Frame Synchronization for Discrete Multi-Tone Wireline Acoustic Sensor Network B.Sc. Guided Research

Zinan Liu
Jacobs University Bremen/ ATLAS-ELEKTRONIK GmbH

Supervisors: Prof. Dr.-Ing. W. Henkel
Jacobs University Bremen
Dipl.-Ing. V. Burstein
ATLAS-ELEKTRONIK GmbH

June 1, 2015

1 Abstract

This Guided Research focuses on frame synchronization for the DMT-based transmission system, which is used on a proposed acoustic sensor network for sonar applications. The time offset estimate not only needs to be robust and accurate to maintain the orthogonality of the various sub-channels in the DMT system, but also computationally efficient to reduce hardware cost. The current frame synchronization method used for the proposed sensor network is the maximum likelihood (ML) estimation [1]. This method is relatively robust and accurate, however requires a large memory at the receiver for storing the overhead of the data frames to compute the average. The overhead typically consists of hundreds of frame symbols, and the averaging of the time offset estimation for each frame symbol consumes a significant amount of computation time.

This paper compares two different frame synchronization methods with the ML estimation method, namely the Schmidl and Cox method [2], and the Park method [3] in order to find an optimum solution. All three methods are evaluated in the flat channel free of noise, the AWGN channel, and the wireline bus channel simulations. The results suggest that the Schmidl and Cox method has a plateau in its proposed timing metric, which introduces great uncertainty in the estimation. The Park method, nevertheless, shows promising results in both the estimation accuracy and the computation efficiency. Two peaks of the timing metric in the Park method exhibit great robustness and accuracy in 300 simulations, and it only needs one training symbol appended at the beginning of the information frame symbol for the synchronization. Even though the peaks are sensitive to SNR, the simulation shows that in the worst-case scenario, the transmitting power per DMT sender (given the fixed noise), or equivalently the SNR, is more than suitable for the implementation of the Park method. Therefore it can be concluded that the Park method is the optimum solution out of all three in the frame synchronization for the proposed DMT wireline acoustic sensor network.

Contents

2 Introduction

Acoustic data transmission for the future sonar systems is expected to reach the data rate of around 10 Mb/s per sensor node, or 1-2 Gb/s per system. Taking account of the underwater environment with varying conditions and mechanical restrictions on the cables, the feasible physical cabling is a DC powered linear bus system consisting of a Multi-Carrier Twisted Pair (MC-TP) for Power-Line Communications (PLC) [4]. Therefore, the aforementioned communication system is capable of supplying power, providing high throughput, and achieving better adaptability of channel conditions due to environmental factors. In realizing such a system, Discrete Multi-Tone (DMT) is deployed to partition the channel into various sub-channels and optimally allocate data to maximize each sub-channel throughput [4]. One of the important aspects of realizing such a DMT system is the frame synchronization. The receiver must be able to estimate the time offset in order to find the start of the transmitted frames without any data loss. The aim of this Guided Research is to compare three different frame synchronization algorithms so as to find the most suitable one for the proposed DMT wireline acoustic sensor network.

3 DMT Overview

3.1 Multi-carrier System

In order to exhaustively utilize the frequency band without Inter-symbol Interference (ISI), the spectrum is split into many quasi frequency-flat sub-channels. The sub-channels are independent of one another and operate in parallel with their own modulated carriers [5], which can be seen as simultaneously transmitting various symbols at low symbol rate. Orthogonal Frequency Division Multiplexing (OFDM) and Discrete Multi-Tone (DMT) are the two commonly used vector coding schemes for multi-carrier system nowadays. The essential difference of OFDM and DMT lies in the range of the frequency spectrum, and whether the channel information is known at the transmitter so as to apply different bit loading algorithms. An OFDM system operates in passband, and the channel is unknown at the transmitter, thus all sub-channels are allocated with the same amount of bits suited for one-way channels broadcast in wireless communication. However, a DMT system operates in baseband, and the total energy ε_n and data rate b_n are optimized given the known channel at the transmitter, contributing to wire-line communication [6].

3.2 QAM Modulation

In the transmitter, the parallel data from the serial to parallel converter is mapped to certain constellation scheme for each sub-channel. In the proposed wireline sensor network, the multi-carrier QAM modulation is used with a constellation size of up to 10.

As each sub-channel in the DMT system has a modulated carrier, the input to the multi-carrier modulator is a vector

$$\mathbf{X} = \begin{bmatrix} X_N \\ X_{N-1} \\ \vdots \\ X_0 \end{bmatrix},$$

where there are $N+1$ sub-channels. At the zero and Nyquist frequencies, the sub-symbols X_0 and X_N, respectively, are one-dimensional real-valued, and thus can at best be Pulse Amplitude Modulated (PAM) [5]. The other sub-symbols from X_1 to X_{N-1} are Quadrature Amplitude Modulated (QAM), a two dimensional generalization of Phase Amplitude Modulation (PAM). The two basis functions are defined as

$$\varphi_1(t) = \sqrt{\frac{2}{T}}\mathrm{sinc}\left(\frac{t}{T}\right)\cos\omega_c t\,,$$
$$\varphi_2(t) = -\sqrt{\frac{2}{T}}\mathrm{sinc}\left(\frac{t}{T}\right)\sin\omega_c t\,,$$

where $\text{sinc}\left(\frac{t}{T}\right)$ can be replaced by any Nyquist pulse shape, and ω_c is the carrier frequency in radians ($\omega_c \geq \pi/T$) [7].

There are three major QAM constellation schemes, namely the QAM Square Constellation, QAM Cross Constellation, and Offset QAM (OQAM). The performance of all the constellation methods can be evaluated by the coding gain γ, fundamental gain γ_f, shaping gain γ_s, and PAR as discussed in [7].

3.2.1 QAM Square Constellation

The QAM square constellations are mapping the $M = 4^{\bar{b}}$ symbols each to $\bar{b}$ bits per dimension, thus the signal points are placed at the coordinates $\pm\frac{d}{2}$, $\pm\frac{3d}{2}$, $\pm\frac{5d}{2}$, ..., $\pm\frac{(\sqrt{M}-1)d}{2}$ in each dimension. As illustrated in Fig. 1, it can be seen that the constellations are essentially the Cartesian products of 2-PAM with itself and 4-PAM with itself, for the 4 QAM and 16 QAM, respectively. Thus in general, the Cartesian product of two $\sqrt{M}$-PAM constellations gives the square M-QAM constellation [7].

The statistics of the constellation is then calculated as follows [7]:

average energy:

$$\begin{aligned} E_{M-QAM} = E_x = 2\bar{E}_x &= \frac{1}{M}\sum_{i,j=1}^{\sqrt{M}}(x_i^2 + x_j^2) \\ &= 2\frac{1}{\sqrt{M}}\sum_{i=1}^{\sqrt{M}} x_i^2 \\ &= 2E_{\sqrt{M}-PAM} \\ &= d^2\left(\frac{M-1}{6}\right), \end{aligned}$$

where x_i and x_j are the horizontal and vertical distance from the constellation symbol to the origin,

average energy per dimension:

$$\bar{E_x} = d^2\left(\frac{M-1}{12}\right),$$

minimum distance:

$$d_{min} = d = \sqrt{\frac{6E_x}{M-1}} = \sqrt{\frac{12\bar{E}_x}{M-1}},$$

average bit rate:

$$\bar{b} = \frac{1}{2}\log_2\left(\frac{6E_x}{d^2}+1\right) = \frac{1}{2}\log_2\left(\frac{12\bar{E}_x}{d^2}\right),$$

symbol error probability

$$P_e = 4\left(1-\frac{1}{\sqrt{M}}\right)Q\left[\frac{d}{2\sigma}\right] - 4\left(1-\frac{1}{\sqrt{M}}\right)^2\left(Q\left[\frac{d}{2\sigma}\right]\right)^2.$$

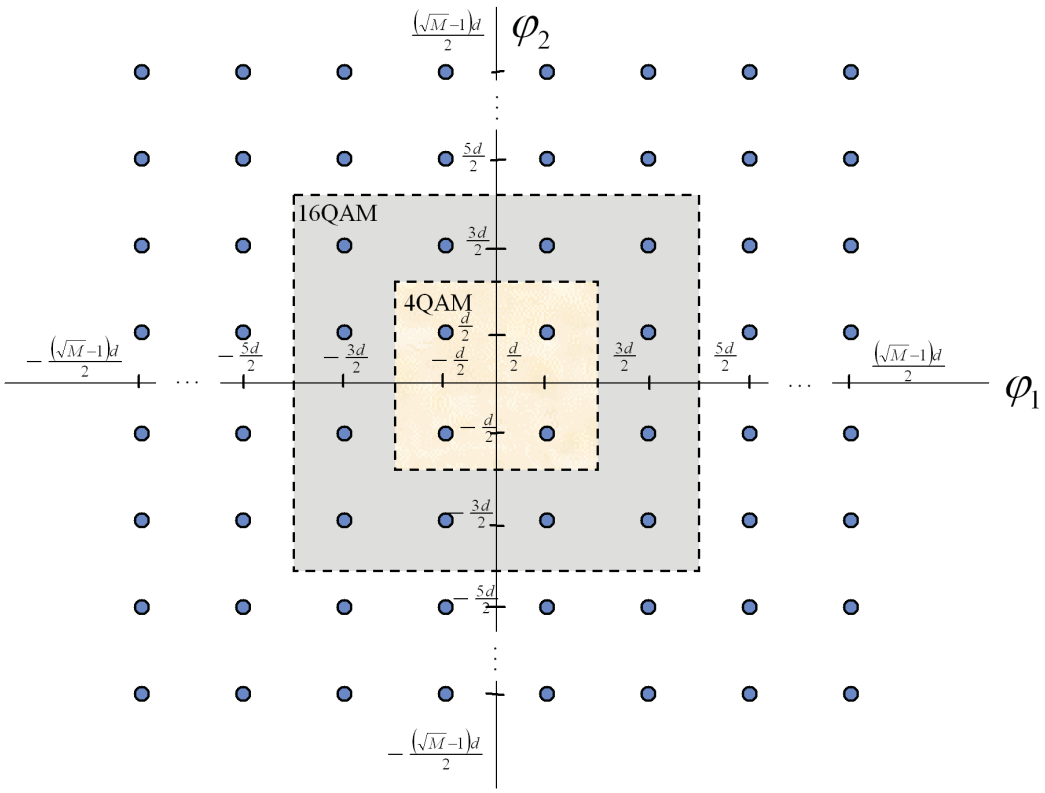

Figure 1: Square QAM constellation [8]

For large M, $\bar{E}_x \approx \frac{d^2}{12}M = \frac{d^2}{12}4^{\bar{b}}$, the same result as obtained by continuous approximation, which suggests that the energy computation error caused by continuous approximation becomes negligible as M increases. Now, for fixed M but increasing b, the following average energy increment recursion is observed while maintaining constant minimum distance [7]:

$$E_x(b+1) = 2E_x(b) + \frac{d^2}{6}.$$

3.2.2 QAM Cross Constellation

A QAM Cross Constellation is constructed by adding 2^{b-1} data symbols to the QAM square sides of $b-1$ bits per symbol, excluding the corners [7], as illustrated in Fig. 2 .

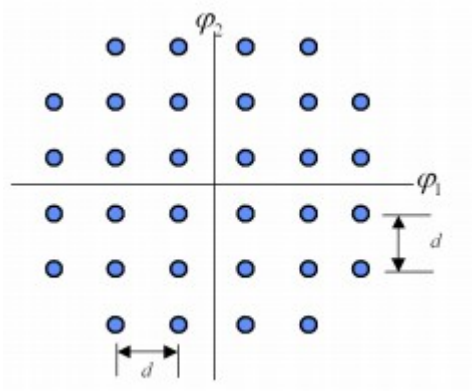

Figure 2: 32CR QAM constellation [9]

The statistics of the constellation is then calculated as follows [7]:

energy of the inner square:

$$E_x(inner) = \frac{d^2}{6}(2^{b-1}-1).$$

total sum of energies:

$$\begin{aligned} E &= \frac{d^2}{4}(4)\sum_{k=1}^{2^{\frac{b-3}{2}}}\sum_{l=1}^{3\cdot 2^{\frac{b-5}{2}}}[(2k-1)^2+(2l-1)^2] \\ &= \frac{d^2}{4}\left[\frac{13}{32}2^{2b}-2^{b-1}\right], \end{aligned}$$

average energy:

$$\begin{aligned} E_x &= \frac{2E-2^{b-1}E_x(inner)}{2^b} \\ &= \frac{d^2}{6}\left[\frac{31}{32}M-1\right], \end{aligned}$$

minimum distance:

$$d_{min} = d = \sqrt{\frac{6E_x}{\frac{31}{32}M-1}} = \sqrt{\frac{12\bar{E}_x}{\frac{31}{32}M-1}} = \sqrt{\frac{6E_x}{\frac{31}{32}4^{\bar{b}}-1}},$$

bounded symbol error probability

$$P_e \ll 4\left(1-\frac{1}{\sqrt{2M}}\right)Q\left[\frac{d_{min}}{2\sigma}\right]-4\left(1-\sqrt{\frac{2}{M}}\right)\left(Q\left[\frac{d_{min}}{2\sigma}\right]\right)^2$$
$$< 4Q\left[\frac{d_{min}}{2\sigma}\right].$$

For large M, $E_x \approx \frac{31d^2}{192}M = \frac{31d^2}{192}4^{\bar{b}}$, equivalent to the result above, using a continuous approximation, which indicates that for increasing M, the energy computation error is negligible. Now, for increasing b, while maintaining constant minimum distance, the following average energy increment recursion is observed [7]:

$$E_x(b+1) = 2E_x(b) + \frac{d^2}{6}.$$

3.2.3 OQAM

OQAM works by "offsetting" the two dimensions by $T/2$ using the alternative basis functions [7]

$$\varphi_1(t) = \sqrt{\frac{2}{T}}\text{sinc}\left(\frac{t}{T}\right)\cos\left(\frac{\pi t}{T}\right),$$
$$\varphi_2(t) = -\sqrt{\frac{2}{T}}\text{sinc}\left(\frac{t-T/2}{T}\right)\sin\left(\frac{\pi t}{T}\right).$$

Therefore the OQAM has a lower rate of change, i.e., derivative of $x(t)$, than QAM for successive transmission.The purpose is to effectively reduce the bandwidth of the transmitted signals when the sinc functions cannot be implemented perfectly, thus relaxing hardware requirements for non-linear transceiver amplifiers. For one-shot transmission, OQAM and QAM are the same [7].

3.3 Channel Partitioning

Channel partitioning is done after the QAM modulation of the parallel data bits, in order to split the channel into $N+1$ ideally flat, independent, and parallel sub-channels in a baseband system. Ideally the partitioning is done by convolving the parallel sub-symbols with their corresponding orthonormal basis function $\varphi_k(t)$ from the set $\{\varphi_n(t)\}$. Each sub-channel is coupled at the carrier frequency of $f_k = k\cdot\frac{f_s}{N}$, where f_s is the sampling rate, and $k = 0,..\frac{N}{2}-1$(due to conjugate constraints). In the demodulator, the received signals are passed through a matched filter to process each baseband sub-channel, and a maximum-likelihood detector to reconstruct the transmitted sub-symbols independently.

The basis function can be any orthonormal basis from the set of functions $\{\varphi(t)\}$ [5] if

$$\int_{-\infty}^{\infty}\varphi_m(t)\varphi_n(t)\,\mathrm{d}t = \delta_{mn}, \quad \text{where } \delta_{mn} = f(n) = \begin{cases}1 & m=n\\ 0 & m\neq n\end{cases}.$$

The optimum way of constructing the transmit basis is to use the orthonormal eigenfunctions of the channel autocorrelation function, namely given r(t)=h(t)* h*(-t), the eigenfunctions $\{\varphi_n(t)\}$, which is non-zero over the symbol period $[-(T - T_H)/2, (T - T_H)/2)$, would then satisfy the relation

$$\rho_n * \varphi_n(t) = \int_{-T/2}^{T/2} r(t-\tau)\cdot\varphi_n(\tau)\,d\tau, \quad n = 1,...,\infty,\ \forall\, t \in [-(T-T_H)/2, (T-T_H)/2),$$

where it can be seen that the convolution of $\varphi_n(t)$ and the matched-filtered channel over the symbol period reproduces the eigenfunction scaled with the constant ρ_n. The eigenfunctions $\varphi_n(t)$ can be interpreted as the different modes of the channels that allow independent data transmission with the gain ρ_n [10].

However, in practice, the use of channel eigenfunctions as the basis functions is infeasible due to the raised hardware computational complexity of the various different channel eigenfunctions, and the fact that the practical system could require infinite delay. Therefore a suboptimal partitioning method, the Discrete Fourier Transform (DFT) is applied [10].

The major advantage of using a DFT lies in the efficient computational methods such as the Fast Fourier Transform (FFT), where N-point FFT can be computed in $N\log_2 N$ operations instead of N^2. The other benefit comes with the circular convolution in DFT (meaning a product in the DFT domain), where the channel partitioning forces the transmit symbol to have $x_{-k} = x_{N-k}$ for $k = 1,...\nu$, and the last repeating ν samples at the beginning of each symbol is the cyclic prefix (which will be discussed in Section **3.4**). The matrix description of the channel is then **y**=**Hx**+**n**, where **H** is the square $N \times N$ circulant matrix, **x**, **y**, and **n** are the transmitted symbols, received symbols, and channel noise, respectively in each sub-channel, in column vector form. The matrix description can be visualized as follows [10]

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \end{bmatrix} = \begin{bmatrix} h_0 & 0 & \cdots & 0 & h_\nu & \cdots & h_1 \\ h_1 & h_0 & 0 & \ddots & 0 & h_\nu & \ddots \\ \ddots & h_1 & h_0 & 0 & \ddots & 0 & h_\nu \\ h_\nu & \ddots & h_1 & h_0 & 0 & \ddots & 0 \\ 0 & h_\nu & \ddots & h_1 & h_0 & 0 & \ddots \\ \ddots & 0 & h_\nu & \ddots & h_1 & h_0 & 0 \\ 0 & \ddots & 0 & h_\nu & \ddots & h_1 & h_0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix} + \begin{bmatrix} n_0 \\ n_1 \\ \vdots \\ n_{N-1} \end{bmatrix}.$$

For eigenvector coding in transmission, the restriction of non-negative real singular values is no longer necessary, thus the singular value decomposition (SVD) can be simplified to an eigendecomposition of the circulant matrix **H** [10],

$$\mathbf{H} = \mathbf{M}\Lambda\mathbf{M}^*, \tag{1}$$

where Λ is the $N \times N$ square diagonal eigenvalue matrix, and $\mathbf{M}$ is the $N \times N$ single unitary vector coding matrix with column eigenvectors ($\mathbf{M}\mathbf{M}^* = \mathbf{M}^*\mathbf{M} = \mathbf{I}$), for both the transmitter and the receiver. In the case of DMT, the vector coding matrix $\mathbf{M} = \mathbf{F}^*$, with

$$\mathbf{F}^* = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{j\frac{2\pi}{N}} & e^{j\frac{2\pi}{N}2} & \cdots & e^{j\frac{2\pi}{N}(N-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & e^{j\frac{2\pi}{N}(N-2)} & e^{j\frac{2\pi}{N}(2)(N-2)} & \cdots & e^{j\frac{2\pi}{N}(N-1)(N-2)} \\ 1 & e^{j\frac{2\pi}{N}(N-1)} & e^{j\frac{2\pi}{N}(2)(N-1)} & \cdots & e^{j\frac{2\pi}{N}(N-1)(N-1)} \end{bmatrix},$$

such that

$$\begin{aligned} &\mathbf{y} = \mathbf{F}^* \Lambda \mathbf{F} x + \mathbf{n}, \\ &\mathbf{x} = \mathbf{F}^* \mathbf{X} \text{ (Transmitter IDFT)}, \\ &\mathbf{Y} = \mathbf{F}\mathbf{y} \text{ (Receiver DFT)}. \end{aligned}$$

The DFT of an N-dimensional sequence $\mathbf{x}$ [5], where

$$\mathbf{x} \triangleq \begin{bmatrix} x_{N-1} \\ x_{N-2} \\ \vdots \\ x_0 \end{bmatrix},$$

is given by

$$\mathbf{X} \triangleq \begin{bmatrix} X_{N-1} \\ X_{N-2} \\ \vdots \\ X_0 \end{bmatrix},$$

where

$$X_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \cdot e^{-j(2\pi/N)kn} \ \forall k \in [0, N-1],$$

and the IDFT of $\mathbf{X}$ is then given by

$$x_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k \cdot e^{j(2\pi/N)kn} \ \forall n \in [0, N-1].$$

The IDFT can be rewritten as $\mathbf{x} = \mathbf{F}^*\mathbf{X}$, where $\mathbf{F}^*$ is the vector coding matrix $\mathbf{M}$ as mentioned in (1). In baseband transmission, the input sequence $\mathbf{X}$ has to obey Hermitian symmetric for the input signal $\mathbf{x}$ to be real, which in practical terms would mean the FFT size is essentially the same as the DFT, but with an interval symmetry [5].

3.4 Bit loading

In multi-carrier DSL modems, bit loading is an essential initialization step to determine the number of bits allocated to each sub-channel. Thus, it is an optimization problem of maximizing the achievable bits $b = \sum_n b_n$ over b_n and energy ε_n, to eventually maximize the data rate, $R = b/T$, for a fixed symbol rate $1/T$ in a set of parallel sub-channels [10]

$$b = \frac{1}{2}\sum_{n=1}^{N} \log\Big(1 + \frac{\varepsilon_n \cdot g_n}{\Gamma}\Big), \tag{2}$$

where g_n is the sub-channel SNR when unit energy is applied by the transmitter, and Γ is the SNR Shannon gap approximation,

$$\Gamma = \frac{1}{3}\left[\frac{\text{Symbol Error Rate (SER)}}{4Q}\right]^2, \quad Q(x) = \int_x^\infty \frac{e^{-u^2/2}}{\sqrt{2\pi}}\,du.$$

It can be seen in (2), g_n is fixed whereas ε_n can be adjusted to maximize b, subject to the energy constraint

$$\sum_{n=1}^{N} \varepsilon_n = N\,\overline{\varepsilon}_x\,. \tag{3}$$

The fundamental method is the *water filling optimization* where a Lagrange multiplier is used to maximize (2) constrained by (3), and the cost function becomes [10]

$$\frac{1}{2\ln 2}\sum_n \ln\Big(1 + \frac{\varepsilon_n \cdot g_n}{\Gamma}\Big) + \lambda(\sum_n \varepsilon_n - N\overline{\varepsilon}_x).$$

Now taking the derivative with respect to ε_n and setting it to 0 yields

$$\begin{aligned}
\frac{\partial}{\partial \varepsilon_n}\frac{1}{2\ln 2}\sum_n \ln\Big(1 + \frac{\varepsilon_n \cdot g_n}{\Gamma}\Big) &= -\lambda \frac{\partial}{\partial \varepsilon_n}(\sum_n \varepsilon_n - N\overline{\varepsilon}_x) \\
\frac{1}{2\ln 2}\,\frac{1}{\frac{\Gamma}{g_n} + \varepsilon_n} &= -\lambda \\
\varepsilon_n + \frac{\Gamma}{g_n} &= \text{constant} \\
\varepsilon_n + \Gamma \cdot \frac{\sigma_n^2}{|H_n|^2} &= \text{constant (for Multi-tone).}
\end{aligned}$$

Thus, it can be observed that given a constant Γ, $\sum_n \varepsilon_n = N\overline{\varepsilon}_x$ and $\overline{\varepsilon}_x \geq 0$, the solution can be visualized as a curve of the inverted channel SNR versus the number of sub-channels, i.e., Γ/g_n versus n, being filled with energy (water), i.e., $\Gamma/g_n + \varepsilon_n$, to a constant level [5] [10].

However, for a large N the water filling method may produce very small or fractional b_n parts, which is computationally prohibitive for the hardware encoders and decoders. Various alternative loading criteria are being explored

to sub-optimally enhance bits b_n and energy ε_n distribution. In general, bit-loading optimization criteria can be summarized into two categories : the Rate-Adaptive(RA) loading, and the Margin-Adaptive(MA) loading [10].

For the RA loading, the number of bits per symbol is maximized, subject to a fixed energy constraint:

$$\begin{aligned} \min b &= \sum_{n=1}^{N} \frac{1}{2} \log_2 (1 + \frac{\varepsilon_n \cdot g_n}{\Gamma}) \\ \text{subject to: } N\overline{\varepsilon}_x &= \sum_{n=1}^{N} \varepsilon_n \,, \end{aligned}$$

where target noise margin and specified BER can also be added to the constraints [10].

For the MA loading method, the energy is minimized given fixed bits/symbol constraint:

$$\begin{aligned} \min_{\varepsilon_n} \varepsilon_x &= \sum_{n=1}^{N} \varepsilon_n \\ \text{subject to: } b &= \sum_{n=1}^{N} \frac{1}{2} \log_2 (1 + \frac{\varepsilon_n \cdot g_n}{\Gamma}) \,, \end{aligned}$$

thus the maximum margin is [10]

$$\gamma_{max} = \frac{N\overline{\varepsilon}_x}{\varepsilon} \,.$$

Depending on the needs of the system, different criteria can be applied, i.e., the MA loading method is favorable for power saving transmission designs, whereas the RA loading scheme suits high capacity demands of the system. Aside from either maximizing b_n or minimizing ε_n, new methods in lowering the aggregated BER using a quasi-power optimal algorithm [11], or jointly optimizing both b_n and ε_n [12] through a weighting coefficient α are developed. There are various other loading algorithms with reduced computational complexity [4].

3.5 Cyclic Prefix

In the DMT transmitter, the serial-to-parallel converted bits are mapped to modulation symbols using QAM. The linear convolution of the transmitted signals with the channel impulse response (CIR) would then smear out the symbols: let us assume an LTI channel of length N, thus the input sequence $s(n)$, $n = n_0, ..., n_0+N-1$ and the channel impulse response $h(n)$ give the output sequence [6]

$$\begin{aligned} r(n) &= h(n) * s(n) \\ &= \sum_k h(k)s(n-k) \\ &= \sum_k s(k)h(n-k), \; n = n_0, ..., n_0 + N + M - 1, M = \lceil \tau_m ax / T_S \rceil \end{aligned}$$

with a resulting length of $N+M$, where M is the dispersion of the channel. Channel dispersion would then cause symbols to overlap and smear out, or the so-called inter-symbol interference (ISI) would occur. Taking into account that there are various parallel sub-channels and each operates on its own carrier frequency in a DMT system, inter-carrier interference (ICI) may be caused by ISI, or arise from the ideal sampling points [6].

Both ISI and ICI can be eliminated using a sufficient Cyclic Prefix (CP). Given N symbols in the IFFT block, the last v samples, $v < N$, are appended to the beginning of the original symbols. Assume a transmitted DMT symbol $\mathbf{X}$, IFFT size of 4, as a simple illustration [13]

$$\begin{bmatrix} x_1 & x_2 & x_3 & x_0 & x_1 & x_2 & x_3 \end{bmatrix},$$

where the size of the CP is 3. The DMT symbol is now consisting of 7 modulation symbols with a CP of 75 % of the payload size, which is, of course, inefficient, however is chosen only for illustration purposes [13].

Now suppose two DMT symbols $\mathbf{X^{(1)}}$ consisting of elements y_0, y_1, y_2, y_3, and $\mathbf{X^{(2)}}$, consisting of elements x_0, x_1, x_2, x_3 are transmitted. The appended symbol with the CP is now

$$[y_1 \;\; y_2 \;\; y_3 \;\; y_0 \;\; y_1 \;\; y_2 \;\; y_3 \;\; x_1 \;\; x_2 \;\; x_3 \;\; x_0 \;\; x_1 \;\; x_2 \;\; x_3],$$

and the channel impulse response is $h = [h_0 \; h_1]$, the received signal through the defined linear convolution

$$r[n] = \sum_{k=0}^{1} h[k]x[n-k] = \sum_{k=0}^{1} x[k]h[n-k]$$

would then be $[h_0 \; h_1]$ reversed and swept through $[y_3 \;\; x_1 \;\; x_2 \;\; x_3 \;\; x_0 \;\; x_1 \;\; x_2 \;\; x_3]$, since the symbols before y_3 has no effect on the received symbol

$$\begin{aligned}
r[0] &= h_1 y_3 + h_0 x_1, \\
r[1] &= h_1 x_1 + h_0 x_2, \\
r[2] &= h_1 x_2 + h_0 x_3, \\
r[3] &= h_1 x_3 + h_0 x_0, \\
r[4] &= h_1 x_0 + h_0 x_1, \\
r[5] &= h_1 x_1 + h_0 x_2, \\
r[6] &= h_1 x_2 + h_0 x_3.
\end{aligned}$$

Throwing away the CP part of the received DMT symbols, i.e., $r[0]$,$r[1]$, and $r[2]$, it can be found that the rest are the received symbol $\mathbf{X}$, i.e., $r[3]$,$r[4]$, $r[5]$, and $r[6]$. Now, the circular convolution can be computed as

$$r_c[n] = \sum_{k=0}^{3} h[k]x[(n-k) \bmod 4],$$

and the results show that $r_c[0] = r[3]$, $r_c[1] = r[4]$, $r_c[2] = r[5]$, $r_c[3] = r[6]$ [13].

Therefore, it can be seen that adding a cyclic prefix is equivalent to transforming a linear convolution to an ideal circular convolution, avoiding ISI. Since a circular convolution in discrete-time domain means a point-by-point multiplication in frequency domain, the ICI can also be eliminated in the DFT-domain block by doing a point-by-point division for reconstructing the transmitted symbols [13].

3.6 Peak-to-Average Ratio Reduction

With the peak corresponding to transmitting a sub-symbol farthest from the origin of the constellation diagram [10], the Peak-to-Average Ratio is defined as

$$20\log\frac{V_{peak}}{V_{RMS}}\,\text{dB} = 10\log_{10}\frac{P_{peak}}{P_{average}}\,\text{dB},$$

where V_{peak} is the peak voltage of the time-domain waveform and V_{RMS} is the Root Mean Square (RMS) of the waveform, and the discrete time domain signal x_k is

$$x_k = \sum_{n=0}^{N-1} X_n \cdot e^{-j\frac{2\pi}{N}nk},$$

of which the sum of a large number of random variables leads to a Gaussian distribution [10]. Thus, the theoretical PAR can be derived as

$$PAR_T = 20\log_{10}(N)\,\text{dB}.$$

It can be seen that for a system with a large N the theoretical PAR is also large, which requires a wider linearity range of the analog components such as filters, amplifiers, and A/D, D/A converters of the modem. Such hardware adjustments can be costly, and non-linear distortion caused by "clipping" when signal amplitude exceeds the maximum would not be efficiently reduced [5].

Tone reservation is one of the most used PAR reduction methods that efficiently addresses the above problems. By adding a peak annihilator $\mathbf{c}$ to the transmit signal $\mathbf{x}$ on each symbol, the peak annihilator adds the negative of the highest peak so as to reduce the amplitude at the instant of the peak. $\mathbf{c}$ is constructed by frequency domain inputs on certain tones in a pre-determined set $I = \{i_1, i_2, ...i_r, ..., N-i_r, ..., N-i_1\}$, typically with $r << N/2$, and $\mathbf{x}+\mathbf{c}$ is the cyclically extended and transmitted symbol, thus $\mathbf{x}$+$\mathbf{c} = \mathbf{Q}^*[\mathbf{X}$+$\mathbf{C}]$, where $\mathbf{C} = [C_0, ..., C_{N-1}]^T$ and $\mathbf{Q}^*$ is the conjugate transpose of the IFFT matrix [14], and C_n is selected when $n \in I$. Therefore the optimization is to compute the vector $\mathbf{c}$ to minimize the maximum peak value, i.e., $\min_{\mathbf{C}} \max x_k|\mathbf{x} + \mathbf{Q}^*\mathbf{C}|$. [10]

The other commonly used PAR reduction method is tone injection, where signals are added to the data symbols already present on certain tones, transparent to the receiver decision process, rather than reserving tones for peak-reduction signals [10]. The challenge now lies in the trade off between reduced

clipping and increased quantization noise brought by the quantization step in the A/D-D/A converters. Recent development on PAR reduction methods have introduced Selected Mapping, Partial Transmit Sequences [4], etc., whereas tone reservation and tone injection are commonly used thanks to their low complexity.

3.7 Frequency Domain Equalization and Time Domain Equalization

Due to the robust performance of DMT systems in frequency selective channels and the avoidance of ISI, only two simple equalizers are implemented in the modem to mitigate transmission-channel impairments and shorten the channel impulse response, namely and respectively, the Frequency-domain Equalizer (FEQ) and Time-domain Equalizer (TEQ).

In the FEQ, each DMT sub-channel is equipped with a Zero Forcing Equalizer (ZFE) to compensate for the sub-channel constellation points scaling and rotation due to the channel gains [5]. Equipped with both matched filtering and equalization, the complex product of equalizer setting $\mathbf{W_n}$ and the received signal $\mathbf{Y_n}$, the equalized signal for the n^{th} sub-channel is $\hat{\mathbf{X}}_\mathbf{n}$=$\mathbf{Y_n}\mathbf{W_n}$, where $\mathbf{W_n}$ = ${\mathbf{P_n}}^{-1}$, and $\mathbf{P_n}$ is the matrix for combined impulse-responses and matched filtering [7] [10].

The TEQ, on the other hand, shortens the impulse response. Given $\nu+1$ (ν is the cyclic prefix) sampling periods of the channel impulse response, a required excess-bandwidth ν/N is introduced. For large ν, the excess-bandwidth ν/N may be too large, thus N can be increased to compensate. However, increasing N may increase hardware complexity and introduce latency that complicates frame synchronization [10]. Thus the reduction of the cyclic prefix ν by the TEQ would efficiently address the problem while maintaining relatively lower complexity.

One of the major TEQ designs is the training method, where a channel estimate or additional overhead is required. Originated from making the impulse response shorter than ν in a TEQ [15], later derivations in maximize the channel capacity, or the geometric mean of the SNR, while maintaining a short impulse response was first introduced in [16] [17]. Throughout the evolution of TEQ algorithms, the most fundamental and essential ones are the Minimum Mean-Square Error (MMSE) TEQ, the maximum shortening SNR [18], and the per-tone equalization [19]. The other newly emerged major TEQ design, the blind method, where the redundancy and the orthogonality between adjacent sub-carriers are exploited to shorten the effective channel [20].

A common formulation of a TEQ design can be summarized as a generalized Rayleigh quotient, or product of generalized Rayleigh quotients maximization [19], which is the following optimization problem

$$\hat{\mathbf{w}}^{opt} = \arg\max_{\hat{w}} \prod_{j=1}^{M} \frac{\hat{\mathbf{w}}^T \mathbf{B_j} \hat{\mathbf{w}}}{\hat{\mathbf{w}}^T \mathbf{A_j} \hat{\mathbf{w}}},$$

where the vector $\mathbf{w}$ contains the TEQ, thus the single quotient solution, i.e., $M = 1$,

$$\hat{\mathbf{w}}^{opt} = \arg\max_{\hat{w}} \frac{\hat{\mathbf{w}}^T \mathbf{B_j} \hat{\mathbf{w}}}{\hat{\mathbf{w}}^T \mathbf{A_j} \hat{\mathbf{w}}} ,$$

is the generalized eigenvector of the matrix pair $(\mathbf{B}, \mathbf{A})$, corresponding to the largest generalized eigenvalue λ, and the computation of $\hat{\mathbf{w}}$ is solving $\mathbf{B}\hat{\mathbf{w}} = \lambda \mathbf{A}\hat{\mathbf{w}}$ [19]

The TEQ design techniques differ in the number of the Rayleigh quotients M and the matrices $\mathbf{A}$ and $\mathbf{B}$. In the case of $M = 1$, $\mathbf{A}$ and $\mathbf{B}$ are determined by the channel characteristics, aiming at modelling the sub-channel SNR and the different constraints. Now let's take one constraint for illustration, Chow and Cioffi's MMSE method for shortening the impulse response of the MC system, which has a unit-norm constraint [19] [15]. Given a transmitted sequence $\mathrm{x}^{\mathrm{l}} = [x(l), ..., x(l-v)]^T$, and a received sequence $\mathrm{y}^{\mathrm{l}} = [y(l), ..., y(l-L_w)]^T$ with sample index l, the transmitted sequence $x(l)$ is convolved with the channel impulse response $\mathbf{h}$, and equalized by the TEQ $\mathbf{w}$. The output $\mathbf{y}$ is then delayed by Δ, and compared with the ideal output that filtered $\mathbf{x}$ with the target impulse response $\mathbf{b} = \mathbf{v} + \mathbf{1}$. The difference sequence $e(l)$ is finally minimized in an MMSE [19] [21] [22], resulting in

$$\mathbf{A} = \mathbf{R_y} - \mathbf{R_{yx}} \mathbf{R_x^{-1}} \mathbf{R_{xy}} ,$$
$$\mathbf{B} = \mathbf{I}_{L_w+1} ,$$

where

$$\mathbf{R_x} = E\{\mathrm{x}^{\mathrm{l}} (\mathrm{x}^{\mathrm{l}})^T\} ,$$
$$\mathbf{R_y} = E\{\mathrm{y}^{\mathrm{l}} (\mathrm{y}^{\mathrm{l}})^T\} ,$$
$$\mathbf{R_{yx}}(\Delta) = E\{\mathrm{y}^{\mathrm{l}} (\mathrm{x}^{\mathrm{l}-\Delta})^T\} .$$

For multiple quotient problems, i.e., when $M > 1$, the problem becomes much more complicated in applying gradient-descend strategies finding only a local optimum. Non-linear optimization can also be applied, however such optimization can only be solved numerically and approximated, which is undesirable for real-time implementations, and significantly increases the computational complexity [19] [17].

3.8 Channel and Noise Estimation

Channel estimation and identification is performed to determine the values of FEQ taps, and sub-channel SNRs for bit allocation. A training signal of M symbols, both known to the transmitter and the receiver, is transmitted repeatedly

over the channel, and the subsequent received symbols are averaged to estimate the channel gain [5]

$$\bar{\mathbf{y}} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{y_m} \, ,$$

where $\bar{\mathbf{y}}$ and $\mathbf{y}$ are vectors of time-domain symbols samples, m is the symbol index, and the first received symbol is discarded for its use of clearing the channel. Then each sub-channel frequency response $\widehat{\mathbf{H}}_\mathbf{n}$ is estimated by dividing the averages of the DFT-domain signal $\mathbf{Y_{m,n}}$ and by each other [5]

$$\widehat{\mathbf{H}}_\mathbf{n} = \sum_{n=1}^{M} \frac{\mathbf{Y_{m,n}}}{\mathbf{X_{m,n}}} \, ,$$

Similar to a frequency response estimation, the noise is estimated by the residual error $\mathbf{E_{m,n}}$ on sub-channel n, assuming zero-mean noise $\mathbf{E_{m,n}} = \mathbf{Y_{m,n}} - \mathbf{X_{m,n}}\widehat{\mathbf{H}}_\mathbf{n}$. The noise variance on the n^{th} sub-channel over the $M+1$ training symbols is then estimated as [5]

$$\hat{\sigma_n^2} = \frac{1}{M} \sum_{m=2}^{M+1} |\mathbf{E_{m,n}}|^2 \, .$$

From both channel and noise identification and estimation, it can be seen that a larger number of symbols M indicates better SNR estimation accuracy [5].

4 Frame Synchronization

Synchronization is one of the major aspects for realizing the DMT system. There are basically three kinds of offsets that need to be synchronized, namely the time offset, frequency offset, and sampling frequency offset.

The time offset is mainly caused by the channel delay, it is crucial to the frame synchronization because otherwise the essential sub-channel orthogonality of the DMT system would be violated, when the incoming symbol frame does not align properly with the FFT window. The frequency offset is caused by the frequency mismatch of sub-carriers in the transceiver, and the sampling frequency offset is caused by the frequency/phase difference of the local oscillators between the transmitter and the receiver. In the scope of this Guided Research, the frame synchronization is of the primary concern.

4.1 ML Estimation Method

One major type of the time offset synchronization would be the blind synchronization. The basic idea is to deploy the cyclic correlation, i.e., correlation of the CP, in the information symbol, instead of using the training symbol.

Van de Beek et al. proposed a method of ML (Maximum Likelihood) estimation [1]. Suppose the transmitted signal $s(k)$ after the IFFT is a linear combination of i.i.d (independent and identically distributed) random variables. By the central limit theorem, $s(k)$ can be characterized by a complex Gaussian process given the number of the sub-carriers in the OFDM system is sufficiently large. The process, however, is not white due to the fact that the CP would have a correlation between the pairs of samples that are spaced N samples apart. Therefore, this probabilistic structure can be used for the received signal $r(k)$, which contains information about the time offset θ and carrier frequency offset ϵ. Typically in the proposed linear-bus structured DMT system, $|\epsilon| < \pi$, which is not concerned in the scope of this Guided Research, since the focus is meanly on the time offset synchronization.

Assume that our observation frame in $r(k)$ consists of $2N+L$ samples, and the channel delay θ is unknown to the receiver. Then the index sets can be defined as [1]

$$I \triangleq \{\theta, \cdots, \theta + L - 1\} \quad \text{and} \quad I^{'} \triangleq \{\theta + N, \cdots, \theta + N + L - 1\}.$$

Now the observed samples are collected in $\mathbf{r} \triangleq [r(1) \cdots r(2N+L)]^T$, and then it can observed that the pairwise correlation for the samples in the CP and their copies $r(k), k \in I \cup I^{'}$, i.e.,

$$\forall k \in I : E[r(k)r^*(k+m)] = \begin{cases} \sigma_s^2 + \sigma_n^2, & m = 0 \\ \sigma_s^2 e^{-j2\pi\epsilon}, & m = N \\ 0, & \text{otherwise} \end{cases}$$

where the remaining samples $r(k), k \notin I \in I^{'}$ are mutually uncorrelated.

The observed $2N+L$ samples can then be characterized by the probability density function $f(\mathbf{r}|\theta)$, the log-likelihood function for θ is then [1]

$$\Lambda(\theta) = \log f(\mathbf{r}|\theta) = \log\left(\prod_{k\in I} f(r(k), r(k+N)|\theta) \prod_{k\notin I\cup I^{'}} f(r(k)|\theta)\right), \tag{4}$$

where in (4) $\mathbf{r}$ is a jointly Gaussian vector, and after some manipulations [1]

$$\Lambda(\theta) = |\gamma(\theta)| - \rho\Phi(\theta), \tag{5}$$

where $\gamma(\theta)$ can be denoted as $\gamma(d)$ (d as the time index)

$$\gamma(d) \triangleq \sum_{k=d}^{d+L-1} r(k)r^*(k+N), \tag{6}$$

$\Phi(\theta)$ in (4) can be represented as $\Phi(d)$

$$\Phi(d) \triangleq \frac{1}{2}\sum_{k=d}^{d+L-1} |r(k)|^2 + |r(k+N)|^2, \tag{7}$$

and ρ in (4) is the magnitude of the correlation coefficient

$$\rho = \frac{SNR}{SNR+1}. \tag{8}$$

Finally the maximum likelihood estimation of the delay would be [1]

$$\hat{\theta}_{ML} = \arg\max_{\theta}\{|\gamma(\theta)| - \rho\Phi(\theta)\}. \tag{9}$$

This method can also be interpreted as constructing a timing metric by (5), (6), and (7). The received samples are fed through a L-sample-long sliding window, which is a correlator lagged by N samples. The sum of the products in (6) is then subtracted by the energy term in (7) scaled with ρ. When the window is perfectly aligned with the start of the frame, a peak will be produced.

4.2 Evaluation of the ML Estimation Method

The ML estimator is simulated with two information symbols each of the size $N = 2048$, with the length of the CP as $L = N \cdot 25\% = 512$, under the flat channel without noise, as well as under AWGN channel with different SNRs. The plot is shown in Fig. 3 .

In Fig. 3 it can be seen that under the ideal condition the peak occurs exactly at $d = 0$, which is expected since there's no channel delay. The other three simulations under AWGN channel peak approximately at $d = 0$, each with an approximate error range within +5 when running the simulation various times. This suggests that when noise is present in the channel, one single estimation

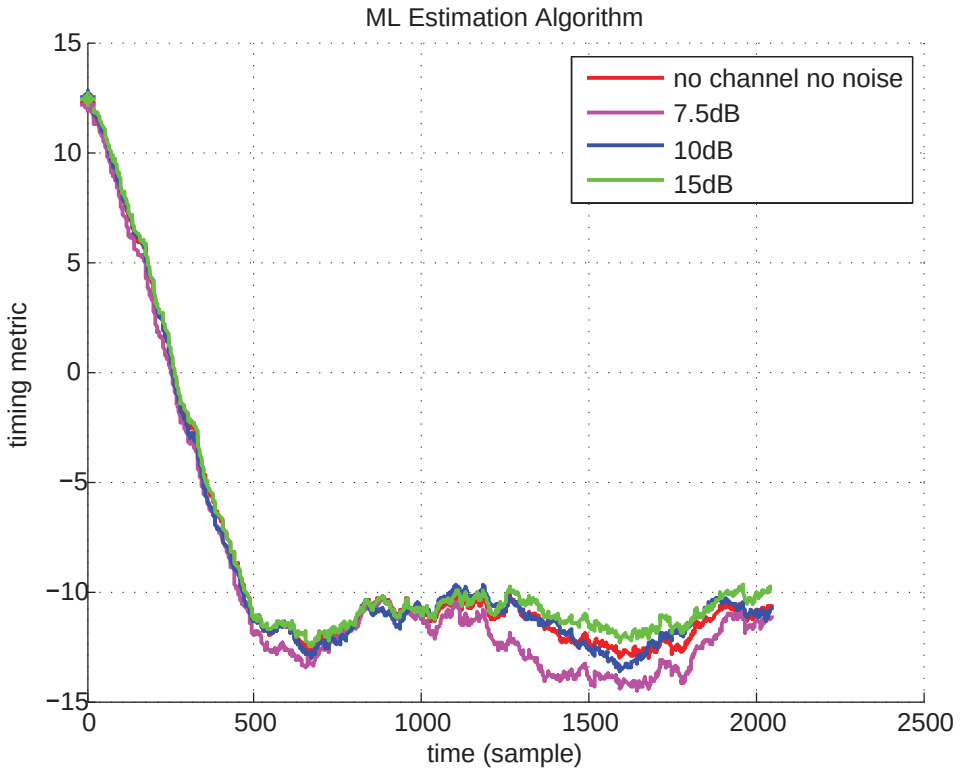

Figure 3: ML estimation method under ideal condition and AWGN channel

using the blind method is not robust enough since the cyclic correlation is not perfect any more. In real application the cyclic correlation is typically done for a overhead of hundreds of symbols, and then compute the average of the ML estimations of the time offset. It can also be observed that the peaks of the timing metric slightly decrease as the SNR goes down, as shown in the plot at 0 of the time (sample) axis.

4.3 Data-aided Synchronization

The other major type of the frame synchronization is the data-aided method. Given that the channel is not known to the receiver, the basic idea is to construct a certain structured preamble training symbol in the time domain, appended to the information symbol, such that when a sliding window at the receiver is aligned with the start of the frame, a peak will be detected in the constructed timing metric.

4.3.1 Schmidl and Cox Method

The Schmidl and Cox method relies on the training sequence of two identical halves in one time domain symbol [2]. The idea is to build a timing estimator by searching for the correlation peak of the identical halves. As long as the cyclic prefix is longer than the channel impulse response, the training symbol will preserve the identical halves through the channel, with phase shift caused by the carrier frequency offset.

The training symbol in the time domain can be generated by transmitting a structured sequence in the frequency domain. The sequence is formed by zero forcing the odd frequencies, and placing a pseudo-noise (PN) sequence on the even frequencies, which means that one of the points of a QPSK constellation is transmitted at each even frequency. The PN sequence used for the evaluation simulation is done using 64-QAM constellation look up table [2]

$$\text{QAMTable} = [7+7i, -7+7i, -7-7i, 7-7i]\,,$$

where the constellation points are randomly chosen from the look up table and placed at even frequencies. The rationale behind generating this training symbol can by explained by the duality property of the Fourier transform [23]

$$\text{if } x(t) \Longleftrightarrow X(w), \quad \text{then } X(t) \Longleftrightarrow 2\pi x(-w)\,. \tag{10}$$

It is known that periodicity in the frequency domain can be achieved by zero insertion in the time domain as part of digital interpolation. Given a band-limited discrete-time signal $x(m)$, and its base-band spectrum $X(f)$, the repetition of I times of the original spectrum can be achieved by inserting $I-1$ zeros between each pair of samples, forming the zero-inserted signal $x_z(m)$, which can be expressed as [23]

$$x_z(m) = \begin{cases} x\left(\frac{m}{I}\right), & m = 0, \pm I, \pm 2I, \cdots \\ 0, & \text{otherwise} \end{cases}$$

The spectrum of the zero-inserted spectrum is then

$$\begin{aligned} X_z(f) &= \sum_{m=-\infty}^{\infty} x_z(m) e^{-j\pi f m} \\ &= \sum_{m=-\infty}^{\infty} x(m) e^{-j2\pi f m I} \\ &= X(If) \end{aligned}$$

using the duality in (10), it can observed that

$$\text{if } x_z(m) \Longleftrightarrow X(If), \quad \text{then } X(It) \Longleftrightarrow x(-If).$$

Now when zeroing the odd frequencies, the original PN sequence can be considered as being inserted with zeros by a factor of two, i.e., $I = 2$. Thus after the IFFT, the resulting symbol would consists of two periodic half-symbols in the time domain.

In order to get the timing acquisition, the particular timing metric is constructed as follows [2]

$$P(d) = \sum_{m=0}^{L-1} (r_{d+m}^* r_{d+m+l}), \tag{11}$$

$$R(d) = \sum_{m=0}^{L-1} |r_{d+m+L}|^2, \tag{12}$$

$$M(d) = \frac{|P(d)|^2}{(R(d))^2}, \tag{13}$$

where r_m denotes the received samples, L is the number of complex samples in one-half of the training symbol (excluding the CP), i.e., $L = \frac{N}{2}$, and d is the time index that corresponds to the first sample in a window of $2L$ samples. Thus the timing metric can be seen as feeding the time-domain samples to an N sample long sliding window correlator of lag $N/2$, i.e., $P(d)$ in (11) (the cross-correlation between the two halves of the window), and a peak will be exhibited when the sliding window is exactly aligned with the start of the frame. The peak is then scaled to 1 in flat channel free of noise by the auto-correlation for the second half-symbol as in (13). As the window keeps sliding, the peak would stay at the same level because the cyclic prefix at the beginning of the frame is exactly the same as the last 50% of the samples of the second half training symbol (given that the CP is 25% of the FFT size), such that the windowed frame would have identical halves as long as the timing index is moving within the CP. Now the timing offset estimator would be [2]

$$\hat{d} = \underset{d}{\text{argmax}}(M(d)).$$

4.3.2 Evaluation of the Schmidl and Cox Method

Now in the evaluation simulation, the FFT size is $N = 2048$, the CP size $L = N/4 = 512$, and the frame size $N_s = N + L = 2560$. The transmitted time-domain symbols, appended with the CP, are one symbol filled with zeros, meaning there's no transmission, the training symbol, and an information symbol, respectively. The information symbol is generated from the same QAM table but without zero insertion at odd frequencies. The algorithm is then simulated under the ideal condition where there is no channel and no noise, and also simulated in AWGN channel with different SNR (signal to noise ratio). The plot is demonstrated in Fig. 4 .

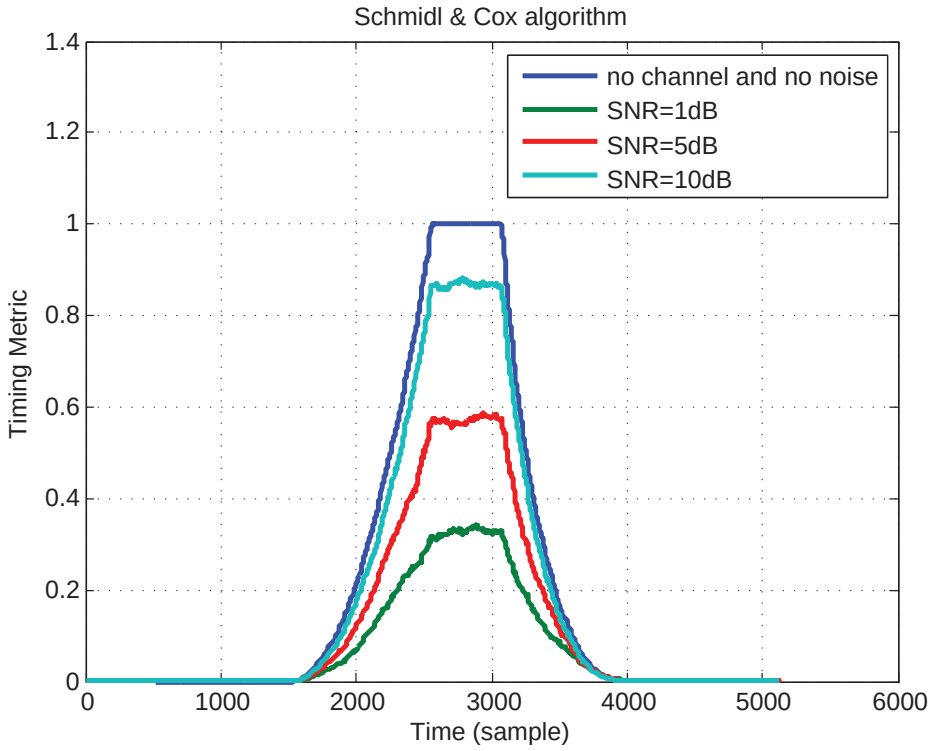

Figure 4: S & C method simulation under ideal condition and AWGN channel

It can be seen from Fig. 4, when simulated without channel or noise, there is a plateau starting from $d = 2561$, which is exactly the start of the training symbol after the zero symbol of size $N_s = 2560$. The plateau ends at $d = 3073$, and thus have a total length of the CP, i.e., $Ng = 512$. When simulated under AWGN channel, both the length and height of the plateau shrinks as the SNR goes down. This is due to the added channel and noise, the length of the plateau would now be the difference between the length of the CP and the length of the CIR (channel impulse response), and the scaled peak is now less than 1.

The Schmidl and Cox method for timing offset synchronization exhibit some uncertainty due to the plateau. However, as long as the length of the CP is longer than the length of the CIR, the signal itself will not be affected by the symbol timing errors since the timing estimate is in the remaining CP of the plateau. For a more robust estimate, $\hat{d}$ can be found by averaging 90% of the peak to the left and right, instead of only finding the maximum.

4.3.3 Park et al. Method

Since the plateau in Schmidl & Cox algorithm would raise uncertainty, Park et al. proposed another method to enlarge the difference among peaks such that the highest peak would indicate the timing offset [3]. Based on the S & C algorithm, the Park algorithm uses a modified training symbol and timing metric. The preamble in the time domain is set as

$$P_{Park} = [A_{N/4} \quad B_{N/4} \quad A^*_{N/4} \quad B^*_{N/4}]$$

where $A_{N/4}$ denotes the samples of length $N/4$ generated from the IFFT of a PN sequence, $B_{N/4}$ represents the symmetry time domain sequence, then $A^*_{N/4}$, and $B^*_{N/4}$ denote the complex conjugate of $A_{N/4}$ and $B_{N/4}$ respectively. Such a time domain sequence can be constructed in the same way as in the S & C algorithm by zeroing out the odd frequencies except for the real PN sequence in the frequency domain instead of the complex PN sequence. Due to the periodicity and Hermitian symmetry of the discrete Fourier transform, the IFFT of such a real PN sequence would produce a complex time domain symbol with complex conjugate symmetry in two halves and mirrored symmetry in each half.

The new timing metric is then [3]

$$P(d) = \sum_{k=0}^{N/2} r(d-k) \cdot r(d+k)\,, \tag{14}$$

$$R(d) = \sum_{k=0}^{N/2} |r(d+k)|^2\,, \tag{15}$$

$$M = \frac{|P(d)|^2}{(R(d))^2}\,. \tag{16}$$

In the S & C algorithm, two adjacent time index d would have the sum of $N/2$ pairs of products, where the pairs of products are almost the same except for only one increment in d. Nevertheless, in the Park algorithm, the $N/2+1$ pairs of products between adjacent time index d are quite different due to the preamble construction. Therefore the difference between the peak values is sufficiently enlarged by the Park algorithm so that the plateau is no longer present. Now instead of having a sliding window correlator of lag $N/2$, the timing metric (16) in the Park method feeds the received time domain samples to a sliding window of the same length N, but with the different products starting with $N/2-1$

and $N/2+1$ until both ends of the window as k increases, i.e., $P(d)$ in (14). The window moves as the time index d increases, and two peaks are expected when the window is perfectly aligned with the start of the frame and the start of the symbol. The sums of the products are then scaled by (15) such that the peak will be 1 under the ideal condition where there's no channel or noise [3].

4.3.4 Evaluation of the Park Method

The Park method is then simulated in flat channel free of noise, and the AWGN channel with different SNRs, respectively. The FFT size and the CP size are the same as in the simulation of the S & C algorithm. The transmitted time-domain symbols, appended with CP, are one symbol filled with zeros, meaning there's no transmission, the park training symbol, and an information symbol. The simulation results are shown as the in Fig. 5 .

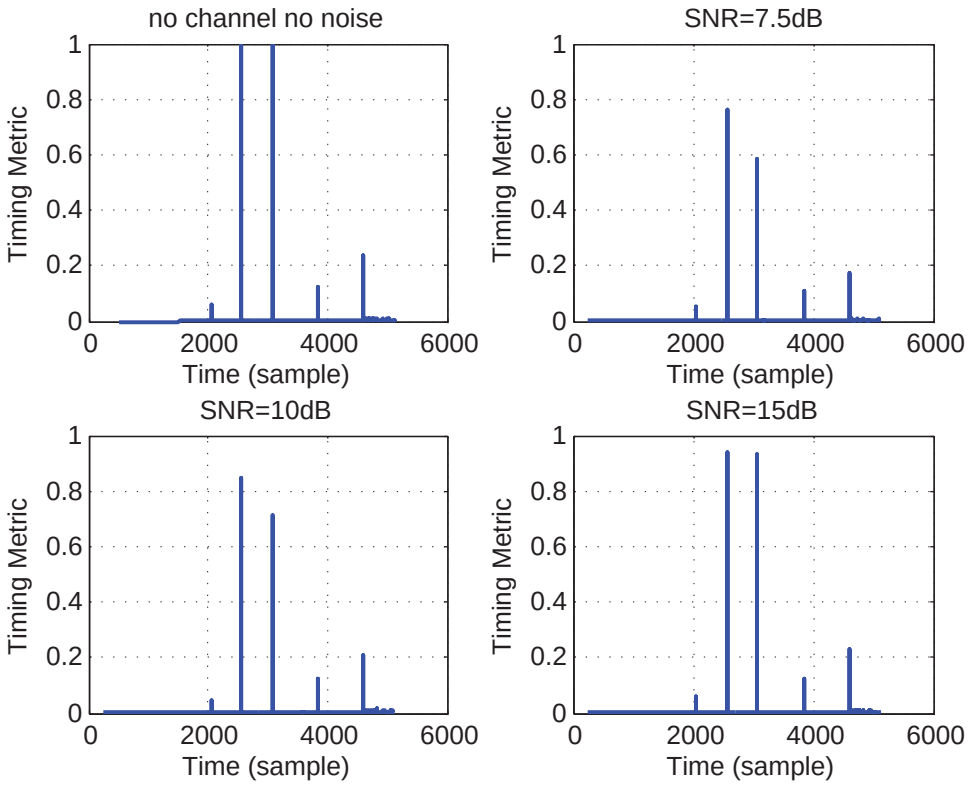

Figure 5: Park et al method simulation under ideal condition and AWGN channel

In Fig. 5, when there's no channel or noise, the Park method gives two peaks exactly at the start of the frame, i.e., $d = 2561$, and the start of the symbol (excluding the CP), i.e., $d = 3073$. Under the AWGN channel, it can be seen that the Park method is quite sensitive to the SNR given that the peaks shrink more from 1 as the SNR decreases, and the difference between the peaks is also enlarged as

the SNR goes down, particularly when $SNR = 7.5\,dB$, and $SNR = 10\,dB$, whereas when $SNR = 15\,dB$, the peaks are nearly of the same level.

The reason that the peaks shrink below 1 is due to the introduction of the additive white Gaussian noise. The enlarged difference between peaks is partly introduced by the AWGN channel, but mainly due to the fact that the training symbol in the time domain is not in the perfect mirrored symmetry within the complex conjugate symmetry. Given the inverse DFT

$$x_n = \frac{1}{N} X_k \cdot e^{j2\pi kn/N}, n \in \mathbb{Z}, \tag{17}$$

if the input sequence X_k is real, then the inversely transformed time domain sequence x_n would have the property

$$x_k^{(Re)} = x_{N-k}^{(Re)} \quad \text{and} \quad x_k^{(Im)} = -x_{N-k}^{(Im)}, \tag{18}$$

which implies that $x_0^{(Im)} = x_{N/2}^{(Im)} = 0$. However the symmetry would exclude the point corresponding to the Nyquist frequency point in the frequency domain , since $x_{N/2}^{(Re)} = x_{N-n/2}^{(Re)}$. Thus the complex conjugate symmetry of the last two quarters starts at the point $N/2+1$. The last quarter, nevertheless, is not affected, and the CP length of 25% of the symbol would just take the last quarter and append it at the beginning of the symbol. Thus when the sliding window is aligned with the beginning of the CP, ideally it would produce a peak of 1, and when the window is aligned with the beginning of the symbol (excluding the CP), the peak would be slightly lower than 1.

5 Evaluation Results

As mentioned in the introduction, the proposed acoustic sensor network is a full duplex linear bus system using the DMT vector coding scheme. Each sensor node is connected to the base station using a pair of twisted copper wires that would enable both upstream and downstream data transmission, and power supply for the sensors. Given that the realization of the multi-carrier modulation is by using the DMT scheme, signal of real values is transmitted over the wireline.

The transmitter processes the data by first converting the serial data bits into parallel, where the number of bits in each parallel channel depends on the constellation scheme in the modulator. The parallel bits are then modulated on a single carrier by using QAM (up to the constellation size of 10). Next the modulated symbols are passed through the IFFT block for channel partitioning. Finally, the CPs are added to the symbols, passed through the 12-bit DAC (Digital to Analog Converter) and transmitted. As for the receiver, it is reversely constructed, namely the 12-bit ADC (Analog to Digital Converter), the FFT block, the demodulator, and the parallel to serial converter, respectively.

The input to the IFFT at the transmitter is constructed in Hermitian symmetry as

$$X_k^* = X_{N-k}^*,$$

with 0 value at the $N/2$ frequency point to ensure the symmetry. Now by taking the IFFT (17), there will be a real output in a baseband transmission. At the receiver, the FFT can be directly applied to the received data to obtain the modulated signals.

The frequency response of the system is a combination of the frequency responses of the DAC, the channel of the twisted pair, and the ADC. The channel of a linear bus system is actually a multipath channel consisted of various taps and branches that introduces many reflections. Such a channel transfer function can be modelled by cascading the ABCD matrices of the two-port network [24].

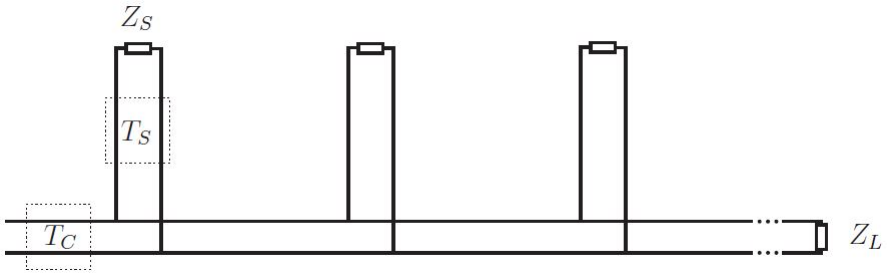

Figure 6: Linear bus topology [25]

Given the linear bus topology depicted in Fig. 6. Each tap can be represented as a two-port network. Given the cable length l, the ABCD matrix of the

network can each be calculated by using its secondary parameters γ and Z_0 as

$$T_c = \begin{bmatrix} \cosh\gamma l & Z_0 \sinh\gamma l \\ \frac{1}{Z_0}\sinh\gamma l & \cosh\gamma l \end{bmatrix},$$

The branching connection can then be modelled as a shunt impedance over the main line

$$T_s = \begin{bmatrix} 1 & 0 \\ \frac{1}{Z_{in}} & 1 \end{bmatrix},$$

where $Z_{in} = \frac{AZ_S+B}{CZ_S+D}$ is derived from the ABCD parameters of the branch two-port network, which is terminated by the impedance Z_s. Therefore the cascaded two-port network is constructed as [24]

$$T_{sys} = T_C^1 \cdot T_S^1 T_C^2 \cdot T_S^2 \cdot \cdots \cdot T_C^N \cdot T_S^N .$$

The transfer function of the linear bus channel can be calculated from the ABCD matrix T_{sys}. Given the output impedance of the transmitter Z_S and the input impedance of the receiver Z_L [24]

$$H(f)_{model} = \frac{Z_L}{AZ_L + B + CZ_SZ_L + DZ_S}.$$

Note that the above transfer function is only a channel model based on cable measurements, it is not the channel estimation and is not known to the receiver. The channel model is set up so as to further derive the channel delay and use it as a reference when comparing to the estimations of the channel delay using the three different synchronization methods. The channel delay θ_{model} can be estimated as the mean of the group delay [26]

$$\theta_{model} \triangleq E\left[-\frac{d}{d\omega}\Theta(\omega)\right] \triangleq E\left[-\frac{d}{d\omega}\angle H_{model}\left(e^{j\omega T}\right)\right]. \tag{19}$$

The DMT system has the FFT/IFFT size of $N = 2048$, the CP size $L = 512$, the frame size $N_s = N + L = 2560$, the modelled channel delay $\theta_{model} = 126$ (time samples), the maximum number of available sub-carriers $N_{sub} = N/2 = 1024$ (due to the Hermitian symmetry structure for the input of the IFFT block in the transmitter), and the actual number of sub-carriers in use $N_{used} = 761$ (due to the lower and upper cut-off frequencies of the filters).

The system is simulated under colored noise environment in each subchannel, and with the white quantization noise of the IFFT block, the DAC, the ADC, and the FFT block. The modulation scheme used in the simulation is BPSK (Binary Phase Shift Keying).

Different from the evaluation simulations earlier, N_{sync} number of the same training symbols, or information symbols in each of the three synchronization methods, are sent at the receiver. $N_{sync}/2$ number of observation frames, which is the range that the sliding window stays in, is constructed to be the first received $2N + L$ samples for every two symbols (including the CPs) at the receiver.

This is done mainly to repetitively simulate the algorithm and find various estimations with less computation time than sending the signals over and over again.

5.1 Blind Synchronization

5.1.1 ML Estimation Method

When $N_{sync} = 2$, there's one observation frame consisting of $2N + L$ samples. The time index d in $\gamma(d)$ (6) and $\Theta(d)$ (7) starts at $d = 0$ and ends at $d = N = 2048$, i.e., the window slides by the length of the N. The complex conjugate in (6) is no longer needed since the output of the IFFT is a real sequence. The coefficient ρ in 8 is omitted in the simulation since all values in $\Theta(d)$ are scaled by ρ, and it would not affect the argument in finding the time index that corresponds to the maximum. In addition, the channel is not known to the receiver, and the SNR would only be an estimation. The plot of the simulated timing metric is shown in Fig. 7 .

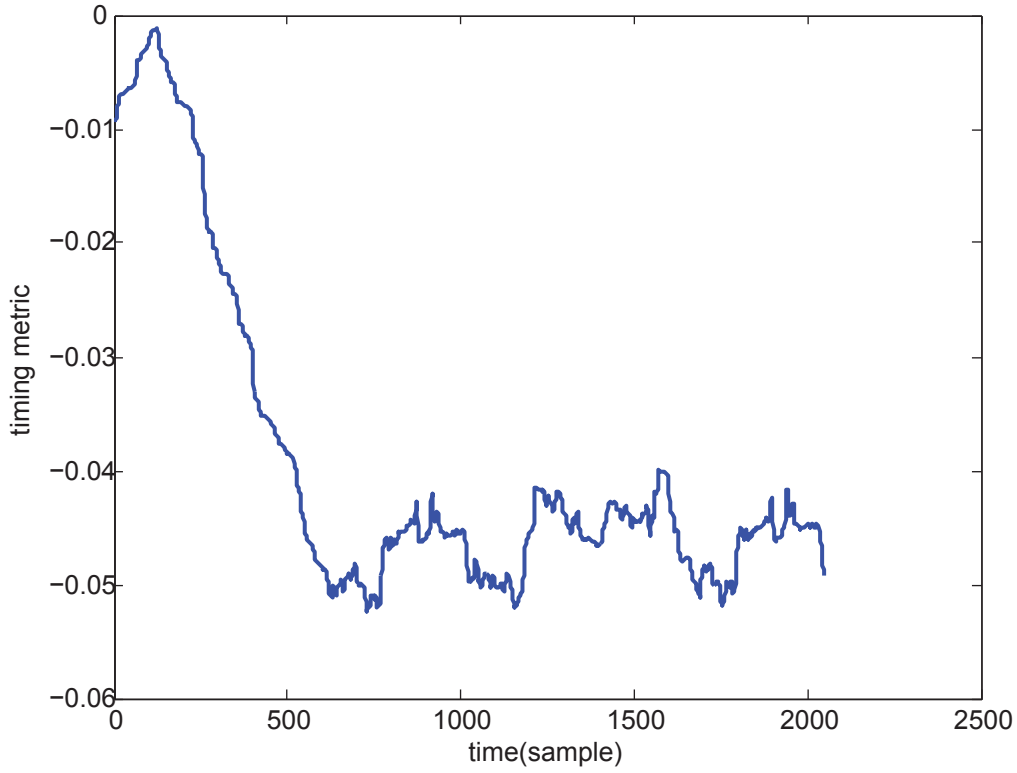

Figure 7: ML estimation method in DMT system

From Fig. 7 it can be seen that the peak locates at around $d = 130$, where ideally the channel delay is $\theta_{model} = 126$ in (19). However the estimate is not

stable when simulated various times due to fact that part of the CPs are corrupted by the channel. The ML estimation method was then run for 300 times, i.e., $N_{sync} = 600$, and the time offset estimation distribution histogram is plotted as the number of estimation occurrences against the estimation of the channel delay in Fig. 8 .

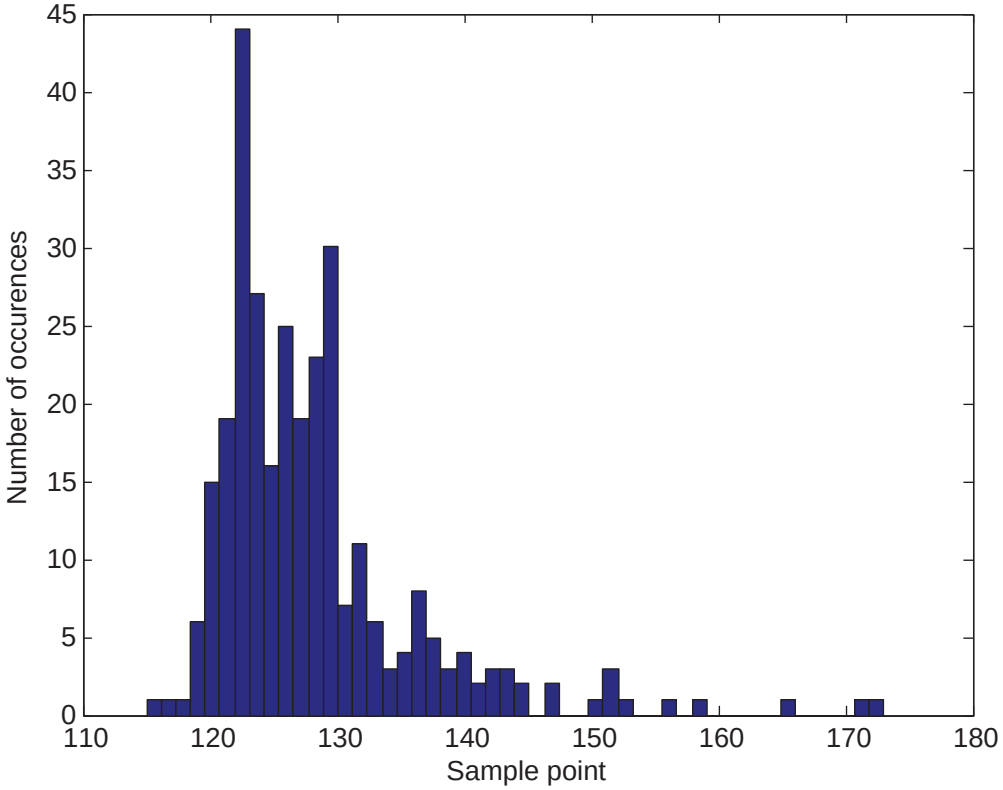

Figure 8: Time offset estimation distribution of the ML estimation method

From 8 it can be seen that peak locates approximately at $d = 126$ (most number of estimations). However the range of distribution is quite wide and suggests great uncertainty in the estimation. Alternatively, a more robust estimation can be obtained by averaging the estimates of various overheads. The statistics are shown in Fig. 9, it can be observed that as the number of overhead increases, the mean of the estimates tends to stabilize around $d = 129$, which is quite close to $\theta_{model} = 126$ and the variance reduces. In order to achieve a estimation variance below 0.1, at least 700 overheads are needed.

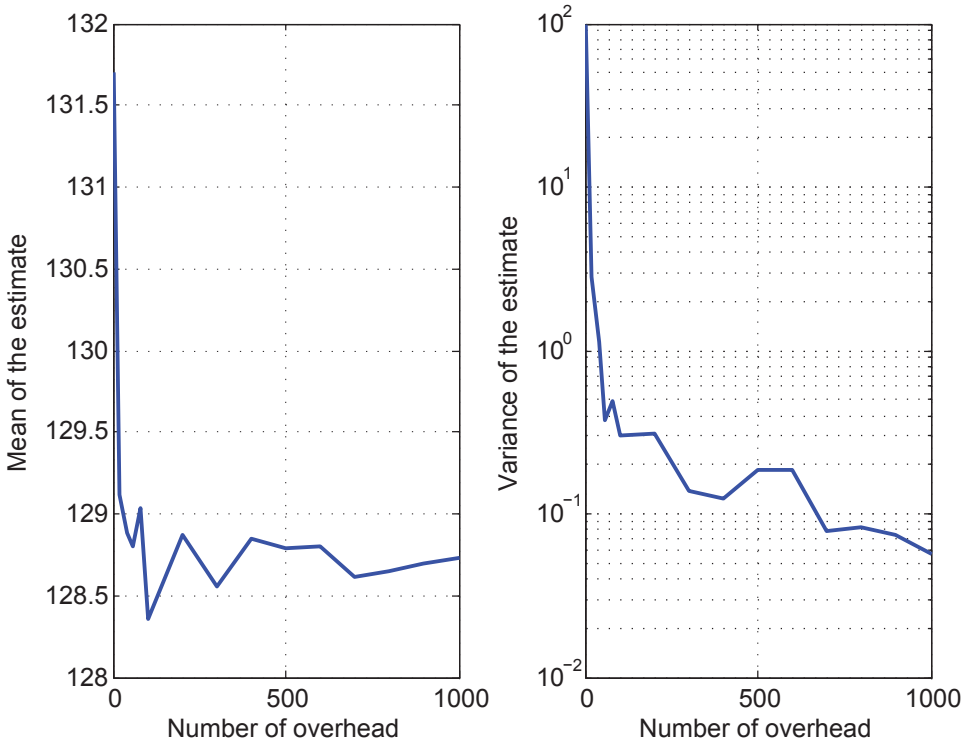

Figure 9: Statistics of the ML estimation against the number of overheads

5.2 Data-aided Synchronization

5.2.1 Schmidl and Cox Method

When $N_{sync} = 2$, there's one observation frame consisting of $2N + L$ samples. The time index d in $R(d)$ (12) and $P(d)$ (11) starts at $d = 0$ and ends at $d = N_s =$ 2560, i.e., the window slides by the length of $N_s = N + L$. The complex conjugate in (12) is not longer needed since the output of the IFFT is real. The plot of the simulated timing metric is shown in Fig. 10 .

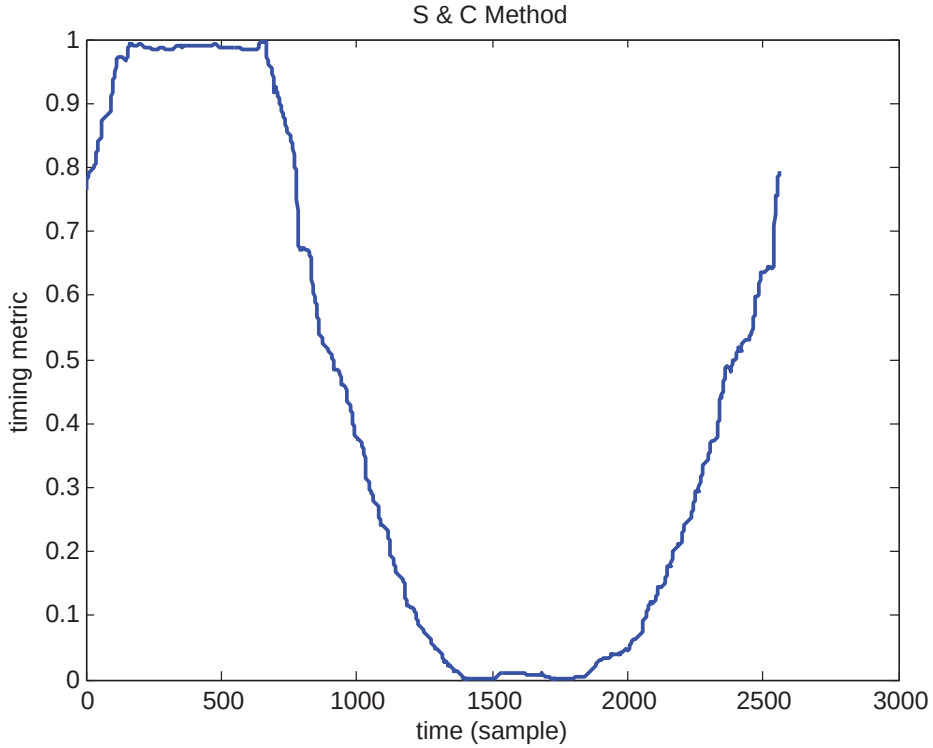

Figure 10: S & C estimation method in DMT system

From Fig. 10 it can be seen that the plateau starts around $d = 160$, i.e., the 160th time sample, and ends around $d = 635$. Ideally the start should be at $d = 126$ as calculated from the mean of the group delay in (19), and ends at $d = \theta_{model} + L = 126 + 512 = 638$. The simulation was run for 300 times, i.e., $N_{sync} = 600$ and the time offset estimation distribution histogram is plotted as the number of estimation occurrences against the estimation of the channel delay in Fig. 11 .

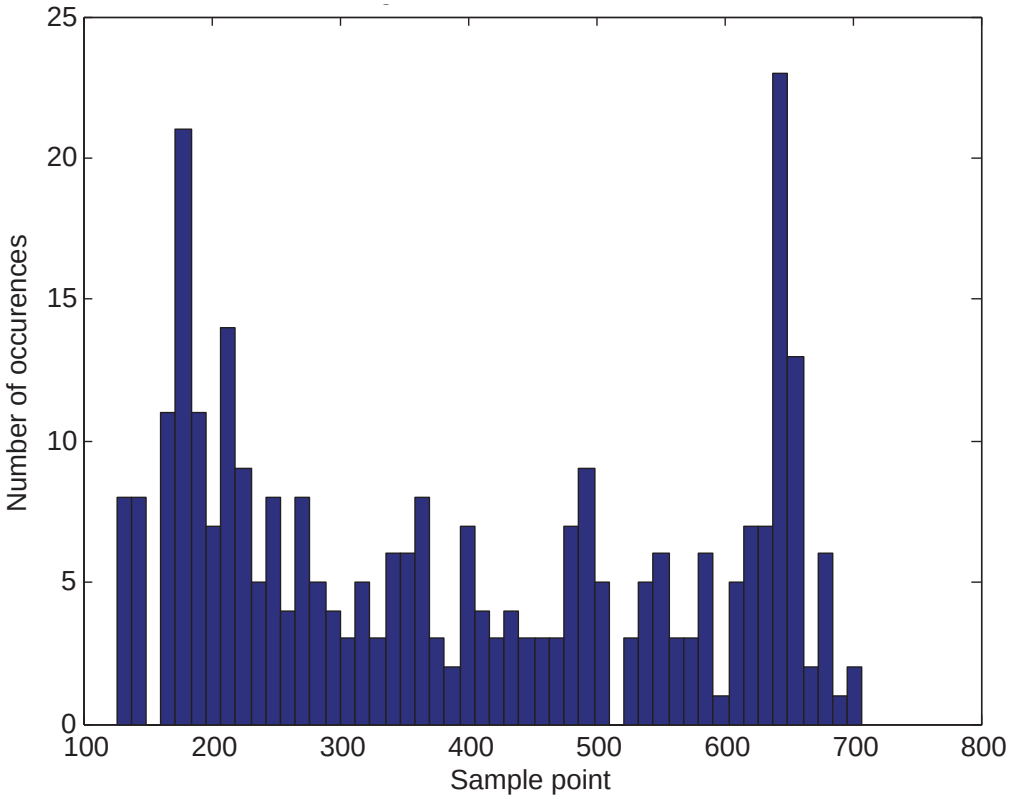

Figure 11: Time offset estimation distribution of the S & C method

In Fig. 11 it can be seen that the peaks (most number of estimations) are located around $d = 160$ and $d = 630$, however the wide range of distribution implies great uncertainty in the estimation. Now by taking 90% average to both sides of the peak, the time offset distribution is depicted in Fig. 12, where the peak locates at around $d = 390$, and the distribution range is slightly reduced.

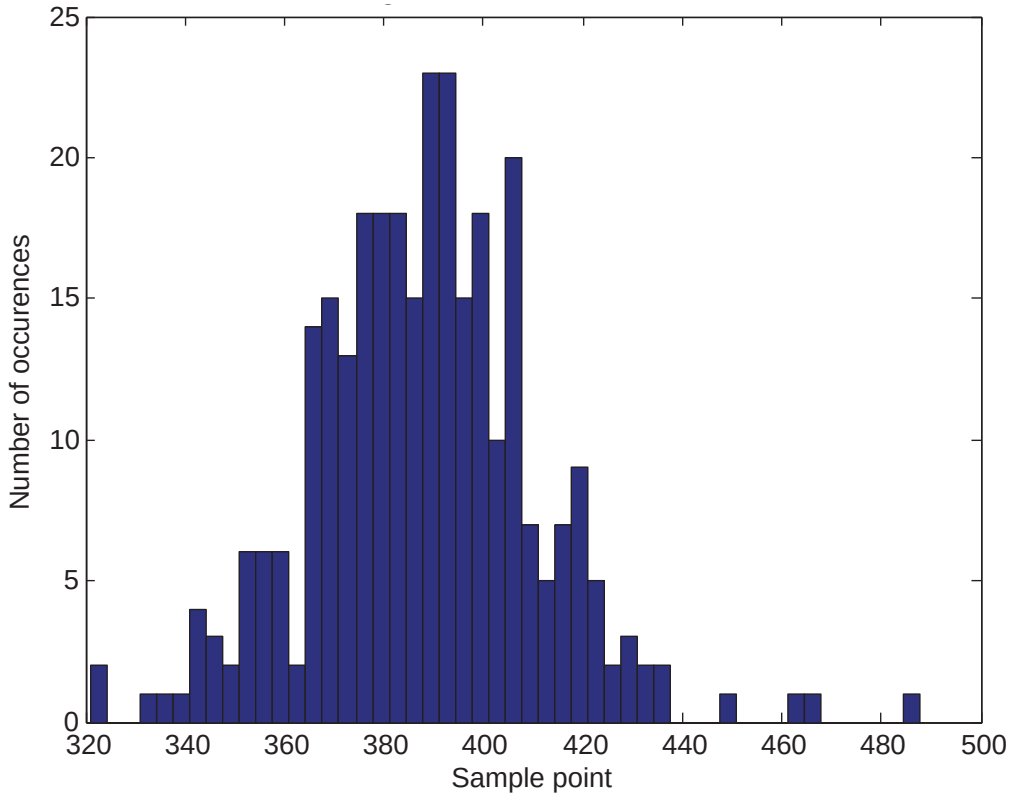

Figure 12: Time offset estimation distribution of the average S & C method

5.2.2 Park et al. Method

When $N_{sync} = 2$, there's one observation frame consisting of $2N + L$ samples. The time index d in $R(d)$ (15) and $P(d)$ (14) starts at $d = N/2 = 1024$ and ends at $d = 2N_s - L = 2560$, i.e., the window slides by the length of $N_s = N + L$. The time index has an offset of $d = N/2$ in order to satisfy $r(d-k)$ in $P(d)$ (14). The offset is then removed at the timing metric $M(d)$ (16) after adding all of the products in the sliding window such that the time index actually starts at $d = 0$, and ends at $d = N_s = 2560$. The timing metric and the estimation is shown in Fig. 13 .

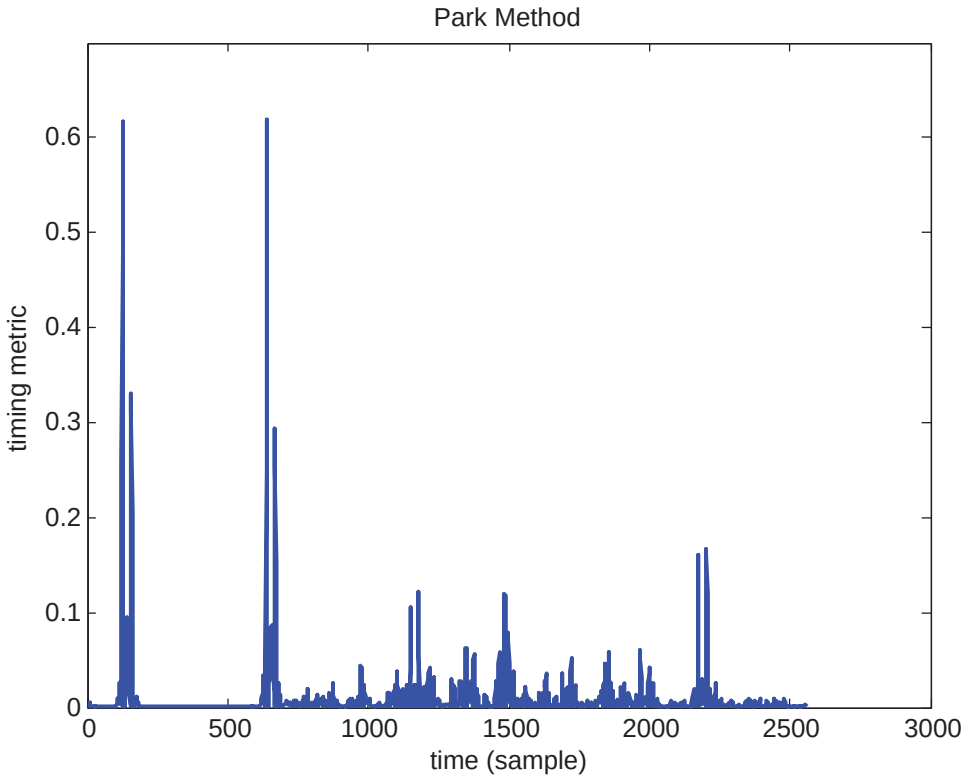

Figure 13: Park method in DMT system

From Fig. 13 it can be seen that the two peaks are located exactly at $d = 126$, and $d = 638$, which are the ideal estimations. The simulation was run for 300 times, i.e., $N_{sync} = 600$ and the time offset estimation distribution histogram is plotted as the number of estimation occurrences against the estimation of the channel delay in Fig. 14 .

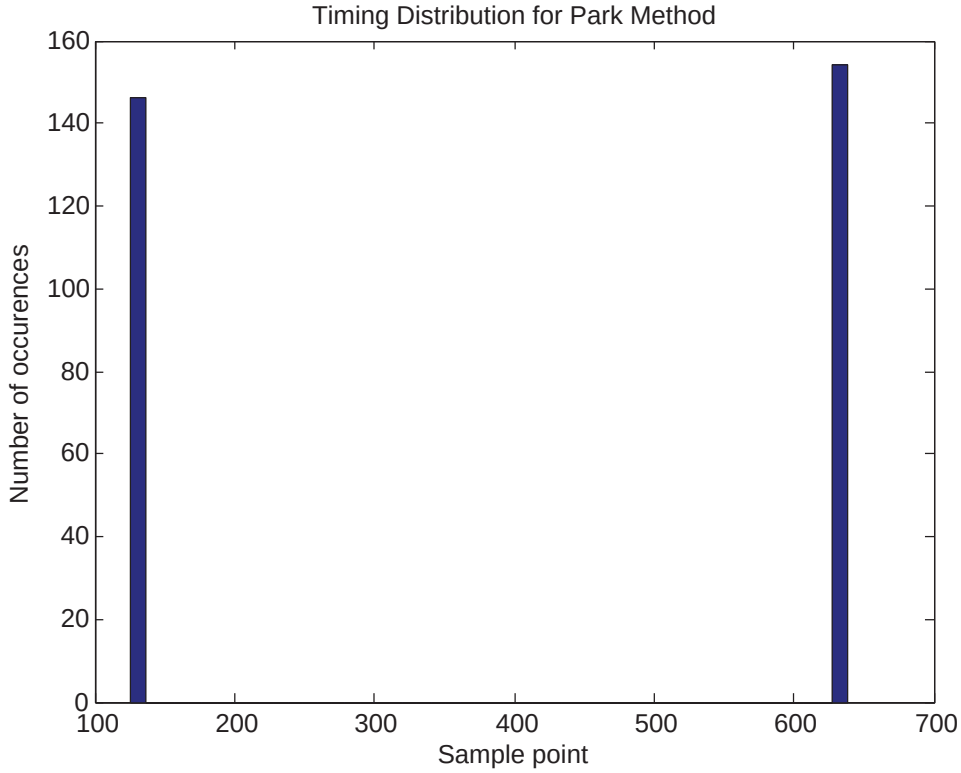

Figure 14: Time offset estimation distribution of the Park method

From Fig. 14 it can be seen that all time offset estimates are exactly either $\hat{\theta} = \theta_{model} = 126$ or $\hat{\theta} = \theta_{model} = 638$, which is quite robust and accurate. As it can be seen from the simulations of the Park method in AWGN channels in Firgure5, the peaks are quite sensitive to the SNR. In the DMT system, the transmitting power of each DMT sender in the sub-channel is varied, equivalently to varying the SNR, and plotted the statistics of the two peaks after 300 runs as depicted in Fig. 15 .

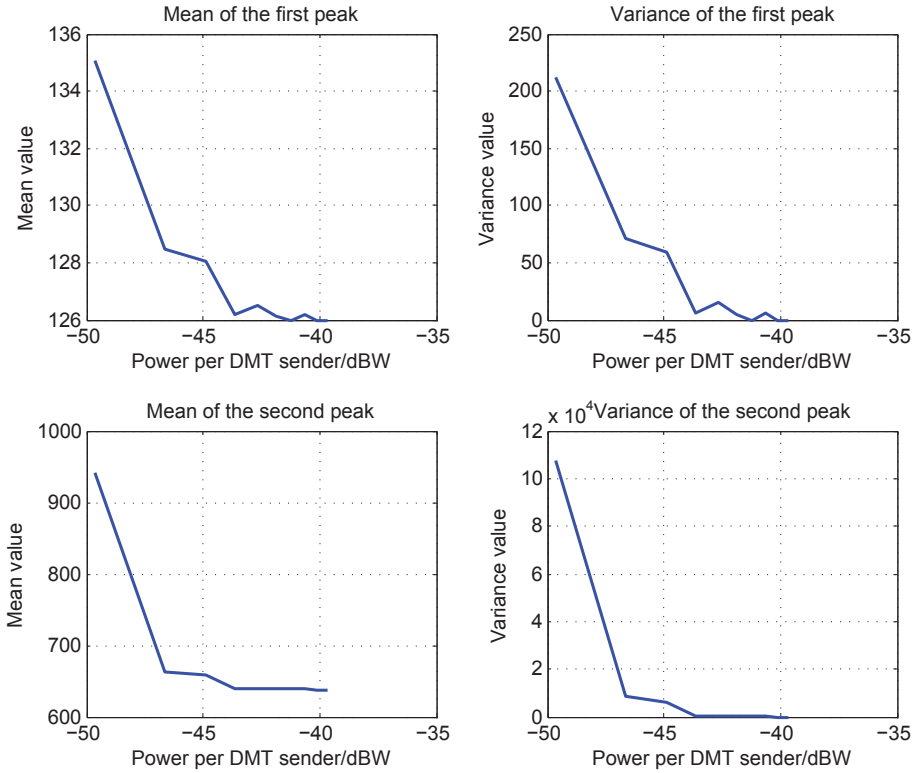

Figure 15: Statistics of the Park method with varying transmitting power

From Fig. 15 it can be seen that starting from the transmitting power of about -40 dBW, the mean of the 300 estimates of the first peak starts exactly at $\hat{\theta} = \theta_{model} = 126$, and increases as the transmitting power is reduced. The same trend can also be observed for the variance, starting at 0 and increases as the transmitting power decreases. For the second peak at $\theta_{model} = 638$, it can be observed with the similar pattern as the first peak. Given that the transmitting power per DMT sender in this particular sensor network is -19.6 dBW, which is about 100 times greater than the point where the means and variances of the peaks start to increase, the Park method is well suited for the particular channel.

5.2.3 Comparison of the Synchronization Methods

The Schmidl and Cox method, averaged Schmidl and Cox method, Park method, and the ML estimation method, are compared with their statistics for 300 runs. The results are depicted as an error bar plot in Fig. 16 .

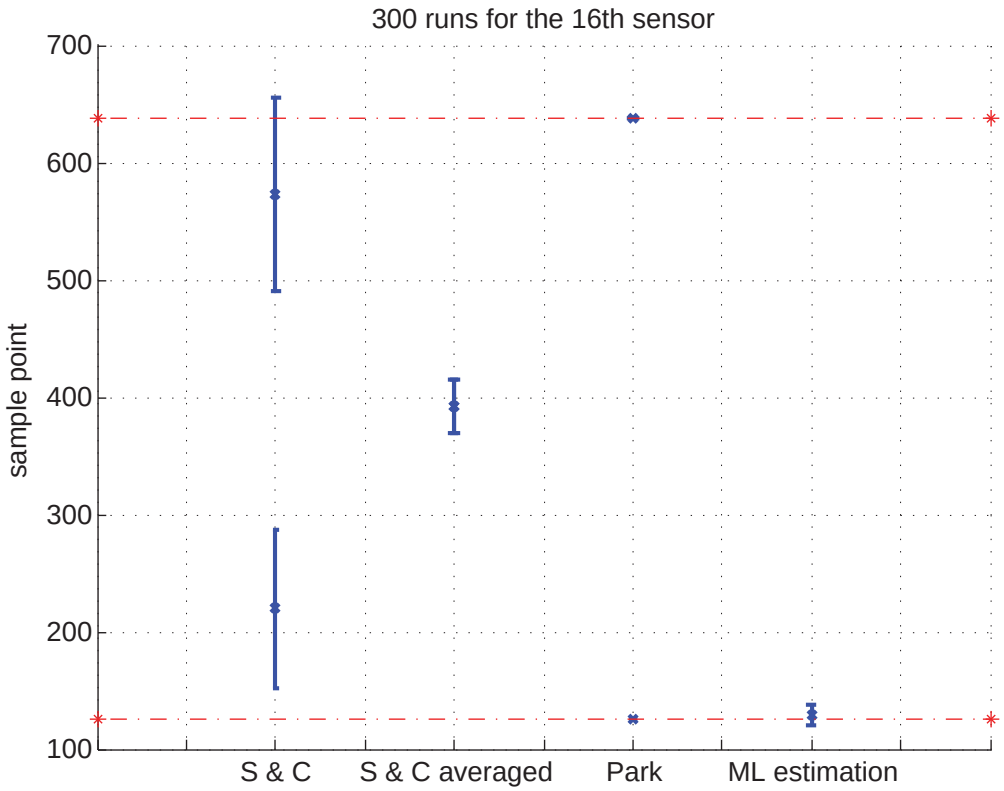

Figure 16: Comparison of the synchronization methods

In Fig. 16, the error bars indicate the upper and lower standard deviation from the mean, i.e., $\mu_{\hat{\theta}} \pm \sigma_{\hat{\theta}}$, by using the different synchronization methods. The red dotted lines at sample point $d = \theta_{model} = 126$, and $d = \theta_{model} + L = 126+512 = 638$ are the modelled delay of the channel, and the modelled starting of the symbol (excluding the CP), respectively.

From Fig. 16 it can be seen that for the Schmidl and Cox method, the mean of the estimates for the start and the end of the plateau are around $d = 219$ and $d = 572$, which is equivalent to the length of CP minus the length of the CIR, and within the interval of $d \in [126, 638]$. However the deviation for the estimates of the end of the plateau extends over the upper boundary $d = 638$, which can give wrong estimations. In addition, the wide range of standard deviation raises uncertainty in the estimates. Thus the Schmidl and Cox method is not suitable for the frame synchronization of this particular DMT system. The averaged Schmidl and Cox method, however gives a relatively more robust estimate and narrower

range of deviations.

The ML estimation method gives a comparatively smaller confidence interval of standard deviations. The lower standard deviation, however, extends below the lower bound $d = \theta model = 126$. This would not be a problem since the 300 runs can be seen as having 300 symbols of overhead, and the estimate is now the mean of al the estimates, which is nearly $d = \hat{\theta} = 129$, quite close to the modelled channel delay.

The Park method gives the best estimates as illustrated in Fig. 16 where there's no standard deviation at all. The mean of all the estimates for the two peaks are exactly located at $d = \theta model = 126$ and $d = \theta_{model} + CP = 638$, meaning that all the estimates for 300 runs are the same. Thus the Park method is the most robust, accurate, and efficient (needs only one training symbol without any averaging) algorithm of all the other algorithms discusses in this paper.

6 Conclusion

This paper compared three frame synchronization methods for a proposed DMT system used for the MC-TP sensor network. Namely the ML estimation method, Schmidl and Cox method, and the Park method.

The ML estimation method makes use of the correlation of the cyclic prefixes. When the sliding window of the timing metric is aligned with the start of the frame, a peak will be generated and detected by the maximum likelihood estimator so as to obtain the corresponding estimate for the channel delay.

The performance of ML estimation method is quite unstable since the received cyclic prefixes in the frames are partly corrupted by the channel. In real practice, an overhead of hundreds of information frames are transmitted, and the receiver needs to first store this large overhead, compute the average, and take the average as the channel delay estimate. The simulation result in Fig. 9 shows that the estimation is quite robust and has a estimation variance lower than 0.1 when the overhead is of the size 700. This however, would demand large memory space at the receiver and much longer computation time than the other two methods discussed in this paper.

The Park method and the Schmidl and Cox method both belongs to the data-aided synchronization, where a frequency domain training symbol is appended at the beginning of the information symbol in the transmitter, and the received time domain training symbol is used for the use of channel delay estimation. The training symbol is typically constructed in the frequency domain such that the time domain sequence would have a certain symmetry property. The sliding window is then applied to form a specific timing metric such that when the window is aligned with the start of the frame and the start of the symbol (excluding the CP), the peaks will be produced. The corresponding time index of the first peak is then the time offset estimation.

From the simulation in Fig. 10 and Fig. 11, it can be seen that the plateau in the Schmidl and Cox algorithm introduces a large range of distribution and estimation uncertainty. The Park method, on the other hand, as demonstrated in Fig. 13 and Fig. 14, the peaks are quite distinctive and the estimations are very robust.

Comparing the three methods, the Park method shows the best performance in robustness and accuracy among all three, as depicted in Fig. 16. In terms of computational efficiency, the Park method requires much less time and memory space at the receiver than the ML estimation method. Therefore, the Park method would be the optimal solution of frame synchronization among the three, in the frame of the proposed DMT sensor network.

References

[1] J.-J. Van de Beek, M. Sandell, P. O. Borjesson *et al.*, "Ml estimation of time and frequency offset in ofdm systems," *IEEE transactions on signal processing*, vol. 45, no. 7, pp. 1800–1805, 1997.

[2] T. M. Schmidl and D. C. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Transactions on Communications*, vol. 45, no. 12, pp. 1613–1621, 1997.

[3] B. Park, H. Cheon, C. Kang, and D. Hong, "A novel timing estimation method for ofdm systems," *Communications Letters, IEEE*, vol. 7, no. 5, pp. 239–241, 2003.

[4] V. Burstein, "New Bus Systems for Acoustical Sensor Networks Ph.D. Proposal," 2014.

[5] *Fundamentals of DSL Technology*. Auerbach Publications, 2005, ch. 7 Fundamentals of Multi-Carrier Modulation.

[6] T. Magesacher. (2006) OFDM & DMT in a nut shell. [Online]. Available: http://www.ist-muse.org/

[7] J. M. Cioffi, "EE379A course notes."

[8] J.M.Cioffi, *EE379A Course Notes*, ch. Signal Processing and Detection, p. 59.

[9] ——, *EE379A Course Notes*, ch. Signal Processing and Detection, p. 63.

[10] J. M. Cioffi, "EE379C course notes."

[11] L. Goldfeld, V. Lyandres, and D. Wulich, "Minimum BER power loading for OFDM in fading channel," *IEEE Transactions on Communications*, vol. 50, no. 11, pp. 1729–1733, 2002.

[12] E. Bedeer, O. A. Dobre, M. H. Ahmed, and K. E. Baddour, "Joint optimization of bit and power loading for multicarrier systems," *Wireless Communications Letters, IEEE*, vol. 2, no. 4, pp. 447–450, 2013.

[13] A. Dixit. OFDM: Combating ISI and ICI with Cyclic Prefix. [Online]. Available: https://lyle.smu.edu/~amitabh/OFDM.htm

[14] J. Tellado and J. M. Cioffi, "Peak power reduction for multicarrier transmission," in *Proc. IEEE GLOBECOM. Conf.* Citeseer, 1998.

[15] J. S. Chow and J. M. Cioffi, "A cost-effective maximum likelihood receiver for multicarrier systems," in *IEEE International Conference on Communications, 1992. ICC'92, Conference record, SUPERCOMM/ICC'92, Discovering a New World of Communications.* IEEE, 1992, pp. 948–952.

[16] N. Al-Dhahir and J. M. Cioffi, "Optimum finite-length equalization for multicarrier transceivers," *IEEE Transactions on Communications*, vol. 44, no. 1, pp. 56–64, 1996.

[17] W. Henkel and T. Kessler, "Maximizing the channel capacity of multicarrier transmission by suitable adaptation of the time-domain equalizer," *IEEE Transactions on Communications*, vol. 48, no. 12, 2000.

[18] P. J. Melsa, R. C. Younce, and C. E. Rohrs, "Impulse response shortening for discrete multitone transceivers," *IEEE Transactions on Communications*, vol. 44, no. 12, pp. 1662–1 2, 1996.

[19] R. K. Martin, K. Vanbleu, M. Ding, G. Ysebaert, M. Milosevic, B. L. Evans, M. Moonen, and C. R. Johnson, "Unification and evaluation of equalization structures and design algorithms for discrete multitone modulation systems," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3880–3894, 2005.

[20] B. Berriah, M. Bouziani, and S. A. Elahmar, "New blind, adaptive channel shortening TEQ for multicarrier modulation systems," *IET Communications*, vol. 8, no. 2, pp. 210–216, 2013.

[21] G. Ysebaert, K. Van Acker, M. Moonen, and B. De Moor, "Constraints in channel shortening equalizer design for DMT-based systems," *Signal Processing*, vol. 83, no. 3, pp. 641–648, 2003.

[22] K. Van Acker, "Equalization and echo cancellation for DMT-based DSL modems," *Katholieke Universiteit Leuven, Belgium*, 2001.

[23] *Essentials of Digital Signal Processing*. Cambridge University Press, 2014, ch. 1.9.

[24] T. Banwell and S. Galli, "A new approach to the modeling of the transfer function of the power line channel," in *Proceedings of the 5th International Symposium on Power-Line Communications and its Applications (ISPLC)*. Citeseer, 2001, pp. 4–6.

[25] V. Burstein, "New Bus Systems for Acoustical Sensor Networks Ph.D. Proposal," p. 13, 2014.

[26] "Computing the group delay of a filter," http://www.dsprelated.com/showarticle/69.php, 2008, accessed: 2015-05-20.