

Mitigating Sampling Frequency Offset in OFDM: A Comparative and Machine Learning-Based Approach

[Moatasem M. E. Kotb](#)^{1*}, [Maha R. Abdel-Haleem](#)¹, [A.Y.Hassan](#)¹, and [Ashraf S. Mohra](#)¹

¹ EE Dept. Benha faculty of engineering, Benha University, Benha, Egypt.

E-mail: Mutasim.Qutb20@beng.bu.edu.eg

Abstract

This paper addresses the critical issue of Sampling Frequency Offset (SFO) in Orthogonal Frequency Division Multiplexing (OFDM) systems, which arises from mismatches between the sampling rates of the transmitter and receiver. Such discrepancies disrupt subcarrier orthogonality, leading to significant performance degradation, including Inter-Carrier Interference (ICI), phase distortion, and increased Bit Error Rate (BER). To ensure reliable data transmission, accurate SFO estimation and compensation are essential. The study examines four widely used SFO estimation techniques: the Phase Difference (PD) method, the Correlation-Based (CB) method, the Phase Difference Weighted by Subcarrier Index (PD-WSI) method, and the Hybrid Estimation (H-EST) method. Additionally, it introduces novel machine learning-based approaches—Linear Discriminant Analysis (LDA)-based, Kernel Support Vector Machine (KSVM)-based, and Artificial Neural Network (ANN)-based SFO estimators—designed to enhance synchronization accuracy. Comparative evaluations demonstrate that these proposed methods significantly outperform conventional and hybrid techniques by achieving lower Root Mean Square Error (RMSE), thereby effectively mitigating SFO-induced impairments and improving overall OFDM system performance.

Keywords: SFO estimation, Data-Aided (DA) techniques, LDA-based SFO Estimator, KSVM-based SFO Estimator, ANN-based SFO Estimator.

I. INTRODUCTION

Orthogonal Frequency Division Multiplexing (OFDM) has become a fundamental modulation technique in modern wireless communication, enabling high data rates, efficient spectrum utilization, and robustness against multipath fading. As a key technology, it underpins various applications, including broadband wireless networks, digital television broadcasting, and next-generation mobile communication systems like 5G and beyond. However, despite its advantages, OFDM is highly vulnerable to synchronization errors, particularly Sampling Frequency Offset (SFO), which can significantly degrade system performance if not properly managed [1]. SFO occurs when there is a mismatch between the sampling clocks of the transmitter and receiver. This discrepancy can stem from oscillator imperfections, temperature variations, hardware constraints, and long-term clock drifts. SFO induces a linearly increasing phase shift across subcarriers, disrupting the orthogonality of the system and leading to Inter-Carrier Interference (ICI). This interference compromises signal integrity, ultimately degrading data reliability and overall transmission

quality.

The impact of SFO on OFDM systems is complex and significant. One of the primary effects is phase distortion, which leads to errors in symbol detection and an increase in the Bit Error Rate (BER). Additionally, SFO causes time-domain misalignment, where the symbols gradually drift, resulting in reduced synchronization accuracy and improper sampling. This misalignment not only hampers data recovery but also diminishes overall system efficiency. Moreover, SFO disrupts the matched filter's output by introducing phase shifts and timing errors, which degrade signal detection and lower the Signal-to-Noise Ratio (SNR). If left unaddressed, these issues can substantially degrade network performance, restricting data throughput and reducing the reliability of wireless communication [2]. To mitigate the effects of SFO, researchers have developed various estimation techniques. A widely used approach is pilot-aided estimation, where predefined pilot symbols embedded within the OFDM signal serve as reference points for detecting and correcting frequency mismatches. Another technique, data-aided synchronization, leverages

known reference data to enhance frequency offset estimation accuracy. Additionally, blind estimation methods analyze inherent signal properties without relying on additional reference symbols, making them beneficial in bandwidth-limited scenarios. While these conventional techniques have improved synchronization accuracy, achieving precise and efficient SFO correction remains a significant challenge, particularly in dynamic environments with unpredictable channel variations [3].

Through machine learning (ML), SFO estimation has advanced with adaptive techniques that optimize synchronization accuracy. Unlike traditional approaches that rely on predefined mathematical models, ML-based methods can analyze vast amounts of received signal data, detect complex patterns, and dynamically adjust estimation strategies to account for changing wireless conditions. This adaptability makes ML particularly useful in environments where channel conditions fluctuate rapidly. One of the key advantages of ML in SFO estimation is its ability to handle nonlinear distortions and unpredictable variations in signal behavior. Traditional estimation techniques often struggle with inaccuracies in real-world scenarios due to hardware imperfections, oscillator drift, and noise interference. In contrast, ML algorithms can continuously learn from real-time data, refining their predictions and improving synchronization precision over time. This capability enables more resilient and efficient communication systems, reducing errors and enhancing overall network performance. Another important aspect of ML-based SFO estimation is its potential for automating complex signal processing tasks. Conventional methods require extensive manual tuning of parameters and reliance on fixed assumptions about the communication channel. ML-driven approaches, however, can automatically extract meaningful features from received signals and adapt synchronization. This not only reduces computational complexity but also enhances the scalability of wireless systems, making ML an attractive solution for next-generation communication technologies [4]. Despite its advantages, the integration of ML into SFO estimation comes with challenges.

Computational demands, data availability, and real-time processing constraints are critical factors. Our research seeks to optimize ML models to overcome these barriers, enhancing efficiency and reducing computational strain. By leveraging the evolving capabilities of ML, we aim to enhance synchronization accuracy and contribute to the development of more robust, adaptive, and intelligent communication systems.

In this work, we pursued three primary objectives:

- **Assess the impact of sampling frequency offset** on OFDM system performance, with a focus on BERs.
- **Evaluate existing SFO estimation methods**, Implement and evaluate existing SFO estimation methods, analyzing their performance, strengths, and limitations.
- **Develop and propose a machine learning-based SFO estimation approach** that enhances accuracy, minimizes synchronization errors, and improves overall system reliability and efficiency.

The Essential Findings and Contributions are:

- **Comprehensive Performance Evaluation:** A detailed comparison of traditional, hybrid, and machine learning-based SFO estimation methods under various SNR levels and fading channel conditions.
- **Superiority of the proposed ML-Based SFO Estimators:** Demonstrated that Linear Discriminant Analysis (LDA)-based and Artificial Neural Network (ANN)-based estimators achieve significantly lower Root Mean Square Error (RMSE) and BER values, outperforming conventional and hybrid approaches in both accuracy and robustness.
- **Analysis Under Fading Channels:** Provided insights into estimator performance in realistic fading environments, showing that ANN and LDA maintain high reliability.

- **No Need for Perfect Channel State Information:** Demonstrated that the proposed ML-based SFO estimators do not require perfect CSI, making them more robust and adaptable to practical deployment scenarios with imperfect channel information.
- **Dimensionality Reduction for Efficiency:** Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) were evaluated to reduce computational overhead. SVD achieved a 96.14% reduction in processing time, while PCA delivered a 37.91% improvement in estimation performance.
- **Latency and Implementation Considerations:** Assessed inference latency across methods, highlighting trade-offs between estimation accuracy and computational speed for practical system deployment.
- **Potential to Replace Existing SFO Estimation Techniques:** Given their superior accuracy, our proposed method has the potential to replace existing SFO estimation techniques in modern digital communication systems.

The paper is organized as follows: Section II reviews related work, Section III introduces the proposed method, Section IV presents the results along with their analysis, and Section V concludes the paper with final remarks.

II. SURVEY

Earlier research on estimating SFO has involved a range of different techniques. Fischer et al. [5] introduced a two-step frequency synchronization method for Orthogonal Frequency Division Multiplexing receivers, designed to enhance the precision of frequency offset detection and correction. The first step, called frequency acquisition, uses an Fast Fourier Transform (FFT)-based power spectral density estimation to detect initial frequency offsets by locating frequency pilots from the Digital Radio Mondiale (DRM). In the second step, frequency tracking, the synchronization process is refined by measuring phase changes between consecutive OFDM symbols, constantly adjusting any

residual errors. This approach also accounts for sample rate offsets by analyzing frequency deviations from multiple pilots. While effective, it requires averaging over several symbols to reduce errors due to fading and noise. Shin, Seo, and You [6] explored SFO estimation in OFDM-based Digital Radio Mondiale systems and proposed two key algorithms. Algorithm A estimates SFO by using phase differences between adjacent Frequency Reference Cells (FRCs), enhancing accuracy through temporal correlation. Algorithm B extends this by including differential relations between FRC indices, improving estimation precision. The authors also introduced a Low-Complexity Estimation technique, which simplifies the process by using only the first and last noise measurements, reducing computational requirements. Shim et al. [7] put forward a two-step synchronization method to tackle key issues in OFDM-based FM broadcasting systems, including timing offset, carrier frequency offset, and sampling frequency offset. The first phase, pre-FFT synchronization, estimates symbol timing via cyclic prefix correlation and computes fractional frequency offset by examining phase differences between the cyclic prefix and the useful section of the OFDM symbol. The second phase, post-FFT synchronization, refines the frequency offset estimation by using time reference cells for integer frequency offset determination and further tracks residual frequency offset and SFO with gain reference cells. Importantly, the mitigation of SFO in this method is achieved using a post-FFT technique that calculates phase differences, leading to more precise estimations. Jung and Young-Hwan You [8] proposed a resilient SFO estimation technique for OFDM systems operating in frequency-selective fading channels. Their method incorporates two essential techniques: temporal correlation and frequential correlation. The temporal correlation method utilizes the relationship between received pilot symbols to reduce the impact of unknown channel responses, mitigating the effects of time-varying channel conditions. On the other hand, the frequential correlation method enhances accuracy by taking advantage of the symmetry of pilot subcarriers around the DC carrier, effectively eliminating

bias introduced by frequency-selective fading. The combination of these two techniques strengthens the robustness of the SFO estimation, making it more resistant to challenging channel conditions.

Harish, Chuppala, Rajasekar Mohan, and R. Shashank [9] proposed a technique for estimating and correcting Sampling Frequency Offset (SFO) in OFDM receivers, building on the assumed proportional link between SFO and Carrier Frequency Offset (CFO). Their method is based on the premise that both the receiver's sampling clock and its carrier frequency oscillator share the same reference source, establishing a mathematical correlation between CFO and SFO. The process begins by estimating the CFO using training sequences within the received OFDM frame, applying autocorrelation methods. Once the CFO is determined, the SFO is derived using the equation $\delta f_c/f_c = \delta f_s/f_s$, enabling systematic correction of errors caused by SFO. The correction is performed in the frequency domain by applying a phase compensation factor to the FFT output, improving synchronization without the need for complex hardware clock recovery systems. While computationally efficient, this approach assumes that both oscillators are derived from the same reference, which might not hold true in practical scenarios. Hsiao et al. [10] introduced a clock synchronization approach for OFDM-based communication systems, focusing on estimating and correcting SFO within the frequency domain. Their technique involves observing phase rotation across OFDM subcarriers to estimate the SFO, offering a computationally efficient method for synchronization. However, the study does not include a direct comparison with other existing SFO estimation methods, making it difficult to assess the relative performance of this approach. Moreover, the method might be sensitive to varying channel conditions, which could affect its robustness in real-world implementations. Another drawback is the assumption of a free-running analog-to-digital converter (ADC), which might not be compatible with all hardware configurations, limiting its practical use in OFDM systems. Liu et al. [11] performed an in-depth examination of various synchronization

algorithms for OFDM systems, specifically targeting symbol timing synchronization, carrier frequency synchronization, and sampling clock synchronization. Their study reviews a range of methods, such as the Schmidl & Cox (SCA) algorithm, the Minn algorithm, pilot-based carrier frequency synchronization, and the Maximum Likelihood algorithm, to evaluate how effectively they address timing and frequency offset correction. Additionally, the paper proposes mitigating SFO using interpolation filters, which adjust the sampling moments at the receiver post-detection, helping to reduce synchronization errors caused by SFO. Kumar [12] investigated frequency synchronization in frequency-domain OFDM- Index Modulation (IM)-based Wireless Local Area Network (WLAN) systems, suggesting an autocorrelation matrix-based technique to estimate frequency offsets. This approach monitors phase distortions, allowing for more accurate synchronization by compensating for SFO as timing drift accumulates across multiple symbols. By exploiting autocorrelation properties, this method improves synchronization accuracy, potentially boosting overall system performance in WLAN environments.

The authors in [13] proposed a synchronization system for MIMO-OFDM based on Extreme Learning Machines (ELM), structured in two stages. In the first stage, an ELM network was used to estimate the residual symbol timing offset (RSTO) by processing preamble signals that were intentionally corrupted with known timing offsets during training. This was preceded by a preliminary coarse timing synchronization using traditional autocorrelation methods. In the second stage, another ELM network estimated the residual carrier frequency offset (RCFO) by analyzing phase distortions in the received preamble signals, which had also been trained with synthetic frequency offsets. Both ELM modules used the known preamble structure as a reference, processing the received signals after the initial coarse synchronization. The system generated its training data by deliberately introducing controlled timing and frequency offsets to the standard preamble signals, eliminating the need for actual channel conditions during the training phase. In [14], the

authors introduced machine learning based techniques to mitigate channel impairments in OFDM and Non-Orthogonal Multiple Access (NOMA)-OFDM systems, addressing issues like channel distortion, carrier frequency offset, sampling frequency offset, nonlinear distortion (NLD), and frequency-selective fading. The study employed neural networks (NNs), recurrent NNs (RNNs), and long short-term memory (LSTM) models for tasks like channel estimation, equalization, and joint CFO-SFO estimation and

compensation. Moreover, ML techniques were proposed to tackle NLD caused by power amplifier nonlinearities. For NOMA-OFDM systems, the approach replaced traditional Successive Interference Cancellation (SIC)-assisted detection with an ML-driven receiver. These techniques were evaluated through simulations across a range of channel conditions. Table 1 summarizes the previous work on SFO removal.

Table 1 : Survey Summary

Authors	Synchronization Problem	Proposed Technique	Drawbacks / Notes
V. Fischer and A. Kurpiers [5]	Frequency Synchronization	The process consists of two distinct stages: first, frequency acquisition is achieved using FFT-based power spectral density estimation, and second, frequency tracking is performed through phase increment measurement. To account for sample rate offsets, frequency deviation analysis is employed for precise adjustments.	Requires averaging multiple symbols to counteract noise-induced errors.
W.-J. Shin, J. Seo, and Y.-H. You [6]	Sampling Frequency offset	Two algorithmic approaches: (A) Phase differences between adjacent FRCs leveraging temporal correlation, (B) Differential relations among FRC indices for precision. Introduces a low-complexity estimation method.	-
E.-S. Shim, et al. [7]	Timing Offset (TO), CFO, and SFO	Two-stage approach: Pre-FFT synchronization using CP correlation and phase differences; Post-FFT synchronization involving integer frequency offset estimation and reference cell-based tracking.	-
Y.-A. Jung and Y.-H. You [8]	Sampling Frequency offset	Uses temporal correlation between received pilot symbols and frequency correlation among pilot subcarriers to mitigate bias from frequency-selective fading.	-
H. Chuppala, R. Mohan, and R. Shashank [9]	Sampling Frequency offset	Estimates CFO via autocorrelation, then computes SFO through proportionality equations. Corrects SFO-induced errors in the frequency domain with FFT-based phase compensation.	Assumes a shared reference source for carrier and sampling oscillators, which may not always hold.
C.-Y. Hsiao, et al. [10]	Sampling clock offset	Uses phase rotation across OFDM subcarriers in the frequency domain for SFO estimation.	No performance comparison with other methods; potential sensitivity to channel conditions; assumes a free-running ADC.
M. Liu, et al.	Symbol timing, CFO, and	Compares different synchronization	-

[11]	Sampling clock synchronization	algorithms (Schmidl & Cox, Minn, Pilot-based, Maximum Likelihood). Suggests SFO correction via interpolation filters.	
N. Kumar [12]	Frequency synchronization	Autocorrelation matrix-based method for frequency offset estimation and SFO compensation.	-
Liu, Jun, et al. [13]	Timing and Frequency Synchronization in MIMO-OFDM	Employs a dual- ELM framework, where one ELM refines timing estimates and another corrects frequency offsets using preamble signals with synthetic impairments.	-
Singh, Abhiranjan, and Seemanti Saha [14]	Channel Distortions, Synchronization Errors, NLD, Multi-User Interference	Machine Learning (NNs, RNNs, LSTMs) used to address channel distortion, CFO/SFO, and nonlinearity in OFDM/NOMA-OFDM systems.	-

III. PROPOSED METHOD

This section on the proposed method is structured into two parts. Subsection A presents the impact of the SFO on system performance, while subsection B presents both the proposed and conventional methods used to counteract this effect.

A. Analyzing the Influence of SFO on OFDM System

Define $S[k]$ as the frequency-domain symbols, which correspond to the modulated data on the subcarriers, with k ranging from 0 to $N-1$, where N denotes the total number of subcarriers. The corresponding time-domain signal, $s[n]$, is given by:

$$s[n] = \frac{1}{N} \sum_{k=0}^{N-1} S[k] e^{j2\pi kn/N}, \quad n = 0, 1, \dots, N-1 \quad (1)$$

The signal, $r[n]$, at the receiver is influenced by the sampling frequency offset. When there is a discrepancy between the receiver's sampling frequency, f_s' , and the transmitter's sampling frequency, f_s , represented by the difference $\Delta f_s = f_s' - f_s$, the time-domain expression of the received signal can be described as follows:

$$r[n] = s\left(n \cdot \frac{f_s}{f_s'}\right) e^{j2\pi \epsilon n/N} + \eta[n] \quad (2)$$

The normalized sampling frequency offset is denoted by $\epsilon = \Delta f_s / f_s$, while $\eta[n]$ stands for the additive noise. In the receiver's frequency domain, the symbols in the frequency domain are

retrieved as follows:

$$R[k] = \sum_{n=0}^{N-1} r[n] e^{-j2\pi kn/N} \quad (3)$$

Replacing $r[n]$ in the time-domain expression:

$$R[k] = \sum_{n=0}^{N-1} \left(s\left(n \cdot \frac{f_s}{f_s'}\right) e^{j2\pi \epsilon n/N} + \eta[n] \right) e^{-j2\pi kn/N} \quad (4)$$

An alternative way to write this would be:

$$R[k] = \sum_{n=0}^{N-1} s\left(n \cdot \frac{f_s}{f_s'}\right) e^{j2\pi \epsilon n/N} e^{-j2\pi kn/N} + \Xi[k] \quad (5)$$

$\Xi[k]$ represents the frequency-domain representation of the noise, obtained by applying the FFT to $\eta[n]$.

The SFO appears in the frequency spectrum through two primary effects. The first is subcarrier leakage, or inter-carrier interference, which breaks the orthogonality between the subcarriers. This phenomenon can be represented mathematically as:

$$\begin{aligned}
 R[k] &= S[k] \cdot H[k] \cdot \text{sinc}(\pi\epsilon) \\
 &+ \sum_{m \neq k} S[m] \cdot H[m] \cdot \text{sinc}(\pi(\epsilon + m - k)) \\
 &+ \Xi[k]
 \end{aligned} \quad (6)$$

$$\theta[k] = 2\pi\epsilon \cdot \frac{k}{N} \quad (7)$$

In this context, $H[k]$ denotes the frequency response of the channel, while the sinc function accounts for the interference caused by adjacent subcarriers. Second, SFO induces a phase shift that increases linearly as the subcarrier index k progresses:

Accurate phase rotation compensation is essential for proper demodulation of the received signal. The presence of phase rotation and frequency offset degrades system performance, emphasizing the critical role of precise SFO estimation and correction in OFDM systems. The next figure demonstrates the impact of SFO on the performance of a 16QAM-OFDM system through simulation.

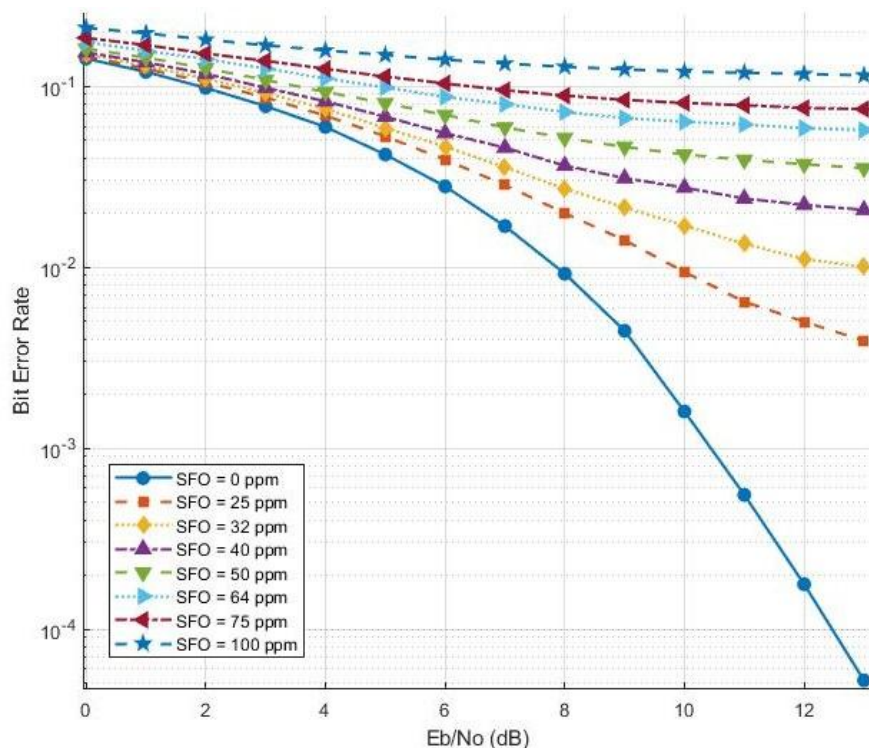


Fig. 1: Effect of SFO on the BER in a 16QAM-OFDM Communication System

As expected, Figure 1—generated in MATLAB—shows the characteristic BER performance of a 16QAM-OFDM system with 32 subcarriers and 150 OFDM symbols under varying sampling frequency offsets. The figure illustrates that increasing SFO introduces performance degradation, reflected in rising BER values even at high E_b/N_0 levels. Larger offsets yield progressively flatter BER curves, deviating from the ideal SFO = 0 ppm case and reaffirming the well-known sensitivity of OFDM systems to

frequency mismatches.

B. Methods for Counteracting SFO Effects: Conventional-Hybrid vs. Proposed Approach

This subsection is structured into five parts. The first four introduce benchmark methods for SFO estimation, while the fifth presents the proposed approach. A summary of all methods is provided as follows:

B.1. Phase Difference (PD) Method

This technique, based on the SFO estimation method outlined in [5], calculates the SFO by examining phase shifts across subcarriers

in the received signal's frequency domain. Given that SFO causes phase rotations that vary with subcarrier indices, a reference subcarrier is chosen, and the phase differences relative to it are measured to determine the frequency offset. The phase difference between subcarrier m and the reference subcarrier ref is given by:

$$\theta_m = \angle R[m] - \angle R[ref] \quad (8)$$

The phase of the received symbol at subcarrier m is denoted as $\angle R[m]$.

The SFO value is subsequently calculated as:

$$\hat{\epsilon}_1 = \frac{1}{2\pi N} \sum_{\substack{m=2 \\ m \neq ref}}^N \frac{\theta_m}{m - ref} \quad (9)$$

where N is the number of subcarriers. To enhance the accuracy of the estimation, a weighting scheme is applied to modify this process.

B.2. Correlation-Based (CB) Method

The correlation-based technique, which is based on the SFO estimation method outlined in [7], calculates SFO by utilizing the phase connection between consecutive subcarriers in the received OFDM signal. As the phase distortion induced by SFO builds up progressively across the subcarriers, a detectable phase correlation emerges between neighboring subcarriers. This phase correlation between adjacent subcarriers is expressed as follows:

$$\theta_m = \angle(R[m]R^*[m-1]) \quad (10)$$

The term $R^*[m-1]$ represents the complex conjugate of $R[m-1]$, while θ_m denotes the phase difference between neighboring subcarriers. When averaged across all subcarriers, the result is:

$$\hat{\epsilon}_2 = \frac{1}{2\pi N} \cdot \frac{1}{N-3} \sum_{m=3}^N \theta_m \quad (11)$$

Where $\hat{\epsilon}_2$ is the estimated SFO.

B.3. Phase Difference Weighted by Subcarrier Index (PD-WSI) method

This technique, based on the SFO estimation methods outlined in [9,15,16], calculates the SFO by examining the phase shift between consecutive OFDM symbols. Unlike traditional methods that rely solely on the phase difference, this approach incorporates a weighting scheme tied to the subcarrier index. The rationale behind this weighting is that subcarriers with higher indices undergo a greater phase rotation caused by SFO, enhancing the accuracy of the

estimate. The estimation process is outlined below:

$$\begin{aligned} \Delta\theta_l[m] &= \angle \left(\frac{R_l[m+1]}{R_l[m]} \right) \\ \xi_3 &= \sum_{l=-\frac{N}{2}}^{\frac{N}{2}-1} l \cdot \frac{1}{N-1} \sum_{m=1}^{N-1} \Delta\theta_l[m] \\ \hat{\epsilon}_3 &= \xi_3 * C_3 \end{aligned} \quad (12)$$

Where $R_l[m]$ is the received OFDM symbol at subcarrier l and symbol m in the frequency domain (assuming symbols equal subcarriers in count). The term $\Delta\theta_l[m]$ indicates the phase difference between consecutive symbols at subcarrier l , while ξ_3 is the weighted total of these phase differences. C_3 refers to the scaling factor, and $\hat{\epsilon}_3$ denotes the estimated SFO.

B.4. Hybrid Estimation (H-EST) Method

The Hybrid SFO Estimation approach is designed to accurately estimate the Sampling Frequency Offset in OFDM systems by leveraging both time-domain and frequency-domain correlations across all subcarriers. This method builds directly from the estimation approach introduced in [9]. The resulting estimation for the SFO is expressed as follows:

$$\begin{aligned} \xi_4 &= \sum_{l=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{m=1}^{N-1} \left(\angle \left(\frac{R_l[m+1]}{R_l[m]} \right) \right. \\ &\quad \left. + k_l \cdot \angle \left(\frac{R_l[m+1]}{R_l[m]} \right) \right) \\ \hat{\epsilon}_4 &= \xi_4 * C_4 \end{aligned} \quad (13)$$

The index k_l identifies the frequency bin associated with subcarrier l (assuming symbols equal subcarriers in count), while ξ_4 captures the phase shift resulting from both time and frequency correlation. The term C_4 serves as a scaling parameter, and $\hat{\epsilon}_4$ represents the estimated SFO.

B.5. Proposed Classifier-Based SFO Estimators

In our proposed method, we chose a classifier-based framework to estimate the sampling frequency offset. This approach stands apart from traditional techniques that typically use

closed-form solutions or iterative processes, as it utilizes optimized feature mappings to better capture nonlinear patterns, ultimately improving estimation accuracy and robustness. To further increase the precision and reliability, we incorporated several classifiers, a common practice in machine learning to identify the most suitable solution. Additionally, we focused on achieving the best performance while minimizing computational complexity. As such, we avoided resource-heavy models like Convolutional Neural Networks (CNNs). This led us to introduce three distinct SFO estimators, each named according to the classifier used: Kernel Support Vector Machine (KSVM)-based, LDA-based, and ANN-based SFO estimators.

This subsection is organized into three parts: subsection B.5.1. introduces the classifiers used in the proposed SFO estimators, subsection B.5.2. describes the dimensionality reduction technique applied, and subsection B.5.3. presents the training set used for the classifier-based estimators.

B.5.1. Classifiers-used in the SFO estimators

The classifiers are described in detail as follows:

B.5.1.1. Linear discriminant analysis

LDA classification involves building a discriminant analysis model that predicts outcomes by considering factors like posterior probability, prior probability, and associated costs. The primary objective is to assign observations to the correct class while reducing the overall classification cost. This optimization process can be accomplished by [17,23,24]:

$$\hat{y} = \arg \min_{y=1, \dots, K} \sum_{k=1}^K \hat{P}(k|x) C(y|k) \quad (14)$$

In this context, K represents the total number of classes, while $\hat{y}$ denotes the predicted classification. The term $C(y|k)$ signifies the cost associated with assigning an observation to class y when its actual classification is k . Meanwhile, $\hat{P}(k|x)$ refers to the posterior probability that an observation belongs to class k .

Our approach employs LDA to estimate

the SFO. The estimation framework relies on LDA to classify specific features—detailed at subsection B.5.3.—into predefined classes, each corresponding to a distinct SFO value. A crucial aspect of this process is dimensionality reduction, which helps minimize computational complexity while preserving essential SFO characteristics for optimal training and classification. Additionally, feature selection ensures that the retained features effectively capture the SFO behavior in the signal. Notably, the dimensionality reduction and feature selection techniques applied remain consistent across all ML-based estimators, ensuring uniformity in feature representation and facilitating fair performance comparisons. To maintain numerical stability and ensure accurate LDA classification, we apply the pseudo-inverse of the covariance matrix, effectively mitigating issues arising from ill-conditioned matrices. The model undergoes training across multiple frames, allowing it to adapt dynamically to fluctuating channel conditions for improved robustness. Once classification is complete, the resulting outputs are mapped back to their respective SFO values using a predefined lookup table, thus completing the estimation process.

B.5.1.2. Kernel support vector machine

A kernel support vector machine extends the capabilities of a linear SVM by applying kernel functions, which transform non-linear data into a higher-dimensional space where it becomes linearly separable. This transformation enables the algorithm to construct an optimal hyperplane that maximizes the margin between classes, even in cases where the data exhibits complex, non-linear patterns. The decision function of an SVM is formulated as [18,23,24]:

$$f(x) = \text{sign} \left(\sum_{i=1}^N \alpha_i y_i K(x_i, x) + b \right) \quad (15)$$

In this context, y_i denotes the class labels associated with the support vectors x_i while, α_i represents the Lagrange multipliers. The term b corresponds to the bias, and the function $K(x_i, x)$ serves as the kernel function. Specifically, our method employs the Radial Basis Function (RBF) kernel, which is expressed as [21]:

$$K(x_i, x) = \exp(-\gamma \|x_i - x\|^2) \quad (16)$$

In this context, γ is a positive kernel parameter

that determines how much impact a single training instance has on the model. A higher γ results in decision boundaries that are tightly shaped around individual data points, leading to more precise but potentially overfitted classifications. Conversely, lower γ values produce broader, more generalized decision regions. The RBF kernel enhances SVMs by mapping data into a higher-dimensional space, enabling them to capture complex, non-linear patterns and achieve linear separability in that transformed space.

Here, we apply the KSVM to predict the SFO. To capture the complex relationships between features and their classes, we use the RBF kernel, which creates nonlinear decision boundaries. The KSVM categorizes the selected features, outlined at subsection B.5.3., into classes that each represent a specific SFO range, which is mapped in a table. The model is trained across multiple frames to adapt to varying channel conditions, ensuring reliable performance. After classification, the results are mapped to the corresponding SFO values using a predefined table, completing the estimation process.

B.5.1.3. Artificial neural network

The Multilayer Perceptron (MLP) is employed to predict SFO based on the received signals. It consists of an input layer, one or more hidden layers, and an output layer, with the goal of classifying the signals using the Backpropagation method. During forward propagation, the input is passed through the network, generating the output. In backward propagation, the error is computed and the network's weights are adjusted to minimize the error. Key elements like activation functions and weights are essential for mapping the input to the output. The training of the network involves iterating through forward and backward propagation cycles to fine-tune the weights and reduce the prediction error.

The output of the j th neuron in the $(k+1)$ layer is determined by a weighted sum of the outputs from the neurons in the k th layer, followed by an activation function. The activation function, denoted as R , processes the sum of the weighted inputs. The parameters involved include M_k , which represents the number of neurons in the k th

layer, $W_{ij}^{(k+1)}$, the weight connection between the i th neuron in the k th layer and the j th neuron in the $(k+1)$ layer, and x_i^k , the output of the i th neuron in the k th layer. This relationship is expressed as [19,23,24]:

$$x_j^{(k+1)} = R \left(\sum_{i=0}^{M_k} W_{ij}^{(k+1)} x_i^k \right) \quad (17)$$

The error function is described by [17]:

$$E = \frac{1}{2} \sum_{j=0}^{M_k} (x_j^k - d_j^k)^2 \quad (18)$$

In the k th layer, the desired output of the j th neuron is given by d_j^k , while its actual output is x_j^k . During the training phase, the Bayesian regularization back-propagation algorithm is applied, specifically leveraging the Levenberg-Marquardt optimization method [20] to update the weights and biases. The goal of this process is to fine-tune the weights and minimize the error between predicted and actual outputs. The Levenberg-Marquardt optimization is a popular method in numerical optimization, as it combines the advantages of both the Gauss-Newton and gradient descent methods, thus proving effective in minimizing nonlinear objective functions.

Our approach uses an MLP neural network to predict the SFO. The network is designed with multiple hidden layers to model the complex relationships between signal features, which include the received signal characteristics subjected to the effect of both SFO impairment and the channel, and the corresponding SFO values. During training, the model adjusts its weights iteratively through optimization to map inputs to accurate SFO predictions. Training continues over several epochs, enabling the model to learn the data's underlying patterns. Once trained, the MLP predicts the SFO, completing the estimation process.

B.5.2. Dimensionality Reduction Techniques

We utilized two matrix dimensionality reduction techniques: principal component analysis and singular value decomposition. The techniques are illustrated as follows:

B.5.2.1. Principal component analysis

PCA is a mathematical technique used to

reduce the dimensionality of a high-dimensional dataset by projecting it onto a lower-dimensional subspace. This is achieved by determining a new set of orthogonal basis vectors, known as principal components, which capture the directions of maximum variance in the data. The transformation preserves the most critical structural information while eliminating less important dimensions [21].

The steps we followed to apply PCA are outlined as follows:

Step 1: Centering the input

PCA requires the input to be zero-mean. This step subtracts the mean of each row (row-wise) from the input.

$$\tilde{X} = X - 1\mu \quad (19)$$

Where X is the original input, μ is the mean along each row, and $\tilde{X}$ is the centered X .

Step 2: Compute Covariance Matrix

The covariance matrix is computed as:

$$C = \tilde{X}\tilde{X}^H \quad (20)$$

Where $\tilde{X}^H$ is the conjugate transpose (Hermitian) of $\tilde{X}$, and C is the covariance matrix.

Step 3: Eigen Decomposition

Eigen decomposition extracts the principal components (eigenvectors) corresponding to the largest eigenvalues as:

$$CV = V\Lambda \quad (21)$$

Where V contains the eigenvectors (principal components), and Λ is a diagonal matrix of eigenvalues.

Step 4: Project input onto Principal Components

Transforms the original input matrix into the lower-dimensional space defined by the principal components.

$$Z = V^H X \quad (22)$$

Where V^H is the conjugate transpose of the eigenvectors,

and Z is the reduced output matrix.

Note the input matrix here is the frequency domain signal after being affected by certain SFO and the channel effect.

B.5.2.2. Singular value decomposition

SVD is a matrix factorization method that decomposes a matrix $X \in \mathbb{R}^{n \times d}$ into three components [22]:

$$X = U\Sigma V^T \quad (23)$$

Where $U \in \mathbb{R}^{n \times n}$ is the orthogonal matrix (left singular vectors), $\Sigma \in \mathbb{R}^{n \times d}$ is the rectangular diagonal matrix of singular values $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$, and $V \in \mathbb{R}^{d \times d}$ is the orthogonal matrix (right singular vectors). The singular values σ_i are the square roots of the eigenvalues of $X^T X$ or XX^T .

To approximate X with rank k , we use:

$$X_k = U_k \Sigma_k V_k^T \quad (24)$$

Where $U_k \in \mathbb{R}^{n \times k}$, $\Sigma_k \in \mathbb{R}^{k \times k}$, and $V_k \in \mathbb{R}^{d \times k}$.

The process we used is defined in the following steps:

Step 1: Center the input

Subtract the mean from each feature (row-wise) to center the input around zero. Essential for both PCA and SVD to focus on variance rather than bias.

$$\tilde{X} = X - \mathbb{E}[X] \quad (25)$$

Where X is the original input matrix, and $\mathbb{E}[X]$ is the mean along rows.

Step 2: Compute SVD

Factorizes the centered matrix into singular vectors and values, then computes a reduced SVD.

$$\tilde{X} = U\Sigma V^H \quad (26)$$

Where U is the left singular vectors, S is the diagonal matrix of singular values, and V is the right singular vectors.

Step 3: Truncate to “target dimensions” Components

Retains only the top k singular vectors/values (most significant

directions) and discards smaller singular values.

$$U_k = U[:, 1:k], S_k = S[1:k, 1:k] \quad (27)$$

Where k is the target dimensions.

Step 4: Project input onto

Reduced Basis

Projects the original (uncentered) input onto the top k singular vectors.

$$Z = U_k^H X \quad (28)$$

Where Z is the reduced matrix, and U_k^H is the conjugate transpose of the top k singular vectors.

B.5.3. Construction of the Training Set for Machine Learning-based SFO Estimators

To enable data-driven estimation of the SFO, we construct a training set that captures the relationship between received OFDM signals (affected by SFO and channel then reduced) and their true SFO values. Below, we formalize the mathematical representation of the inputs (features) and targets (labels) used for training the proposed ML-based estimators.

• Input Features

For each sample i , the input is constructed as:

$$X_i = \left[\text{Re}(Z_i), \text{Im}(Z_i), \frac{|Z_i|}{\max(|Z_i|)}, \frac{\angle(Z_i)}{\pi} \right] \quad (29)$$

Where X_i is the input feature vector, and Z_i is the dimensionality-reduced output matrix.

• Targets (True SFO Values)

The targets are the true SFO values (in ppm) for each sample:

$$\epsilon = \{t_i\}_{i=1}^{S_{\text{samples}}} \quad (30)$$

where t_i is repeated for each channel realization. The input feature set and corresponding target values are used to train the models, enabling them to

predict ϵ for new inputs, with a table mapping classes to their respective SFO values in KSVM and LDA.

Our approach to feature-target training is highly efficient because it reduces computational complexity while preserving the key characteristics essential for accurately estimating the SFO. By focusing on the most relevant features and minimizing unnecessary dimensionality, we maintain the critical information needed to model both amplitude and phase distortions. This enables us to achieve high estimation accuracy while ensuring the process remains computationally efficient.

IV. RESULTS AND DISCUSSION

The MATLAB-based implementation of the proposed estimation methods and simulation setup is discussed in this section, which is organized into two subsections. In subsection A, the methodology for evaluating various SFO estimation approaches is described. Subsection B provides a detailed comparison of the results obtained from each method under the same test conditions.

A. Performance Assessment Framework

All techniques were evaluated using Monte Carlo testing, supplemented by BER curve analysis for comparative assessment. The Monte Carlo testing procedure is outlined as follows:

1. Configuration Phase

- Total simulation trials: $M = 10,000$
- Error tracking variables are zero-initialized for each estimator: PD, CB, PD-WSI, H-EST, LDA-based, KSVM-based, and ANN-based SFO estimators

$$\sum RMSE_m = 0 \quad (31)$$

$m \in \{\text{PD, CB, PD-WSI, H-EST, LDA-based, KSVM-based, and ANN-based SFO estimators}\}$

2. Trial Execution Loop

For every trial i ($1 \leq i \leq M$):

a) *Parameter Sampling*

- Randomly draw true SFO value $\epsilon^{(i)}$ from a uniform distribution across the operational range.

$$\epsilon^{(i)} \sim P(\epsilon) \quad (32)$$

b) Signal Processing

- Apply the selected SFO to an OFDM distorted signal (by the channel)
- Transform the signal to frequency domain representation

3. Estimation Phase

- Each algorithm m computes its SFO estimate $\hat{\epsilon}^{(i)}$

4. Error Quantification

a) Cumulative Error Tracking

- For every estimator m :

$$\sum RMSE_m = \sum_{i=1}^M (\epsilon^{(i)} - \hat{\epsilon}_m^{(i)})^2 \quad (33)$$

b) Final Metric Computation

- Post-simulation, root-mean-square error is derived as:

$$\sum RMSE_m = \sqrt{\frac{1}{M} \sum_{i=1}^M (\epsilon^{(i)} - \hat{\epsilon}_m^{(i)})^2} \quad (34)$$

- The **percentage improvement (reduction) in RMSE** is calculated as:

$$\% \Delta RMSE = \left[\frac{Final RMSE_{baseline method} - Final RMSE_{improved method}}{Final RMSE_{baseline method}} \right] \times 100 \quad (35)$$

The configuration of all hyperparameters employed in the proposed method is outlined in Table 2.

Table 2: The Hyperparameters Table

Name	Hidden Layers and Activation Functions	Features and Classes	Back-propagation and training	Additional Notes
LDA-based SFO estimator	N/A	➤ Features: The features and targets for training the estimator are: $X_i = \left[Re(Z_i), Im(Z_i), \frac{ Z_i }{\max(Z_i)}, \frac{\angle(Z_i)}{\pi} \right]$ $\epsilon = \{t_i\}_{i=1}^{S_{samples}}$ ➤ Classes: 100 Class (SFO=0:100ppm). A lookup table is used to set the targets to a class sets, then this table is saved, and used when the model is applied.	Trained on 1:25 frames (depending on the channel)	Pseudo-inverse for covariance matrix inversion.
KSVM-based SFO estimator	N/A	• Features: The features and targets for training the estimator are: $X_i = \left[Re(Z_i), Im(Z_i), \frac{ Z_i }{\max(Z_i)}, \frac{\angle(Z_i)}{\pi} \right]$ $\epsilon = \{t_i\}_{i=1}^{S_{samples}}$ ➤ Classes: 100 Class (SFO=0:100ppm).	Trained on 1:25 frames (depending on the channel)	RBF kernel.

A lookup table is used to set the targets to a class sets, then this table is saved, and used when the model is applied.

ANN-based SFO estimator	• 3 Hidden Layers. - Number of neurons in each layer $h_l = \left\lceil \frac{n}{2^{l-1}} \right\rceil + 1, l \geq 1$ - Where n is the number of subcarriers, l represents the hidden layer index. (e.g., with three hidden layers and 32 subcarriers, the neuron distribution is: 33, 17, and 9 neurons, respectively) - Activation Functions= Hyperbolic Tangent Sigmoid (Hidden), Linear (Output)	➤ Features: The features and targets for training the estimator are: $X_i = \left[Re(Z_i), Im(Z_i), \frac{ Z_i }{\max(Z_i)}, \frac{\angle(Z_i)}{\pi} \right]$ $\epsilon = \{t_i\}_{i=1}^{S_{samples}}$ ➤ Classes: • 100 Class (SFO=0:100ppm).	• Levenberg-Marquardt with Bayesian regularization • Trained on 1:25 frames (depending on the channel).
--------------------------------	---	---	--

B. Sampling frequency offset estimation result

Figure 2 illustrates the Monte Carlo RMSE performance of the SFO estimation methods under different SNR conditions. It is evident that machine learning-based methods, particularly the LDA-based and ANN-based estimators, achieve significantly lower RMSE values compared to traditional approaches, demonstrating superior estimation accuracy and robustness to noise. Among all the methods, the LDA-based estimator consistently performs the best across all SNR levels. In contrast, traditional techniques like PD and CB exhibit the highest RMSE, indicating poor performance, especially in low SNR scenarios. Hybrid methods such as PD-WSI and H-EST show moderate performance, offering a compromise between complexity and accuracy. Overall, the results highlight the effectiveness of ML-based approaches in SFO estimation,

especially under challenging noise conditions. Figure 3, 4, and 5 show the Bit Error Rate performance of the SFO methods as a function of Eb/No (energy per bit to noise power spectral density ratio). Among all the methods evaluated, the ANN-based, LDA-based, and KSVM-based SFO estimators demonstrate performance that closely matches the Ideal case, indicating excellent estimation and compensation capability with minimal BER degradation. In contrast, traditional methods like PD and CB show significantly higher BER, suggesting less effective handling of SFO. The PD-WSI and H-EST methods perform moderately better than PD and CB but still fall short of the machine learning-based estimators. Overall, the three plots clearly highlight that ML-based approaches substantially improve BER performance, making them more reliable for robust communication.

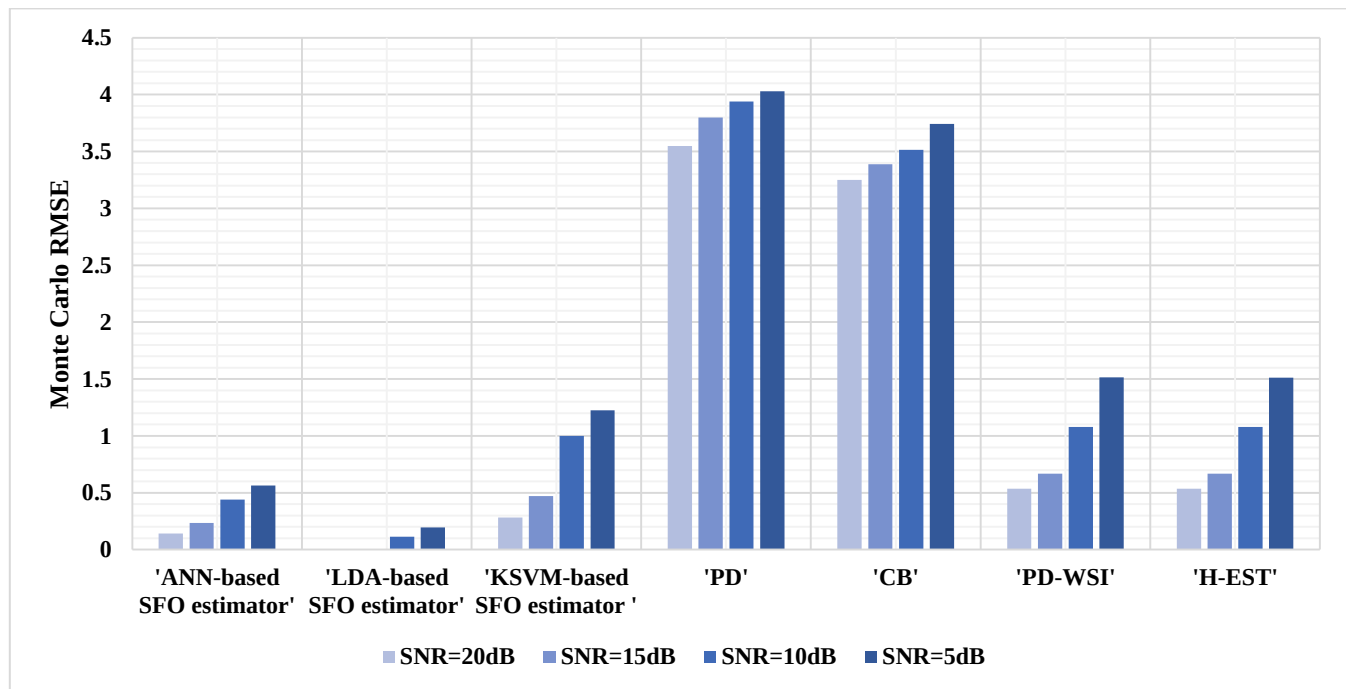


Fig. 2: Comparative Analysis of Monte Carlo RMSE Under AWGN Conditions

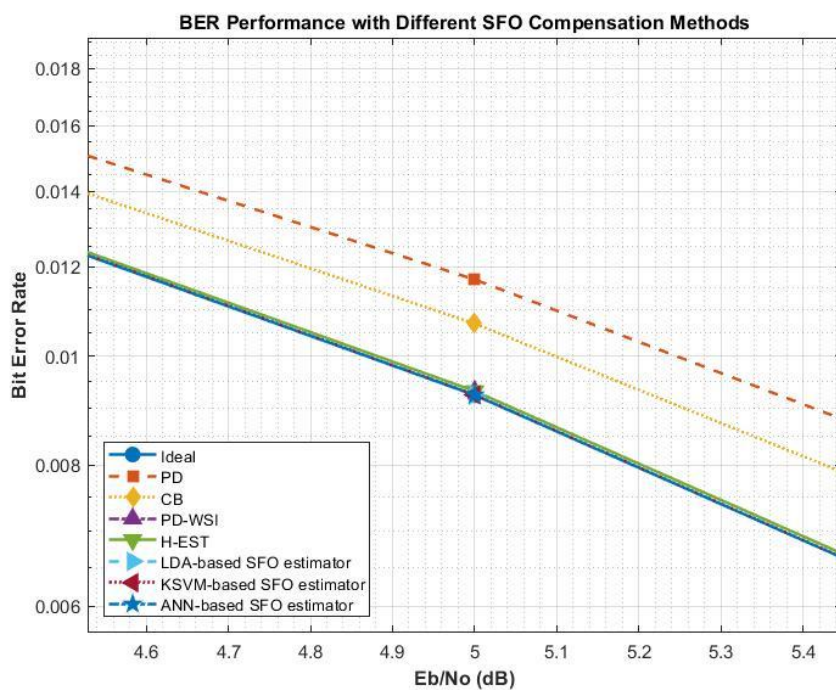


Fig.3: Comparison of SFO method error rates around a 5 dB signal-to-noise ratio

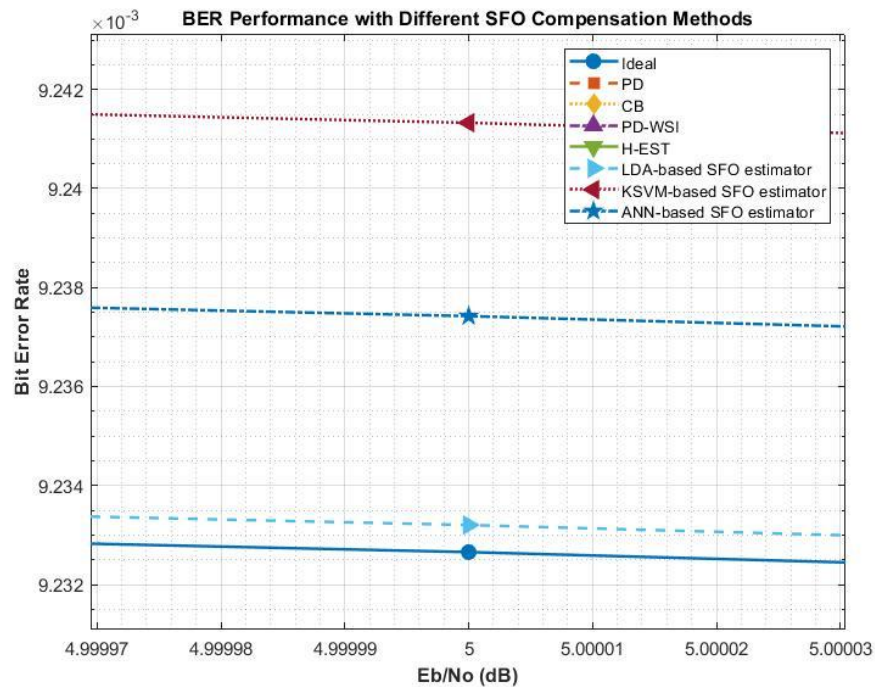


Fig. 4: Zoomed perspective on BER behavior of SFO approaches around 5 dB signal-to-noise ratio

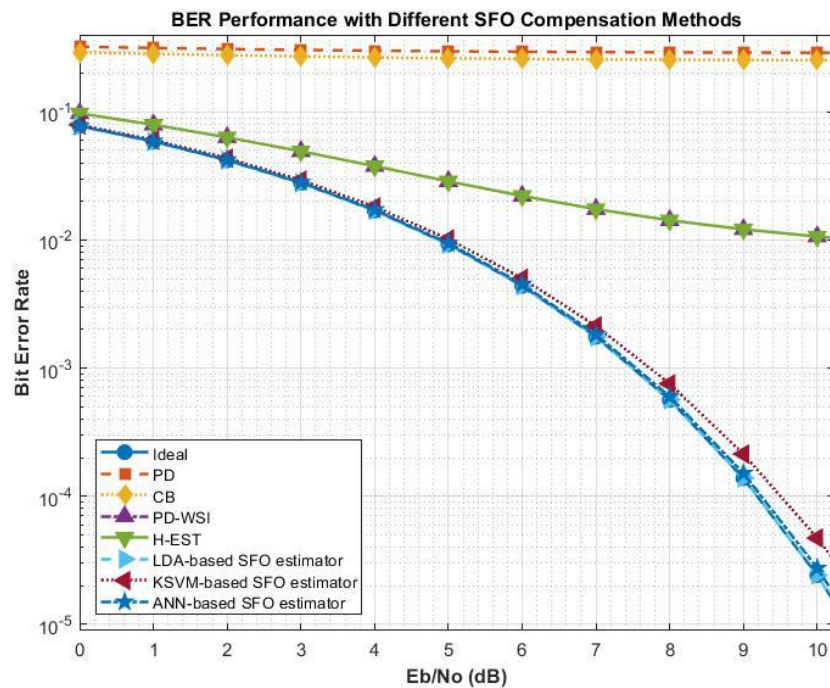


Fig.5: Assessment of Bit Error Rates Across Different Eb/No Levels for SFO Techniques

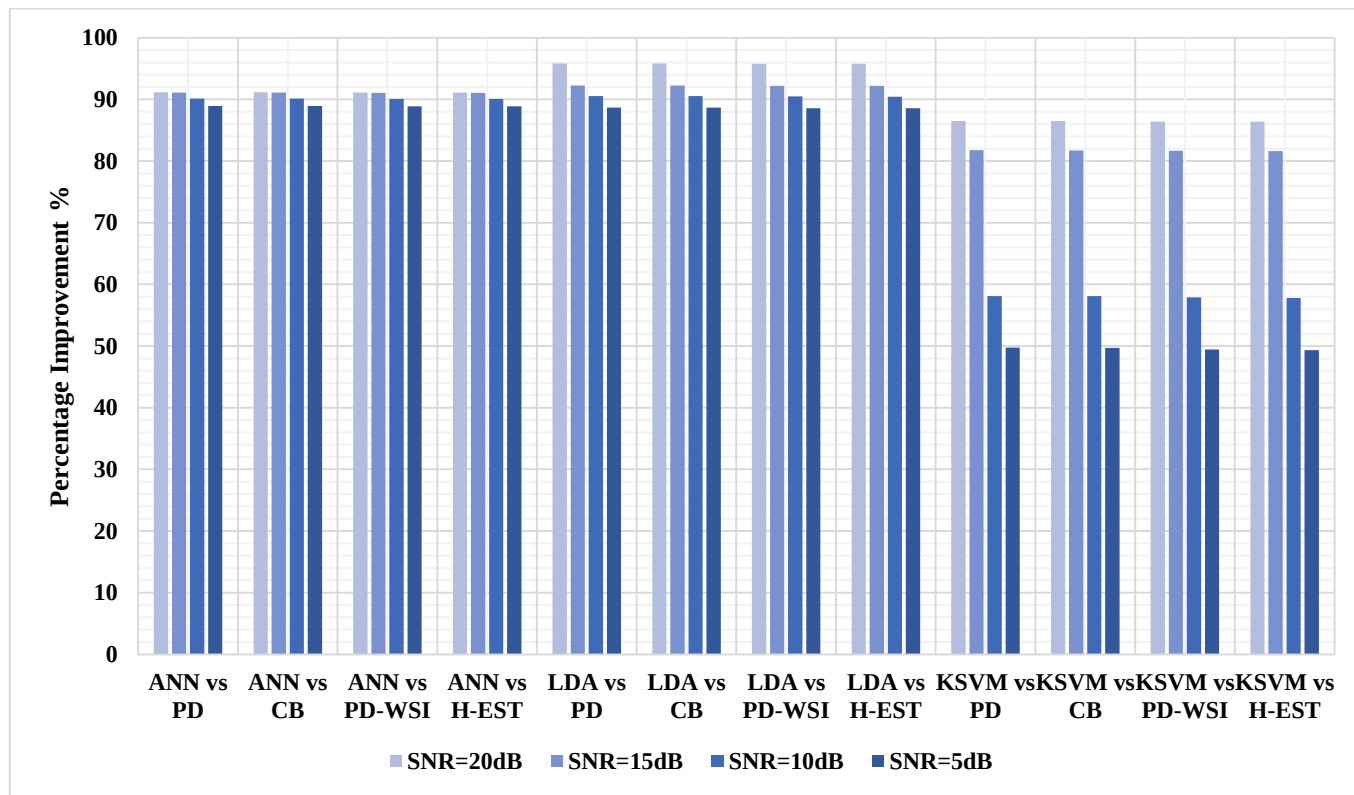


Fig. 6: Performance Enhancement of Machine Learning Models Over Baselines in Fading Channel 1

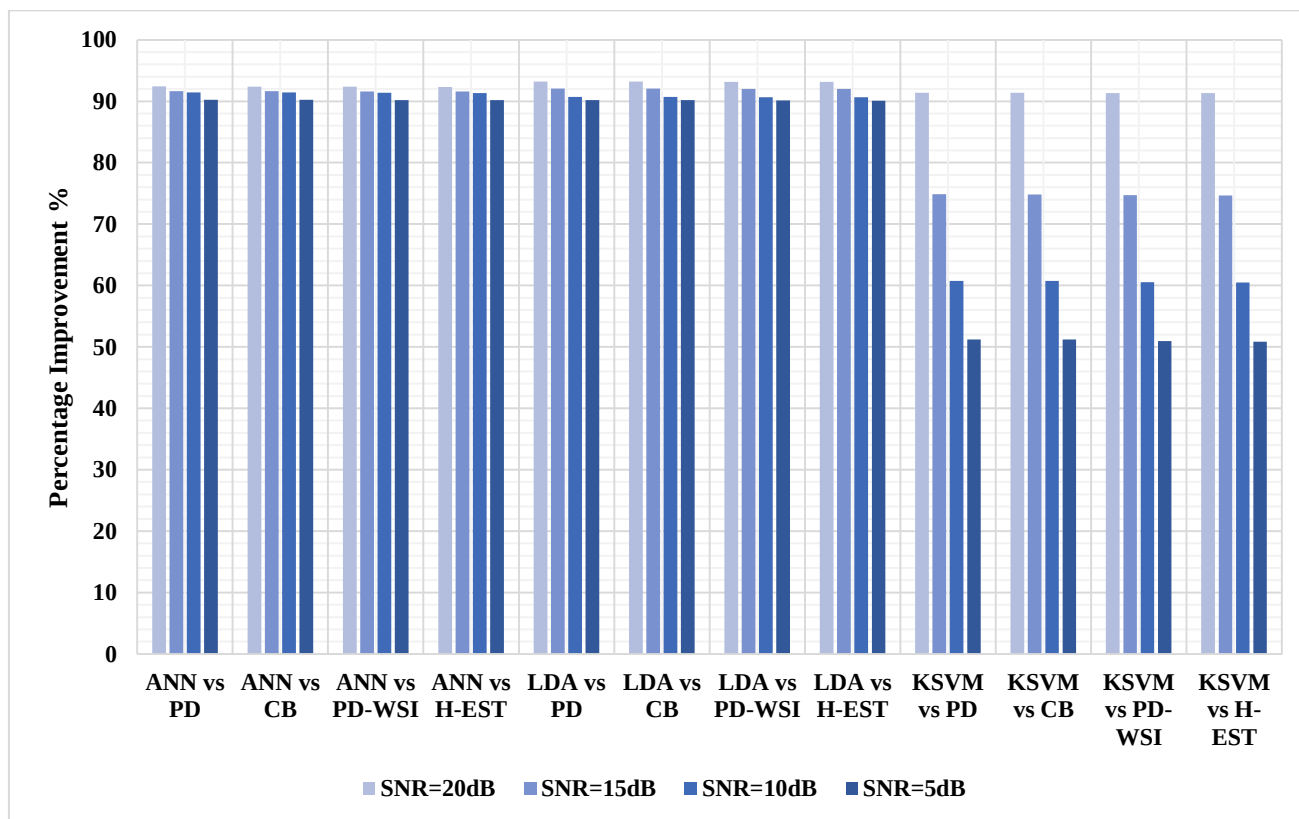
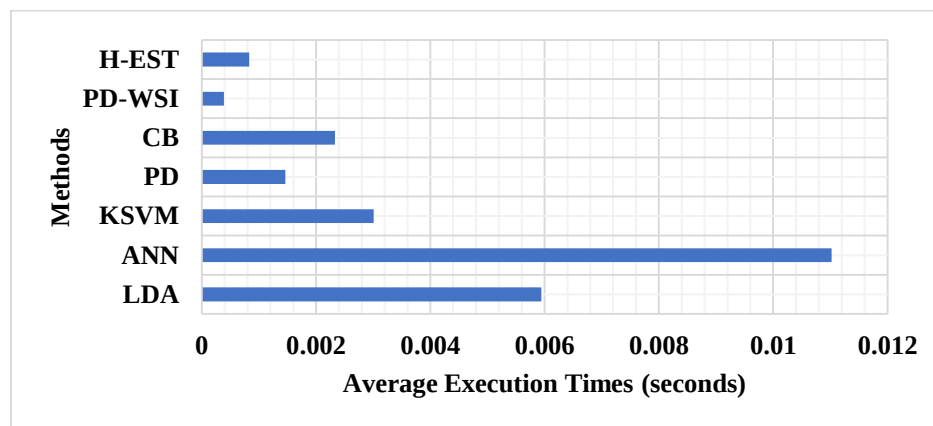


Fig.7: Performance Enhancement of Machine Learning Models Over Baselines in Fading Channel 2

Table 3: Simulation Parameters for Flat and Frequency-Selective Rayleigh Fading Channels

Parameters (Normalized System)	Frequency-Selective Rayleigh Fading Channel 1	Frequency-Selective Rayleigh Fading Channel 2
Doppler Shift ($\times R_s$) Hz	1	3
Number of Multipath Components	5	8
Path Delays (sec)	[0,0.3Ts,0.7Ts,1.5Ts,2Ts]	[0,0.2Ts,0.5Ts,0.9Ts,1.3Ts,1.8Ts,2.5Ts,3Ts]
Average path gains (dB)	[0,-3,-6,-9,-12]	[0,-4,-8,-12,-16,-20,-25,-30]

**Fig. 8: Average execution times when the training is executed offline**

Figures 6 and 7 illustrate the percentage improvement in SFO estimation under Fading Channels 1 and 2, respectively, as outlined in Table 3, using the methods evaluated across SNR levels of 20 dB, 15 dB, 10 dB, and 5 dB. Both LDA-based and ANN-based estimators achieve high accuracy, consistently delivering over 88% improvement across all baselines (PD, CB, PD-WSI, and H-EST). The LDA-based SFO estimator shows slightly better performance than the ANN at higher SNRs, particularly under clean channel conditions. KSVM-based SFO estimator, although effective at higher SNRs, show a marked decline in improvement under fading conditions, dropping to around 50% against the PD baseline at 5 dB, indicating sensitivity to channel degradation. Overall, while LDA may offer a slight edge in optimal conditions, ANN stands out for its consistent reliability across a range of SNRs, outperforming conventional and hybrid techniques in challenging fading channels. Both SVD and PCA are effective dimensionality reduction techniques that preserve the essential variation

needed for accurate SFO estimation, while significantly reducing processing time and computational cost. PCA demonstrated superior average performance, showing an improvement of approximately 37.91%, whereas SVD achieved the lowest average processing time, reducing it by about 96.14%. PCA is particularly effective at retaining critical variance for higher estimation accuracy, while SVD provides substantial computational savings, especially in complex fading environments. However, under favorable channel conditions, both methods perform similarly in terms of performance, making SVD a suitable option when the channel is stable and less noisy. Figure 8 reports the inference latency of various SFO estimators, averaged over 10,000 runs on a system with an AMD Ryzen 5 5600G (3.90 GHz) and 16GB RAM, excluding training time. The PD, CB, PD-WSI, H-EST, and KSVM-based methods demonstrate low execution times, whereas the LDA- and ANN-based estimators are notably slower, with the ANN-based method exhibiting the highest latency. From figure 8 we

can understand that, Although KSVM does not match the high estimation accuracy of ANN and LDA, it delivers more reliable performance than both hybrid and conventional methods in challenging environments with significantly lower computational time than ANN and LDA. In real-world applications, selecting an appropriate estimator involves navigating trade-offs between accuracy, computational overhead, and processing speed. The optimal choice depends heavily on the specific use case—systems that require precise estimation may tolerate higher computational demands, whereas environments with limited processing power or energy availability favor lightweight, efficient solutions. Ultimately, the estimator must be chosen in accordance with the system's operational constraints and performance goals.

V. CONCLUSION

This study comprehensively evaluated various sampling Frequency Offset estimation methods under diverse SNR conditions and channel environments. The results consistently demonstrate the superiority of machine learning-based approaches—particularly LDA-based and ANN-based estimators—over traditional and hybrid techniques. These ML methods achieved significantly lower RMSE and BER values, indicating enhanced estimation accuracy and robustness to noise and fading effects. Among them, the LDA-based estimator delivered the best overall performance, especially at higher SNRs, while the ANN-based approach showed remarkable consistency across all tested conditions. The KSVM estimator also showed promising results at higher SNRs but was more sensitive to degradation under fading channels. In contrast, traditional methods such as PD and CB were significantly less effective, especially in low-SNR and fading environments. Hybrid methods like PD-WSI and H-EST offered a balanced performance but did not match the reliability and accuracy of ML-based solutions. Dimensionality reduction techniques, namely PCA and SVD, played a crucial role in enhancing the computational efficiency of the ML models. While PCA stands out for its enhanced estimation accuracy—delivering an average improvement of around 37.91%—SVD takes the lead in efficiency,

cutting average processing time by roughly 96.14%. Nevertheless, when channel conditions are stable and relatively free of noise, the performance gap between the two narrows considerably, allowing for flexible method selection based on system priorities rather than performance differences. In conclusion, ML-based SFO estimators, supported by effective dimensionality reduction techniques, provide a powerful solution for robust and accurate synchronization in modern communication systems. The choice of estimator should ultimately align with the specific system requirements, balancing accuracy, latency, and computational resource availability.

REFERENCES

- [1] de Oliveira, Lucas Giroto, et al. "Pilot-Based SFO Estimation for Bistatic Integrated Sensing and Communication." *IEEE Transactions on Microwave Theory and Techniques* (2024).
- [2] Kim, Haesik. "A Hybrid Frequency Offset Estimation Combining Data-Driven Method and Model-Driven Method for 6G OFDMA Systems." *2024 IEEE Annual Congress on Artificial Intelligence of Things (AIoT)*. IEEE, 2024.
- [3] Yoon, Eunchul, et al. "Doppler spread estimation based on machine learning for an OFDM system." *Wireless Communications and Mobile Computing* 2021.1 (2021): 5586029.
- [4] Hu, Qiang, et al. "Deep learning for channel estimation: Interpretation, performance, and comparison." *IEEE Transactions on Wireless Communications* 20.4 (2020): 2398-2412.
- [5] Fischer, Volker, and Alexander Kurpiers. "Frequency synchronization strategy for a PC-based DRM receiver." *7th International OFDM-Workshop (InOWo'02), Hamburg*. 2002.
- [6] Shim, Eu-Suk, et al. "Synchronization receiver for OFDM-based FM broadcasting systems." *Wireless Personal Communications* 58 (2011): 355-367.
- [7] Shin, Won-Jae, Jeongwook Seo, and Young-Hwan You. "MSE analysis of sampling frequency offset estimation scheme for OFDM-based digital radio mondiale (DRM) systems." *Wireless personal communications* 71 (2013): 1271-128.
- [8] H. Chuppala, R. Mohan, and R. Shashank, "Sampling clock offset estimation and correction in frequency domain for OFDM receivers," in *TENCON 2017 - 2017 IEEE Region 10 Conference*, IEEE, 2017.
- [9] Jung, Yong-An, and Young-Hwan You. "Robust sampling frequency offset estimation for OFDM over frequency selective fading channels." *Applied Sciences* 8.2 (2018): 197.
- [10] Liu, Mengyang, et al. "Synchronization Simulation and Analysis of OFDM Wireless Communication Technology." *2019 IEEE 4th International Conference*

- on Computer and Communication Systems (ICCCS). IEEE, 2019.
- [11] Hsiao, Chien-Yao, et al. "A clock synchronization technique for OFDM-based communications." *2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan)*. IEEE, 2020.
- [12] Kumar, Navin. "Frequency Synchronization in Frequency Domain OFDM-IM based WLAN Systems." *Journal of Communications Software and Systems* 19.3 (2023): 199-206.
- [13] Liu, Jun, et al. "Fine timing and frequency synchronization for MIMO-OFDM: An extreme learning approach." *IEEE Transactions on Cognitive Communications and Networking* 8.2 (2021): 720-732.
- [14] Singh, Abhiranjan, and Seemanti Saha. "Machine/deep learning based estimation and detection in OFDM communication systems with various channel imperfections." *Wireless Networks* 28.6 (2022): 2637-2650.
- [15] Morelli, Michele, and Marco Moretti. "Fine carrier and sampling frequency synchronization in OFDM systems." *IEEE Transactions on Wireless Communications* 9.4 (2010): 1514-1524.
- [16] Shi, Kai, Erchin Serpedin, and Philippe Ciblat. "Decision-directed fine synchronization in OFDM systems." *IEEE Transactions on Communications* 53.3 (2005): 408-412.
- [17] Tao, Yuting, Jian Yang, and Heyou Chang. "Enhanced iterative projection for subclass discriminant analysis under EM-alike framework." *Pattern recognition* 47.3 (2014): 1113-1125.
- [18] Kecman, Vojislav. "Support vector machines—an introduction." *Support vector machines: theory and applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. 1-47.
- [19] Kruse, Rudolf, et al. "Multi-layer perceptrons." *Computational intelligence: a methodological introduction*. Cham: Springer International Publishing, 2022. 53-124.
- [20] Marquardt, Donald W. "An algorithm for least-squares estimation of nonlinear parameters." *Journal of the society for Industrial and Applied Mathematics* 11.2 (1963): 431-441.
- [21] Greenacre, Michael, et al. "Principal component analysis." *Nature Reviews Methods Primers* 2.1 (2022): 100.
- [22] Wang, Yongchang, and Ligu Zhu. "Research and implementation of SVD in machine learning." *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*. IEEE, 2017.
- [23] Kotb, Moatasem Mohammed Elsayed, et al. "Fine carrier frequency offset estimation for OFDM and MIMO-OFDM systems: A comparative study." *Scientific Reports* 15.1 (2025): 15622.
- [24] Elsayed, Moatasem Mohammed, et al. "Sampling Frequency Offset Estimation in OFDM Systems: A comparative study." *Benha Journal of Applied Sciences* 10.4 (2025): 89-100.