DOI: 10.17725/rensit.2020.12.275

Synchronization Systems Modeling for IEEE 802.11ah Receiver in MATLAB

Fedor B. Serkin, Anton Yu. Dubrovko
Moscow Aviation Institute, https://mai.ru/
Moscow 125993, Russian Federation
E-mail: serkinfb@list.ru, dubrovkoay@mai.ru
Received December 6, 2019, after finalization on July 01, 2020, adopted on July 06, 2020

Abstract. **The problem of applicability of signal processing algorithms that was used in previous Wi-Fi standards to IEEE 802.11ah is discussed. The goal of the work was to study performance of the synchronization algorithms on a low signal to noise ratios. A computer simulation was carried out, that models packet reception in a channel with white noise. Failure probabilities of timing and frequency synchronization systems was measured, as well as bit-error rate. In a model for frequency and timing systems in coarse estimation was used autocorrelation method, in fine estimation – cross-correlation method; for equalizer least-square method was applied; for tracking system two method were explored: classical pilot based and time-frequency decision feedback loop (TF-DFL). As a result, it was confirmed, that TF-DFL method show better results, than classical pilot based one even for traveling pilots. Moreover, in order to approach the theoretical dependence of bit error rate for 10-th modulation coding scheme (MCS10), it is necessary to improve reviewed frequency synchronization and fine timing systems, as well as performance of equalization and tracking methods.**

Keywords: **IEEE 802.11ah, Wi-Fi HaLow, PHY, MCS10, Low SNR, Synchronization, Failure Probability, RFO Tracking, CFO Tracking, Bit Error Rate**

PACS: 84.40.Ua

For citation: Fedor B. Serkin, Anton Yu. Dubrovko. Synchronization Systems Modeling for IEEE 802.11ah Receiver in MATLAB. *RENSIT,* 2020, 12(2):275-286. DOI: 10.17725/rensit.2020.12.275.

Contents

1. INTRODUCTION

The IEEE 802.11ah standard was developed by Institute of Electrical and Electronics Engineers to support various Internet of Things (IoT) and Machine-to-Machine (M2M) applications. Its main features are low energy consumption, number of stations (STA) connected to a single access point (AP) is up to 8192 and radio connection distance up to 1 km [1]. This connection distance is possible because of sub-1GHz band operation and usage of Modulation Coding Scheme 10 (MCS10). This scheme arouse interest since it allows synchronization systems operation

with negative values of signal-to-noise ratio (SNR).

For now, there is one publication about synchronization systems for IEEE 802.11ah receiver [2]. In this paper algorithms of coarse and fine synchronization as well as residual frequency and phase offset (RFO) compensation are outlined. There is also improved channel estimation method is presented with SIG field usage. Preamble synchronization algorithms which were used in the paper is similar to algorithms for previous Wi-Fi standards (a/g/n/ac) [3], after all, ah is very close to them. At the same time, we did not reproduce productivity of RFO compensation system for MCS0, and MCS10 there was not examined at all. That is why systems from articles [4, 5] were chosen.

In the first case [4] was proposed blind estimation method for schemes with modulations from QPSK and higher. The method uses time-frequency decision feedback loop (TF-DFL). It gives quite good performance in SNR range from 4 dB and higher since relation of bit error rate (BER) vs SNR loses around 0.15 dB to theoretical limit. In the same article authors show relation of BER for the most common method which requires usage of pilot sequences. In this case lose to theory is about 1 dB, but in the same time this approach gives better performance with SNR bellow 4 dB. A description is given in the paper [5].

In our work synchronization algorithms performance for IEEE 802.11ah receiver with additive white gaussian noise (AWGN) channel with low values of SNR is considered that were used in the previous Wi-Fi standards.

2. MATERIALS AND METHODS

2.1. Standard

There are many various applications provided by IEEE 802.11ah [6] from smart grids, security systems and targeted advertising to surveillance systems, increasing connection distance to existing hotspots and outdoor Wi-Fi to offload traffic in cell networks. Some applications require coverage of large territories by means of single AP.

Typical IEEE 802.11ah network architecture is shown at **Fig. 1** [1]. This architecture is a centralized network that contains root AP, Relays and STAs. The Relay consist of Relay STA, relay function and Relay AP. At Fig. 1 Relay STAs of Relay 1 and 2 are connected to root AP. Relay STA of Relay 3 is connected to Relay AP of Relay 1. STA 1, which is not AP, is connected to Relay AP of Relay 1. Similarly, STAs 2 and 3 are connected to Relay AP of Relay 3, as well as STAs 4 and 5 are connected to Relay AP of Relay 2. To transmit a frame, for example, from STA 1 to Root AP, it should use relay function from Relay AP to Relay STA of Relay 1. The same but inverse way is passed by frames from Root AP to STA 1.

To reduce latencies in this architecture Transmission Opportunity (TXOP) mechanism is provided. Meanwhile, mandatory access method is Enhanced Distribution Channel Access (EDCA). Besides, there are two optional access

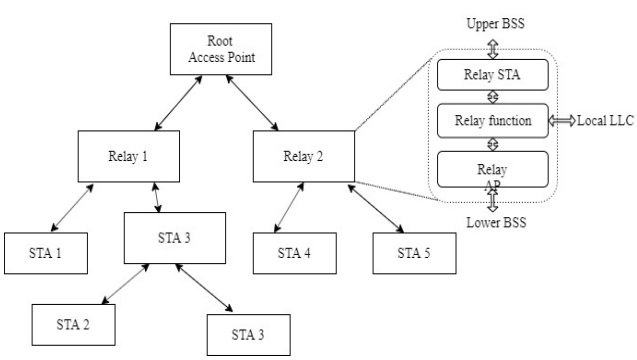

Fig. 1. *Relay network architecture in IEEE 802.11ah.*

methods: Restricted Access Window (RAW), Target Wake Time (TWT). The first one allows to reduce channel load by dividing devices into groups. Within this group there is distribution limited time windows for packet transmission. The second method allows to control packet transmission time from each STA. More information about these methods can be found in standard [1] or in papers [7,8].

On PHY standard provides data transmission with rate from 150 kbit/s over a distance of up to 1 km. This result is achieved by usage of unlicensed sub-GHz band and MCS10. The key parameters of MCS10 and the closest to it schemes that were used in our work are shown in **Table 1.** Papers [8, 9] and document [11] provide a detailed calculation of the radio link budget for our case. We are also interested in what kind of losses on implementation and fading are allowed. According to [12], in previous Wi-Fi standards, these losses accounted for about 12 dB. These losses are calculated using the following formula:

$M[\text{dB}] = RecSens - 30 - P_{MIN}$,

where $RecSens$ [dBm] – minimum receive

Table 1

IEEE 802.11ah parameters of MCS10/0/1

Parameter	Designation	MCS10	MCS0	MCS1
Receiver sensitivity	RecSens [dBm]	-98	-95	-92
Sampling frequency	$1/T_s$ [MHz]	1		
Sampling period	T_s [μs]	1		
FFT length	N_{fft}	32		
FFT period	T_{fft} [μs]	32		
Distance between subcarriers	$\Delta f = \frac{1}{N_{fft}T_s}$ [kHz]	31.25		
Number of subcarriers modulated	N_{dsc}	24		
Number of pilot subcarriers	N_p	2		
Cyclic prefix duration	T_{gi} [μs]	8		
OFDM symbol duration	T_F [μs]	40		
Relative code rate	R	1/2		
Code constraint length	k	7		
Modulation	-	BPSK		QPSK
Generating polynomial	-	[133 171]		

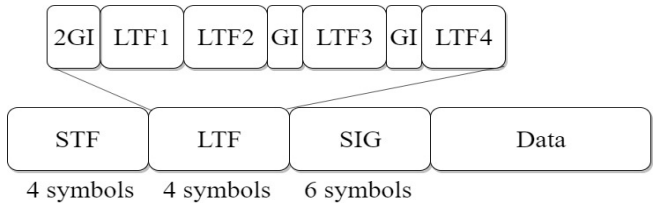

Fig. 2. *S1G_1M packet format.*

sensitivity at which packet error rate (PER) is equal to 10%, P_{MIN}[дБ] = $SNR + 10\lg kT_0W + NF$ receiver sensitivity, SNR – minimum theoretical signal-to-noise ratio at which PER = 10%, for example, for MCS10 it is closely equal to –3.23 dB, [J/K] – Boltzmann constant, T_0 = 293 [K] – the standard noise temperature, W [Hz] – bandwidth, a NF [дБ] – receiver noise figure.

When calculating the radio link budget, the ah working group set the noise coefficient value NF = 7 dB [11]. Fading and implementation loses for MCS10 are equal to the same 12 dB with the corresponding receiver sensitivity.

The main features of MCS10 are: x2 data repetition on half of the subcarriers, short training field (STF) 3 dB gain (**Fig. 2**). Due to repetition relation of BER vs SNR for MCS10 shifts to the negative side by 3 dB relative to MCS0, since the energy of the transmitted symbols is doubled. Thus, an operation area of MCS10 lies in the range of negative SNR. **Fig. 3** shows the shift of the

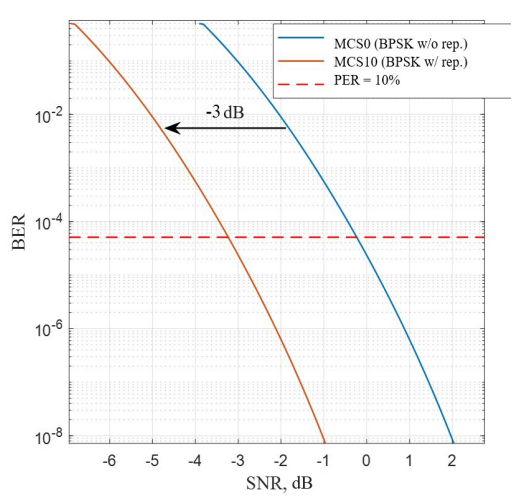

Fig. 3. *Repetition result in MCS10.*

BER curve in this case. Packets transmitted in MCS10 mode have S1G_1M format. Packet structure in this format is shown in Fig. 2.

STF is the short training field, which is used by an automatic gain control system and a coarse synchronization system. LTF is a long training field, which is used by fine synchronization system and equalizer. GI means guard interval. SIG is a signal field, which carries service information about the packet. Data is a data field, which contains a payload.

Data transmission on orthogonal subcarriers **Fig. 4** in the time domain can be described

$$g_i(t) = e^{\frac{j2\pi it}{T_{fft}}},$$

where T_{fft} – FFT duration, i – index of the corresponding subcarrier.

Orthogonality means that the following condition is met

$$\int_0^{T_{fft}} g_i(t) g_l(t) dt = 0, \text{ for } i \neq l.$$

In particular, in standard IEEE 802.11ah to transmit some symbols sequence $c = \{c_1, c_2, ..., c_N\}$ by means of OFDM modulation, it should be split into blocks of M symbols, M = 6 for MCS10.

Next, scrambling is performed. The scrambler structure is shown in **Fig. 5,** under delay blocks their default states are written.

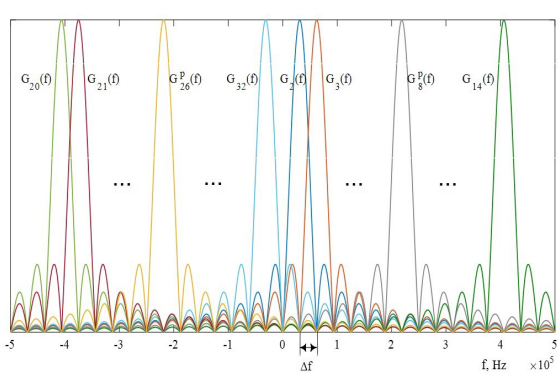

Fig. 4. *Orthogonal subcarriers in MCS10.*

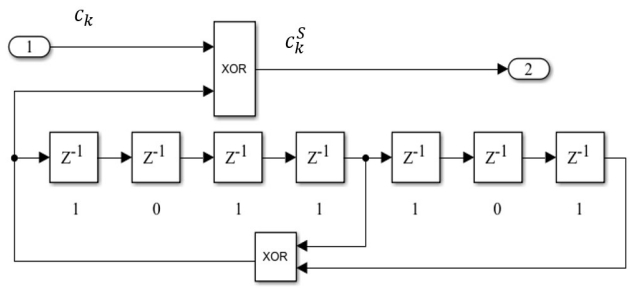

Fig. 5. *Scrambler structure.*

Scrambler's output is c^S.

Now sequence c^S goes to the convolutional encoder, the parameters of which are given in Table 1. The result is a block of 12-bits c'. Then the repetition is performed. The essence of the operation is to copy blocks of 12-bits. The copy is added modulo two with the sequence s = [1 0 0 0 0 1 0 1 0 1 1 1] and then the result is concatenated with block c', thus getting 24-bits block c^{x2}.

Afterward goes interleaving, which allows reducing the probability of correlated multiple errors. An interleaver has a size of 8 rows and 3 columns.

Then the blocks of symbols are converted by BPSK baseband modulator into modulating sequences. To get an OFDM block, pilot sequences and protective zeros are added into modulating sequence. Then IFFT for each block is performed, thus modulating the orthogonal subcarriers. In discrete-time, this looks like

$$s[n] = \sum_{k=0}^{N_{fft}-1} c_k e^{\frac{j2\pi kn}{N_{fft}}}, \ 0 \le n < N_{fft}$$

where N_{fft} is FFT size.

A CP consisting of the last 8 samples of the current OFDM block is added to its beginning, so the OFDM symbol is obtained. After that, windowing is performed that smooth the amplitude of nearby samples at the OFDM symbols borders.

Then digital-to-analog conversion is performed. After interpolation, the signal in

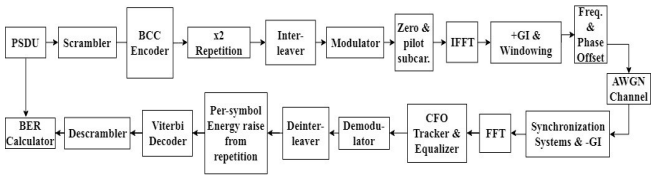

Fig. 6. *Model block diagram.*

continuous time looks like

$$s(t) = \sum_{k=0}^{N_{fft}-1} c_k e^{\frac{j2\pi kt}{T_{fft}}} = \sum_{k=0}^{N_{fft}-1} c_k g_k(t), 0 \le t < T_{fft}.$$

OFDM symbol creation process is shown in **Fig. 6**.

On the receiver side, the signal after being transferred from the carrier, amplified and analog-to-digital converted looks like

$$r[n] = s[n]e^{j2\pi f_{ot} n T_s + \theta_{ot}} + w[n],$$

where $w[n]$ – n-th sample of WGN, f_{ot} and θ_{ot} – some frequency and phase offsets.

2.2. Simulation

A simulation was performed using the synchronization systems described below, as well as equalizer with an estimation of the channel transfer function using the least square (LS) method for the case of AWGN channel with random frequency shifts (Fig. 5). MCS10 packets containing 256 octets of information were transmitted. Packets that did not pass SIG cyclic redundancy check (CRC) were not counted. That is how synchronization failures were filtered out. Frequency offset simulates effect of frequency instability in the transmitter and receiver heterodynes. The standard sets the limit of frequency instability of the transmitter's heterodyne equal to ±20 ppm. Thus, for a carrier frequency equal to 1 GHz, it turns out that the offset after receiver frequency conversion should not exceed ±40 KHz.

To compare our model with other digital data transmission systems the following relation between an energy per bit per noise power spectral density and SNR is fair

$$SNR = EbNo + 10\lg\frac{N_{dsc} + N_p}{N_{fft}} + 10\lg R$$

where $SNR = P_s/N_w$, P_s – the power of a received signal, N_w – the power of WGN, $EbNo$ – the energy per bit per noise power spectral density, N_{dsc} – number of data subcarriers, N_p – number of pilot subcarriers, N_{fft} – FFT size, R – code rate.

The formula for converting BER to the PER has the following form [9]:

$$PER = 1 - (1 - BER)^L,$$

where L – number of transmitted information bits in a packet. This formula is valid only if the errors are independent of each other.

The sample size is calculated based on the following ratio [14]:

$$N = t_\varphi{}^2 \frac{1-p}{\varepsilon_0{}^2 p},$$

where $t_\varphi = 1.96$ is the gaussian distribution quantile for significance level equal to 5%, N – the size of sample bits, ε_0 – related precision of estimation, p – the desired probability.

Incorrect operation criteria of the synchronization system have been adopted: for coarse time synchronization – case when an estimate is out of range of values $t \in [160; 208]$ μs, for fine timing – a difference of estimated value from the true value with some coarse timing estimate, for coarse frequency synchronization – case when an estimate is out of range $\Delta\overline{f}_{coarse} \in [-31.25; 31.25]$ kHz, for coarse and fine frequency synchronization together – case when an estimate is out of range $\Delta\overline{f}_{coarse} + \Delta\overline{f}_{fine} \in [-2; 2]$ kHz.

2.3. Systems

The synchronization system provides OFDM symbols selection at the moment when they begin, as well as performs frequency offset compensation. The offset occurs cause of both transmitter and receiver heterodynes

detuning and the Doppler shift.

The synchronization system divides into time and frequency systems, and each of them also into coarse and fine. Besides, there is a time synchronization up to an integer number of samples and up to fractional one. Fractional time synchronization is not considered in this paper. Now let us look at them separately.

2.3.1. Coarse timing

The coarse frequency synchronization is performed over the STF of the received packet. To do this, the autocorrelation of the signal [3] is calculated using the formula

$$R[n] = \sum_{i=0}^{L-1} r^*[n+i] r[n+i+M],$$

where ( )* - complex conjugate, L – the size of a sliding window, it influences on a value of autocorrelation estimate averaging. In this model L = 80, i.e. a half of STF length. M = 8 is a period of elementary STF sequence. Value M should be multiple of 8, although, with its increase, the delay at the autocorrelator output also increases. By varying the value of M, you can get the peak of autocorrelation at the start of LTF.

Furthermore, a time point is looked for, which corresponds to this maximum, through comparison with some level [3] or by differencing and finding a zero [2]. In this work, the second method was used, since, for the first one, the problem of determining the level arises for small values of SNR. Otherwise, the algorithm's accuracy will be unsatisfactory.

Differentiation is performed using a differentiator, which can be described by equation

$$D[n] = \sum_{i=0}^{N-1} R[n+i] - R[n+i-1],$$

where N – averaging window length, increasing this value leads to the coarse timing estimate with less noise in it, but as a payment, it also leads to some mean bias, i.e. coarse synchronization will be delayed on a corresponding number of samples. In the model N = 32.

The coarse timing is influenced by a moment when the packet was detected since its algorithm depends on the number of STF elementary sequences that are involved in the autocorrelation estimation. As the essence of the algorithm lays a conclusion that the beginning of autocorrelation estimation is located in some interval of values relative to the real beginning, it allows us to achieve a clear peak in LTF beginning moment, which means a more accurate estimate.

2.3.2. Coarse frequency synchronization

The average value of the STF autocorrelation phase was used as an initial or coarse estimate of the frequency offset

$$\Delta\overline{f}_{coarse} = \frac{\overline{\angle R[n]}}{2\pi T_c},$$

where $\angle$ – complex argument, $T_c = TM$ – elementary STF sequence duration, M = 8 – number of samples in elementary STF sequence.

The more elementary sequences are incorporated in calculations, i.e. the faster AGC and detection systems, the more accurate this estimate.

2.3.3. Fine timing

Before demodulation of OFDM symbol data part, it requires to know exact moment of its beginning. That's why after receiving a signal of LTF beginning performs capture of the next 32 samples, then FFT is performed in order to obtain frequency domain signal. In the frequency domain a product of this result $\widehat{LTS}_i$ and reference conjugated LTS. The result is transferred back to the time

domain and complex modulus is taken.

$$xcorr_i = | IFFT(FFT\{\widehat{LTS_i}\}FFT\{LTS\}^*) |,$$

where $i = 1,2,\ldots$ – number of current LTS.

Now maximum sample is in search and its argument is fine timing estimate

$$\hat{t}_{fine_i} = \arg(\max\{xcorr_i\}),$$

where arg( ) – argument of real function.

This value is a number of samples, which should be skipped to start capturing the payload.

2.3.4. Fine frequency synchronization

When OFDM symbol payload is correctly selected from the total sequence, you can refine the coarse frequency estimate. For this mean phase difference between maximum samples of cross-correlation of close LTF symbols is computed, i.e. the phase shift during one OFDM symbol caused by frequency offset.

$$\Delta\hat{f}_{fine_i} = \frac{\angle\max(xcorr_{i-1}) - \angle\max(xcorr_i)}{2\pi T_F},$$

$$\Delta\overline{f}_{fine} = \sum_{i=1}^{3}\Delta\hat{f}_{fine_i}.$$

As a result of refining the variance of aggregated estimate (sum of coarse and fine) significantly reduced (Fig. 11).

2.3.5. Frequency offset tracking

The presence of a frequency offset caused by inaccuracy of estimates of preamble-based synchronization systems described higher due to the presence of AWGN strongly affects the performance of the receiver, so it has to be compensated. For this purpose, tracking algorithms are used. Algorithms from works [4, 5] were studied.

Primarily we consider the most common pilots tracking method [5] without AWGN influence to avoid overloading of mathematics. Pilot subcarriers are averaged and phase is estimated

$$\hat{\varphi}_i = \arctan\left(\frac{\mathrm{Im}\{P_{i,7}\hat{p}_{i,7} + P_{i,-7}\hat{p}_{i,-7}\}}{\mathrm{Re}\{P_{i,7}\hat{p}_{i,7} + P_{i,-7}\hat{p}_{i,-7}\}}\right),$$

where i – index of the current OFDM symbol, P_7 – element of scrambling sequence for 7th subcarrier, $\hat{p}_7$ - received element pilot sequence for 7th subcarrier.

After estimation goes compensation of data vector

$$\hat{D}_i = D_i e^{-j\varphi_i},$$

where $D_i = \{d_{i,1}, d_{i,2}, \ldots d_{i,24}\}$ – data vector, a $d_{i,k} = c_{i,k}e^{j\varphi_i}$ – k-th data subcarrier of i-th OFDM symbol.

Now let us look on the method from [4], so called Time Frequency – Decision Feedback Loop (TF-DFL). In it instead of pilot subcarrier all non-zero subcarriers are used. This method (**Fig. 7**) uses loop that is for a phase estimation is locked in a frequency domain and for a frequency estimation is locked in a time domain.

First of all, for data vector D_i hard decision demodulation is performed. Thess estimates $C_i' = \{c'_{i,1}, c'_{i,2}, \ldots, c'_{i,24}\}$ are used to remove data from data vector, as a result only complex shift is remained for each subcarrier

$$A_i = \{e^{j\psi_{i,1}}, e^{j\psi_{i,2}}, \ldots, e^{j\psi_{i,24}}\}.$$

Then their phase is estimated

$$\varphi_i = \arctan\left(\frac{\mathrm{Im}\{A_i\}}{\mathrm{Re}\{A_i\}}\right).$$

These phases are averaged

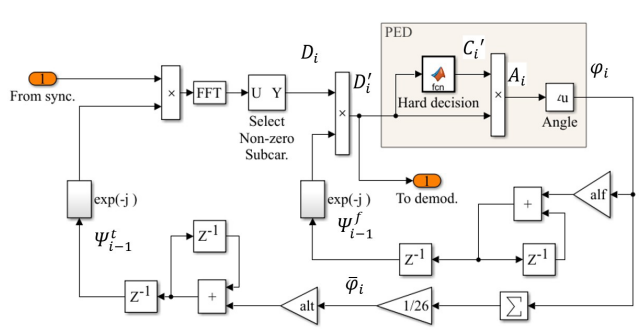

Fig. 7. *TF-DFL block diagram.*

$$\overline{\varphi}_i = \frac{1}{26}\sum_{k=1}^{26}\varphi_{i,k},$$

and the result is multiplied by time loop parameter *alt* = 0.12. Then after integration compensation vector $e^{-j\Psi_{i-1}^{t}}$ is produced. At the end it is multiplied on a current OFDM block in time domain.

To estimate residual phase offset, phases are multiplied by frequency loop parameter *alf* = 0.012. From the result compensation vector $e^{-j\Psi_{i-1}^{f}}$ is produced by integration which corrects current subcarriers in frequency domain

$$D_i' = D_i e^{-j\Psi_{i-1}^{f}}.$$

Reproduced results of TF-DFL tracking from the paper [4] are shown in **Fig. 8**, Also on the same figure BER curve for pilots-based tracking algorithm [5] is presented. This results were obtained with 802.11ah signal in MCS1 mode without coding and with bandwidth of 1 MHz.

2.3.6. Equalizer

In addition to synchronization systems, the equalizer was used in several measurements. It measured channel transfer function with the LS method [15]. This type of equalizer was chosen for its ease in implementation.

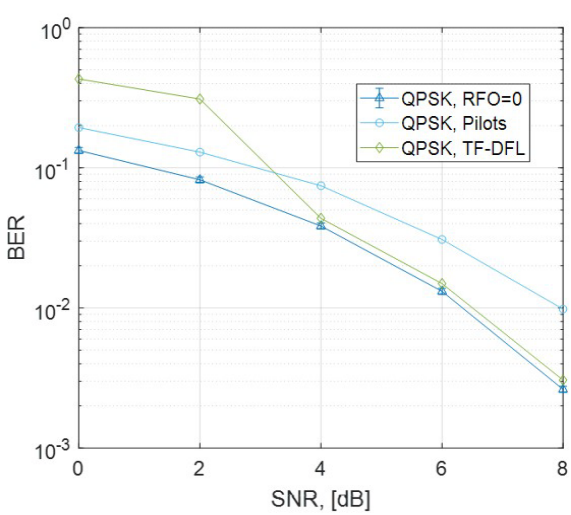

Fig. 8. *Reproduced BER curves from* [4].

Besides, there was no multipath propagation in the model, which means that LS equalizer should not lose in performance against its optimal analog. The estimation was performed using synchronized LTF symbols. The estimates were averaged, based on the assumption that the channel change on the packet duration is insignificant. This process is described in detail below.

Let the spectral density (SD) of the received signal for the *k*-th subcarrier be

$$R_f'[k] = R_f[k]H_f[k] + W_f[k],$$

where R_f[k], W_f[k], H_f[k] – accordingly, the SD of the transmitted signal, the SD of AWGN, the channel transfer function for *k*-th subcarrier.

To estimate transfer function with LS method for *i*-th OFDM symbol on *k*-th subcarrier the following equation is used

$$\widehat{H}_f^{\,i}[k] = \frac{R_f'^{i}[k]}{R_f^{i}[k]}.$$

Then the 4 estimates of the LTF are averaged

$$\overline{H}_f[k] = \frac{1}{4}\sum_{i=1}^{4}\widehat{H}_f^{\,i}[k].$$

Finally, the signal is equalized in the frequency domain

$$R_f''[k] = \frac{R_f'[k]}{\overline{H}_f[k]}.$$

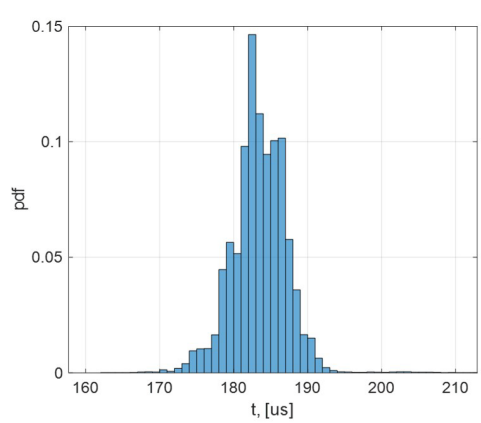

Fig. 9. *PDF of coarse timing.*

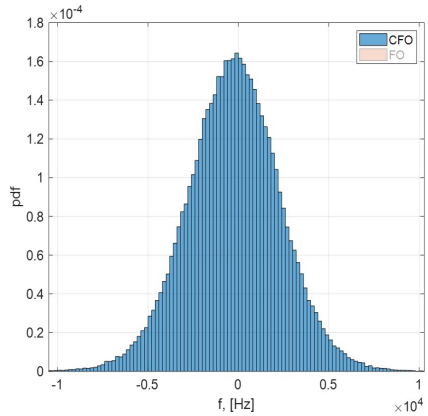

Fig. 10. *PDF of coarse frequency estimates.*

3. RESULTS

This section presents measurements of synchronization system performance for a sample size of 180000 packets with SNR = –2 dB, random phase, and frequency offset (uniform distribution in range ±π and ±40 kHz). Along with BER measurements which are done for a significance level of 5% and relative precision of 10%.

In **Fig. 9** coarse timing histogram is presented.

In **Fig. 10** and **11** coarse and a sum of coarse and fine frequency offset estimates histograms are presented.

In **Table 2** probabilities of preamble-based algorithms failures are presented as well as relative precisions ε0 of these probabilities and confidence intervals for a significance level of 5%. This table shows that the main contribution to the total probability is made by the fine timing and fine frequency synchronization. Based on this fact, the number of packets was limited to the number, which gives close to 10% precision in measuring the probability of failure of these systems.

Table 2

Failure probabilities for timing and frequency synchronization

System	P_{ow}	ε_0	ε_∂
Coarse timing	3.333e-4	0.253	8.433e-5
Fine timing	2.400e-3	0.094	2.260e-4
Coarse frequency sync	1.667e-5	1.131	1.186e-5
Fine frequency sync	2.300e-3	0.096	2.213e-4
∑	5.050e-3	-	-

During the BER estimation the following cases were considered:

1. Ideal synchronization without equalizer for MCS0 and MCS10 (**Fig. 12**, Ideal sync.).
2. Preamble based synchronization with LS equalizer and zero residual frequency offset (Fig. 12, MCS10, RFO = 0).
3. Preamble based synchronization with LS equalizer and TF-DFL RFO compensation (Fig. 12, MCS10, TF-DFL).
4. Preamble based synchronization with LS equalizer and pilots-based RFO compensation (Fig. 12, MCS10, Pilots).
5. Preamble based synchronization with LS equalizer and pilots-based RFO compensation for traveling pilots (Fig. 12,

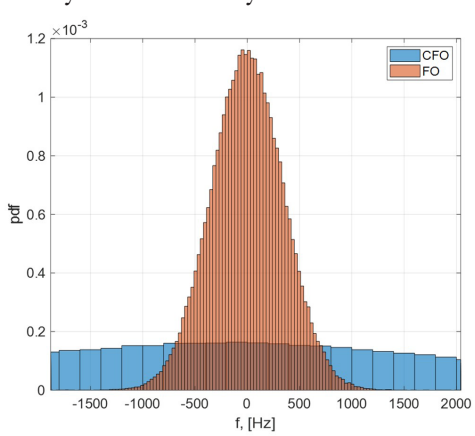

Fig. 11. *PDF of sum of coarse and fine frequency estimates.*

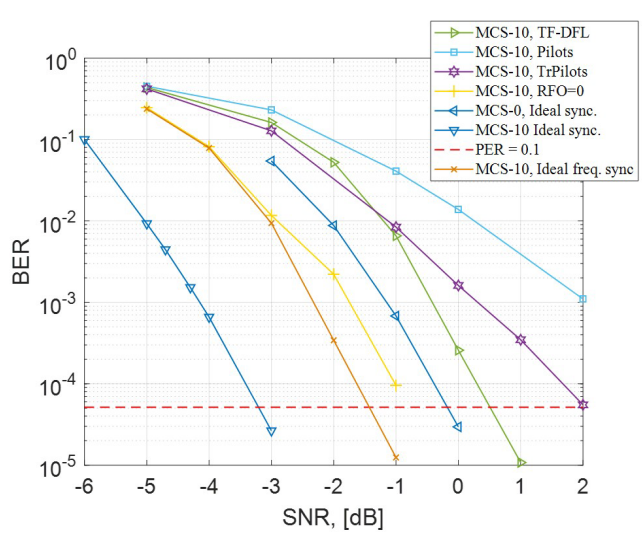

Fig. 12. *BER curves of our receiver model.*

MCS10, TrPilots)

6. Preamble based timing with LS equalizer and zero frequency offset (Fig. 12, MCS10, Ideal freq. sync.).

4. DISCUSSION

The probability of coarse timing failure that was obtained (Table 2) is close to the probability presented in the working group (TGah) [16].

The synchronization errors and LS equalizer errors lead to 2.4 dB degradation of receiver noise immunity relative to the theory.

Besides, a considerable contribution is made with timing systems failures that are illustrated with (MCS10, Ideal freq. sync.) curve in Fig. 12. In this case, degradation is around 1.8 dB.

At first, it is not clear why there is a difference in noise immunity between these two synchronization cases since for both there is no RFO. Here, the relationship between time and frequency estimates from each other plays a role. Before estimating the exact timing, the frequency offset is compensated with a coarse frequency estimate. As a result, a significant inaccuracy in the frequency estimate leads to an error in time estimate since the last one evaluated with cross-correlation.

Hence, the potential gain from improving frequency estimation systems is 0.6 dB. At the same time, the main contribution to degradation is made with fine estimation systems, since the probabilities of coarse and fine differ by about 2 orders of magnitude. As for the gain due to the improvement of timing systems, the probability of an error in the fine timing system is about an order of magnitude higher than in the coarse timing (**Table 2**), and it can be said that the improvement of the fine timing will give a gain in the receiver's noise immunity.

For the case of receiver operation with RFO compensation systems, the minimum loss (TF-DFL) relative to the theoretical boundary of MCS10 is very significant and is 3.7 dB. Even relative to the idealized case for MCS0, the TF-DFL method applied to MCS10 loses 0.7 dB.

Comparing the receiver with the TF-DFL algorithm with zero RFO case gives 1.3 dB loss to the first one. Therefore, even within the framework of the studied preamble-based synchronization systems, it is possible to increase the receiver's noise immunity by applying more advanced RFO compensation methods.

Comparing the noise immunity of classical pilot-based and TF-DFL methods, it is seen that TF-DFL shows better results, and the classical method does not help even traveling pilots (~3.5 dB gained). TF-DFL algorithm shows better results starting with BER ~1e-2, as approaching the receiver working area (PER ≤ 0.1) the gain reaches 1.5 dB.

The last thing has to be noted, the fact that LS equalizer is not the best solution in terms of receiver noise immunity, so it is possible to use better channel estimation techniques to increase it.

5. CONCLUSION

In this paper, the synchronization and tracking algorithms for IEEE 802.11ah receiver have been studied. The measurement results which were obtained during simulation in MATLAB show that usage of classical preamble-based synchronization systems with LS equalizer leads to a degradation of the BER vs SNR curve by ~2.4 dB relative to the idealized theoretical case. The main contribution to degradation is made with

equalizer and fine synchronization systems. Therefore, to increase receiver noise immunity, it is necessary to optimize these systems.

The best result in RFO compensation is achieved with TF-DFL method (PER = 10% for SNR = 0.5 dB). At the same time, other more optimal solutions for RFO compensation are possible.

REFERENCES

1. 802.11ah-2016 - *IEEE Standard for Information technology-Telecommunications and information exchange between systems - Local and metropolitan area networks--Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Sub 1 GHz License Exempt Operation.*
2. Wang Y, Sun S, Tan PH, Kurniawan E. Baseband receiver design for IEEE 802.11ah. *TENCON 2017-2017 IEEE Region 10 Conference*, Penang, Malaysia, 2017. DOI: 10.1109/TENCON.2017.8227976.
3. Hajjar CE. *Synchronization Algorithms for OFDM Systems (IEEE802.11ah, DVB-T) Analysis, Simulation, Optimization and Implementation Aspects*. Erlangen, 2007, 166 p.
4. Kuang L, Ni Z, Lu J, Zheng J. A Time-Frequency Decision-Feedback Loop for Carrier Frequency Offset Tracking in OFDM Systems. *IEEE Transactions on wireless communications*, 2005, 4(2):367-373.
5. Jimenez VPG, Armada AG, Gonzalez-Serrano FJ. Design and Implementation of Synchronization and AGC for OFDM-based WLAN Receivers. *IEEE Transactions on Consumer Electronics,* 2004, 50(4):1016-1025.
6. Halasz D. IEEE P802.11 Wireless LANs. *Categories of TGah Use Cases and Straw Polls,* IEEE 802.11-11/0301r2, 2011.
7. Yusupov RR. Modelirovanie peredachi trafika mezhmashinnogo vzaimodeistviya v setyakh Wi-Fi HaLow s ispolzovaniem mekhanisma okna ogranichennogo dostupa v rezhime s peresecheniem granic. [M2M traffic modeling in Wi-Fi HaLow Networks with usage of restricted acces window mechanism in cross slot boundary mode]. *Proceedings of the 42nd Interdisciplinary School-Conference of IITP RAS "Information Technologies and Systems* 2018", p. 430-439 (in Russ.).
8. Loginov VA, Lyakhov A, Khorov E, Analiticheskaya model raboty mekhanizma mgnovennoy retranslyacii paketov v setyakh IEEE 802.11ah [Transmission opportunity (TXOP) analytical model in IEEE 802.11ah networks]. *Proceedings of the 39nd Interdisciplinary School-Conference of IITP RAS "Information Technologies and Systems* 2015, p. 1140-1453 (in Russ.).
9. Hazmi A, Rinne J, Valkama M. Feasibility Study of IEEE 802.11ah Radio Technology for IoT and M2M use Cases. *IEEE Globecom Workshops,* p. 1687-1692, Anaheim, CA, USA, 2012. DOI: 10.1109/GLOCOMW.2012.6477839.
10. Aust S, Prasad RV, Niemegeers IGMM, Outdoor Long-Range WLANs: A Lesson for IEEE 802.11ah. *IEEE Communications Surveys & Tutorials,* 2015, 17(3):1761–1775.
11. Porat R. Link Budget, IEEE 802.11-11/0552r2.
12. Perahia E, Stacey R. *Next Generation Wireless LANs 802.11n and 802.11ac*, 2nd ed. Cambridge, U.K., Cambridge Univ. Press, 2013.

13. Gupta RK, Jain A, Singodiya P. Bit error rate simulation using 16 qam technique in matlab, *International Journal of Multidisciplinary Research and Development*, 2015, 2(5):59-64.
14. Kleine J. *Statisticheskie metodi v imitacionnom modelirovanii [Statistical methods in simulation modeling]*. Iss. 2. Moscow, Statistika Publ., 1978, 335 p.
15. Hwang J. *Simplified Channel Estimation Techniques for OFDM Systems with Realistic Indoor Fading Channels*. Waterloo, Ontario, Canada, 2009.
16. Vermani S et al. Preamble Format for 1 MHz. *IEEE-802.11-11/1482r4*, 2012.