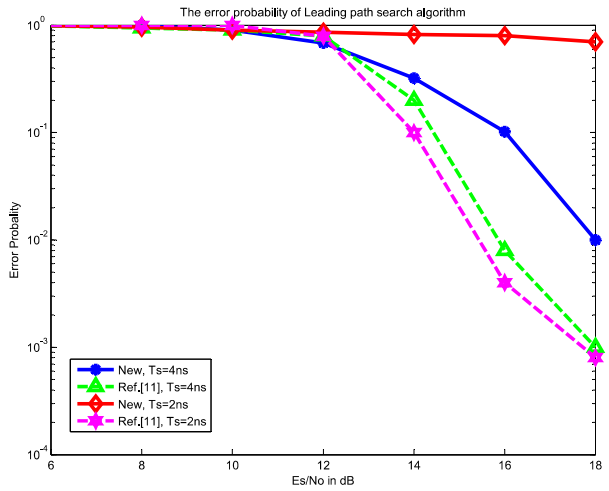

Fig. 10. Error probability of the estimation of τ for $T_s \in \{2, 4\}$ [ns]

B. Fine synchronization

Fig. 9 and 10 illustrate the comparison between our new threshold λ' and threshold λ used in [11]. Since the previous coarse synchronization has located SYNC at the beginning of $(\lfloor K/2 \rfloor)$-th segment with $K = N_{sync} - 1$, M is configured as $N_{sync}/2$ to guarantee that all M symbols contributing to the sums in (2) and (4) belongs to SYNC.

It can be seen that our proposed algorithm is better when the sampling period T_s increases. When $T_s = 4$ [ns], our algorithm is worse. Furthermore, when $T_s = 2$ [ns], our algorithm does not work at all. Nevertheless, when increasing T_s to 8ns and 16ns, the use of threshold λ' achieves better performance than λ. Therefore when implemented on FPGA, instead of using the GHz sampling rate ADC, the receiver now can use the one which has a lower sampling rate, for example, about $50 - 100$MHz.

When a MHz sampling rate ADC is utilized, the receiver is certainly not able to locate the ranging marker with accuracy of about $1-2$ns as described in [11]. The synchronization error is always approximately a sampling period T_s. However, for the purpose of implementing a low-data-rate UWB system, this disadvantage is trivial because the receiver is still able

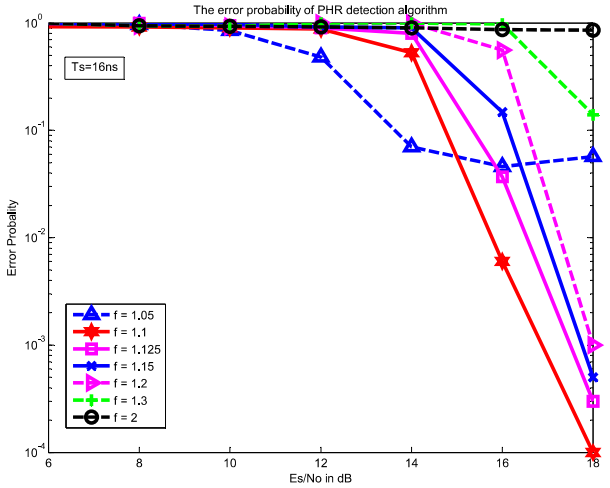

Fig. 11. The error probability of PHR detection algorithm for various values of f when $T_s = 16$ns.

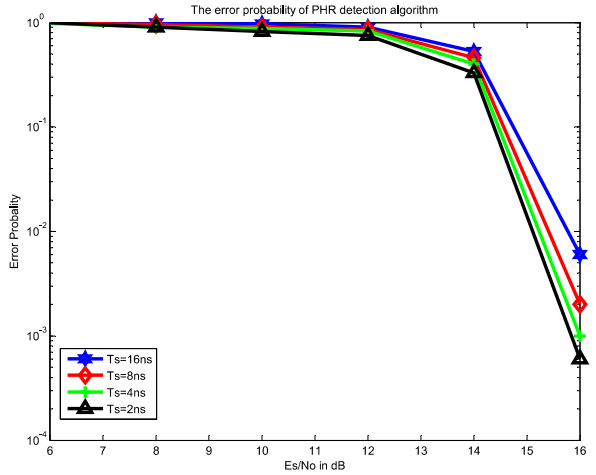

Fig. 12. The error probability of PHR detection algorithm for various values of T_s when $f = 1.1$

to successfully decode the transmitted signal even when the synchronization error is about $1/2$ burst duration. Therefore, the synchronization error here is allowed to be less than $1/2$ burst duration. In the simulation, N_{cpb} is set to 16, so the burst duration is $T_{burst} = N_{cpbs}T_c \approx 32$ns and the acceptable synchronization error is approximately 16ns, i.e., just a sampling period T_s.

Lastly, Fig. 11 and 12 indicates the error probability of the PHR detection algorithm for various values of comparing factor f and sampling period T_s, respectively. In Fig. 11, we set $T_s = 16$ns to compare among the different values of f. As can be easily seen, the optimal value of f is in the range from 1.1 to 1.125. With $f < 1.1$ or $f > 1.3$, the error probability is enormous even when E_s/N_0 is high. So we choose comparing factor $f = 1.1$ for the following simulations.

As shown in Fig. 12, with $f = 1.1$ the performances of the PHR detection algorithm for different sampling rates are nearly similar. Therefore, along with the estimation of τ algorithm using threshold λ' described above, an expensive high-sampling-rate ADC is not needed for our system.

V. Conclusions

In this paper, we have derived in detailed a novel two-step synchronization algorithm. The first step, coarse synchronization, guarantees that the receiver is working under any segment of received data. The second step, fine synchronization, offers a practical way to correctly find and detect the data symbols without needing to measure the noise power. Moreover, as shown in simulation, our proposed algorithm even achieves superior performances for low-sampling-rate cases, which hints that it is more suitable for low power, low data rate applications as one of targeted applications of the IEEE 802.15.4a standard. Therefore, FPGA implementation can be developed straightforwardly.

References

[1] "FCC notice of proposed rule making, revision of part 15 of the commission's rules regarding ultra-wideband transmission systems," Federal Communications Commission, Washington, D.C., Tech. Rep. ET-Docket 98-153, Apr. 2002.

[2] *IEEE standard 802.15.4a-2007, Specific Requirement Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*, IEEE Standard 802.15.4a-2007 Task Group Std., 2007.

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

[4] L. Yang, G. B. Giannakis, and A. Swami, "Noncoherent ultra-wideband (de)modulation," *IEEE Transactions on Communications*, vol. 55, pp. 810–819, Apr. 2007.

[5] R. Hoctor and H. Tomlinson, "Delay-hopped transmitted-reference RF communications," in *IEEE Conference on Ultra Wideband Systems and Technologies*, 2002, pp. 265–270.

[6] A. Amico, U. Mengali, and E. A. de Reyna, "Energy-detection UWB receivers with multiple energy measurements," *IEEE Transactions on Wireless Communications*, vol. 6, no. 7, Jul. 2007.

[7] K. Witrisal and J. Romme, "Oversampled weighted autocorrelation receivers for transmitted-reference UWB systems," in *Proc. IEEE Vehicular Technology Conference, VTC Spring*, Stockholm, Sweden, 2005.

[8] J. Foerster, "Channel modeling sub-committee report final," IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs), Tech. Rep., 2003.

[9] A. Molisch, J. Foerster, and M. Pendergrass, "Channel models for ultrawideband Personal Area Networks," *IEEE Personal Communications Magazine*, vol. 10, pp. 14–21, Dec. 2003.

[10] I. Guvenc and Z. Sahinoglu, "Threshold-based TOA estimation for impulse radio UWB systems," in *Proc. IEEE International Conference on UWB*, Zurich, Switzerland, Sep. 2005.

[11] A. Amico, U. Mengali, and L. Taponecco, "TOA estimation with the IEEE 802.15.4a standard," *IEEE Transactions on Wireless Communications*, vol. 9, no. 7, Jul. 2010.