PERFORMANCE ANALYSIS OF A FREQUENCY OFFSET CORRECTION METHOD FOR AN OFDM SYSTEM

By: S. Rauhala, Carleton University, Canada; N. Birkett, IceFyre Semiconductor, Canada;
J. Wight, Carleton University, Canada

Abstract

The ability to track carrier frequency offsets in IEEE 802.11a architectures is imperative for the proper decoding of high data-rate information. A method to detect and correct a carrier frequency offset in an IEEE 802.11a architecture is proposed. The tracking frequency offset (TFO) design exploits the use of the pilot symbols found in each OFDM (Orthogonal Frequency Division Multiplexed) data symbol. The system is simulated under multi-path and noise conditions that would be expected from an indoor wireless local area network. Simulation results show that the TFO architecture is capable of accurately detecting and correcting a small frequency offset that could typically occur in the data portion of a packet.

Keywords:

OFDM, Pilot-Assisted, Frequency Offset Tracking, IEEE 802.11a

Introduction

IEEE 802.11a specifies an OFDM format for high data rate transmission, ideal for wireless indoor multimedia content [1]. Data rates vary from 6Mbps with BPSK subcarrier mapping to 54 Mbps with 64-QAM subcarrier mapping. However, the indoor environment has many characteristics that become more of a design challenge as data rates increase. Multi-path, frequency offsets, phase offsets and sampling offsets are some of the distortions that can occur during transmission. Therefore, synchronization is very important to detect and correct these offsets. This paper provides the reader with a basic understanding of OFDM. From there, the paper describes two tracking frequency offset (TFO) architectures, and compares and contrasts them with each other. Finally, the paper concludes with a discussion and analysis of the simulation results.

OFDM and IEEE 802.11a

An 802.11a packet consists of a preamble, followed by a SIGNAL and the data symbols. A symbol is 4us long, broken into 3.2us of data and an 800ns cyclic prefix. The cyclic prefix (CP) occurs before the data and is a repetition of the last 800ns of the data portion. Since the average indoor multi-path delay spread is shorter than the length of the cyclic prefix, the effects of multi-path are reduced [1].

Data bits are convolutionally encoded, reordered, bit interleaved, mapped to a constellation and subdivided onto 48 data subcarriers. Each data symbol is made up of 48 data subcarriers and 4 pilots on subcarrier numbers 7, 21, -21 and -7. In total there are 52 nonzero subcarriers and 12 zero subcarriers [1]. The subcarriers are orthogonal since they are separated in the frequency domain by integer multiples of 312500 Hz or 1/3.2us. An IFFT is used to combine the subcarriers for transmission [1]. OFDM reduces the effects of frequency selective fading by spreading the data over several frequencies [3]. In the receiver, an FFT converts the transmitted signal back to the individual subcarriers.

There are three main types of offsets that could occur between transmission and decoding of the data symbols. A carrier frequency offset (CFO) will cause all subcarriers to experience the same frequency offset - the data constellation will rotate at an even rate and intercarrier interference is possible. A carrier phase offset (CPO) occurs when the carrier is not aligned with the data during modulation. This will cause a single static rotation of the constellation to the value of the phase offset. A sampling frequency offset (SFO) occurs when the receiver and transmitter clocks are not exactly the same rate and will cause the constellation to slowly drift throughout the duration of the packet.

Although the purpose of the preamble is to detect an initial frequency offset, the estimate may be poor since the preamble is relatively short and multi-path and noise could severely affect the result. As well, the packet could be up to 4095 octets [2;6], long enough for the frequency offset to change throughout the packet. For example, a small remaining frequency offset of 5kHz will cause the data to rotate around the constellation at a rate of 0.1257 radians each symbol ($2\pi 5000*4e\text{-}6$). For BPSK data, this might be manageable, however, for 64-QAM this rotation is unacceptable. Therefore, the CFO must be tracked during the steady state operation of the data symbols.

CFO Detection Algorithm

According to the Fourier Transform, the following property holds true [4]

$$x(n)e^{jw_o n} = X(w - w_o)$$

Equation 1

where w_o is the fixed frequency offset. All frequencies are shifted by the same offset in the frequency domain. Since the data symbols have 4 pilots of known value and phase, this information can be used to track the frequency offset, leading to the following algorithm:

$$\phi = \frac{\theta_7 + \theta_{21} + \theta_{-21} + \theta_{-7}}{4}$$

Equation 2

where θ_n is the phase of subcarrier n. Performing an average over the 4 pilots gives a better estimate of the frequency offset, ϕ.

Hsu et al [5] suggests a similar approach, except they take the difference between two data symbols rather than using the value of a single data symbol, i.e.

$$\Delta\phi(n) = \sum_k \angle X_{n,k} - \angle X_{n-1,k}, k = -21, -7, 7, 21$$

Equation 3

CFO Correction Method

Theoretically, both of the above detection methods will work, but they will require different correction methods. To correct a frequency offset, the phase error caused by the frequency offset must be removed. From Equation 1, the frequency offset must be removed by applying $e^{-jw_o n}$.

Three methods exist to correct a frequency offset:

1. Time Domain Correction: Correct the current symbol with the value of the previous symbols CFO estimate by feeding back the estimate to the time domain for correction, as seen in Figure 1.

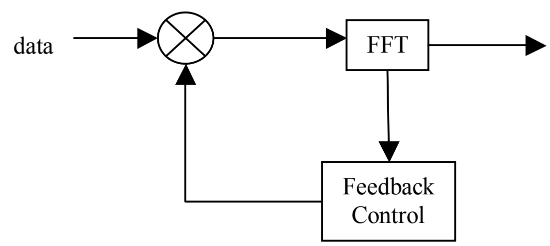

Figure 1: TFO Detect/Correct Method #1

2. Frequency Domain Correction: Correct the current symbol with the value of the current symbols CFO estimate by feeding forward the CFO into the frequency domain. This method is similar to an architecture proposed by Torrance and Hanzo [6].

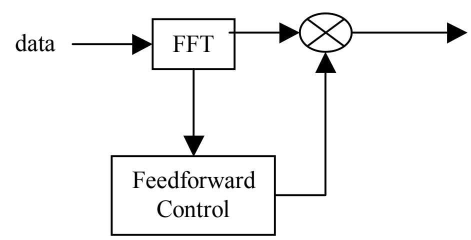

Figure 2: TFO Detect/Correct Method #2

3. Time Domain Correction: Correct the current symbol with the current symbols CFO using a combination of feedback and feedforward control. A copy of the symbol must be stored while the symbol is sent through the FFT and the CFO estimate is determined. Then the stored symbol is corrected before being sent through the FFT for normal symbol decoding. This method has large delays because each symbol is sent through the FFT twice, so it will not be considered in this paper.

Without correction, the pilots rotate around the complex plane with a delta phi of $\omega_o T$ from symbol to symbol.

Since the phase difference between subsequent symbols will be equivalent to the phase of the frequency offset applied, Equation 3 appears to be useful. However, this depends on the selected correction method. Since Method #2 involves detection *before* correction, the absolute phase offsets of each data symbol will be rotating around the complex plane. Therefore, each data symbol must be corrected for the CFO due to rotation as well as the initial phase offset. Equation 4 illustrates the rotation of uncorrected symbols. Without correction, symbol 3 has rotated three times further than symbol 1, so it requires a different correction. Notice the difference between the symbols is constant and equal to θ_{CFO}.

$$\phi_1 = \theta_{inital} + \theta_{CFO}$$
$$\phi_2 = \phi_1 + \theta_{CFO} = \theta_{initial} + \theta_{CFO} + \theta_{CFO}$$
$$\phi_3 = \phi_2 + \theta_{CFO} = \theta_{initial} + \theta_{CFO} + \theta_{CFO} + \theta_{CFO}, etc$$

Equation 4: ϕ_n is the phase of the FO detected by averaging the pilots for symbol n

On the other hand, Method #1 involves correcting the symbol *then* detecting an offset. After correction in the time domain, the symbol will be transferred to the frequency domain by the FFT, after which the pilots will be analyzed. Ideally, after correction the symbol will no longer be rotating, so the subsequent phase difference between symbols should be zero. However, if the phase difference of the pilots is not zero, a residual CFO exists which must also be applied to the data symbol. In this case, Equation 3 is the correct algorithm to chose, since it will determine the true value of the frequency offset, ignoring any possible initial phase offsets.

Results

For testing purposes, the data was modulated using 64-QAM and all coding and scrambling was performed as per the 802.11a specifications. Additive White Gaussian Noise was added, giving an SNR of 20dB, and multi-path with a delay spread of 50ns was applied. To test the architectural designs, a frequency offset was applied to the data in the time domain. Then the data was sent through each tracking method. Figure 3 shows the number of bit errors for each method at each applied tracking frequency offset. Method #2 performs much better overall. Method #1 performs well only for very small frequency offsets, which limits its application.

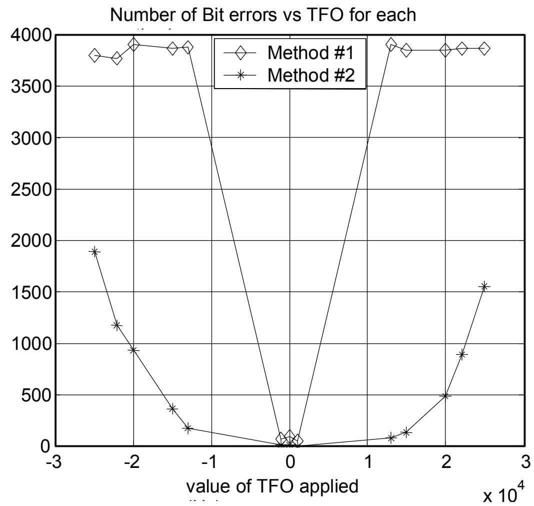

Figure 3: Comparison of bit errors versus applied frequency offset for each method.

The selected algorithm must perform well in the presence of other offsets. Figure 4 shows a comparison of the average CFO estimation error versus the applied CFO. Figure 5 shows the standard deviation of the CFO estimation error for both methods. Overall, Method #2 performs a better estimation, as seen by a lower standard deviation and an average error closer to zero. In the presence of a sampling frequency offset or a carrier phase offset, both methods perform equally well.

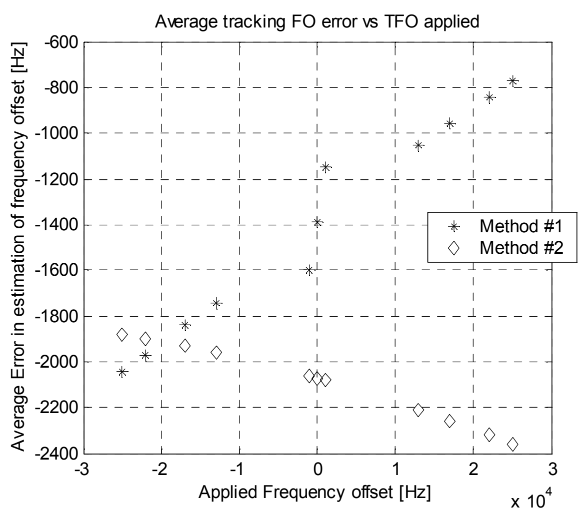

Figure 4: Average CFO estimation error versus applied CFO (SNR=20dB)

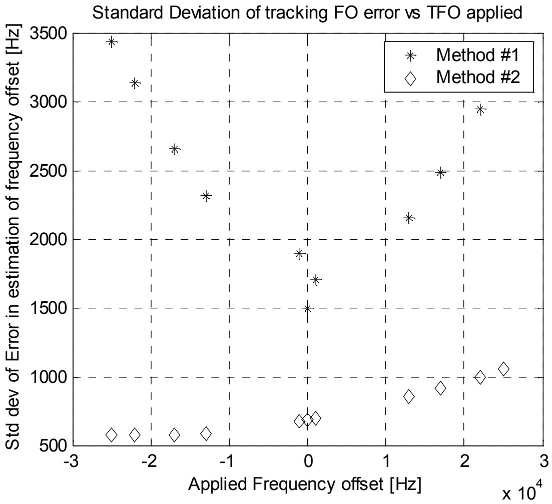

Figure 5: Standard Deviation of CFO estimation error versus applied CFO (SNR=20dB)

Finally, Figure 6 shows a BER vs SNR curve. Method #2 outperforms method #1 for all signal to noise ratios. Method #2 also more closely resembles the expected results. The expected results are based on the following equation:

$$SNR = 10\log_{10}\left(\frac{E_b}{N_o}\frac{R_T}{B}\right)dB$$

Equation 5

R_T is the transmission rate, B is the Bandwidth and E_b/N_o is the bit energy to noise ratio.

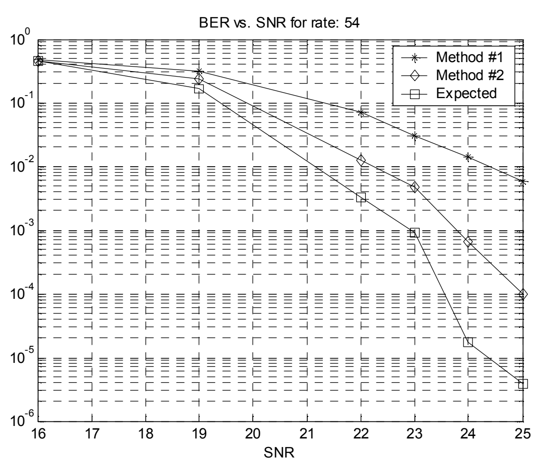

Figure 6: BER curve for both methods.

Conclusion

The results indicate that correction in the frequency domain is more robust than a feedback correction to the time domain. Fortunately, frequency domain correction (Method #2) uses feedforward control, which requires less hardware. Method #2 can also overcome residual phase offsets since it uses the absolute phase value of each symbol, rather than the phase difference between symbols. Overall, frequency domain correction outperforms time domain correction in all test cases examined.

References

1. Petrick, Al; O-Hara, Bob; *IEEE 802.11 Handbook – A Designer's Companion*; IEEE Press; 1999.
2. *Supplement to IEEE standard for Information technology –Part 11: Wireless LAN Medium Access Control9MAC) and Physical Later(PHY) specifications: High speed Physical Later in the 5 GHz band*; IEEE Std 802.11a-1999.
3. Richharia, M; *Satellite Communication Systems;* McGraw-Hill; 1999.
4. Proakis, John; *Digital Signal Processing;* 3rd ed; Prentice Hall; 1996.
5. Hsu, Chien-Fang; Huang, Y.H.; Chiueh, T. D.; Design of an OFDM receiver for High-speed Wireless LAN; *The 2001 IEEE International Symposium on Circuits and Systems;* ISCAS 2001.
6. Torrance, J.M.; Hanzo, L.; Comparative Study of Pilot Symbol Assisted Modem Schemes; *Proceedings of Radio Receivers and Associated Systems Conference,* RRAS '95; Sept. 26-28, 1995.