Pointing Error And Mitigation Techniques Using Machine Learning Algorithms in Free Space Optical Communication

Souvick Chakraborty
Electrical and Information Engineering
Christian-Albrechts-Universität zu Kiel
Kiel, Germany
souvick.chakraborty@stu.uni-kiel.de

***Abstract*—Free-space optical (FSO) communication offers a promising solution for high-bandwidth satellite-ground and inter-satellite communications. However, maintaining line of sight, beam alignment, and minimal pointing error remains a significant challenge. This study provides a survey of recent advancements in pointing error mitigation techniques for FSO communication. Key approaches include tree-based regressors, the K-Nearest Neighbor (KNN) algorithm, and convolutional neural networks (Conv1D). The survey highlights that ML-based methods significantly enhance pointing accuracy compared to traditional correction techniques, with KNN excelling in nonlinear error correction, tree-based models offering robust predictions, and Conv1D providing precise gain estimation.**

***Index Terms*—Free-space optical communication, pointing error, machine learning, convolutional neural network, K-nearest neighbor, tree-based regression.**

I. Introduction

Over the years, free-space optical (FSO) communication has become significantly important, from Starlink and SpaceX to NASA's deep space exploration. FSO communication offers unique features: high bandwidth, a license-free spectrum, high data rates, easy and quick deployability, low power requirements, and minimal mass constraints. It also has applications in remote sensing, radio astronomy, military operations, disaster recovery, last-mile access, wireless cellular network backhaul, and more [1], [2].

Although FSO communication systems offer many benefits, they face significant challenges, one of the most critical being pointing errors. Pointing errors occur when there is a misalignment between the transmitted communication beam and the receiver due to factors such as mechanical vibrations, atmospheric disturbances, or terminal movement. These errors can result in substantial loss of received signal power and, in severe cases, lead to a complete loss of the communication link. Addressing pointing errors is crucial to ensure reliable and efficient long-distance optical communication [1], [3].

This paper provides a comprehensive survey on pointing errors and their mitigation techniques for inter-satellite links (ISLs) in free-space optical communication (FSO). Additionally, the survey will cover some mitigation techniques utilizing the latest machine learning models and compare them with existing models. Mitigating these errors in satellite FSO systems is crucial to improve acquisition probability, shorten capture time, and maintain the quality of the communication link.

II. Pointing Error and related Challenges

Pointing error refers to the misalignment of the optical beam from its intended target, which in the context of free space optical (FSO) communication, is typically the receiver aperture. It represents a deviation from the ideal line-of-sight (LOS) connection and can significantly degrade the performance of an FSO system [4]. Several challenges contribute to pointing errors in FSO communication:

- *Platform Jitter and Mechanical Vibrations*: Thruster firings, reaction wheels, solar panel movement, or antenna adjustments can induce mechanical disturbances [1], [2]. Vibrations and platform jitter cause deviations in the direction of the transmitted optical beam. These deviations can be in the order of microradians, which is significant over long distances [5].
- *Point-Ahead Angle (PAA) Misalignment*: Due to the relative motion between satellites, the transmitter must point ahead of the receiver's current position to account for the beam's travel time. Errors in estimating the PAA due to orbital uncertainties or incorrect sensor data result in beam misalignment [3], [5], [6].
- *Acquisition, Pointing, and Tracking (APT) Errors*: The APT system locks the transmitted beam onto the receiver. Beam acquisition and tracking become difficult due to inaccuracies in the optical and electronic subsystems, especially during the initial link establishment [2], [6].
- *Beam Divergence and Propagation Loss*: Inter-satellite links use narrow laser beams for high data rates, but narrow beams are more prone to pointing errors. A small angular misalignment can cause the beam to completely miss the receiver at long distances [1], [2].

These challenges lead to reduced received signal power, increased bit error rates, decreased throughput, and, ultimately, link failure.

Over the years, many mitigation techniques have been used to reduce pointing errors. These include mechanical systems such as fast-steering mirrors (FSMs), gimbals, and inertial measurement units (IMUs) for beam steering, as well as adaptive optics, modulation and redundancy, feedback control systems, and wavelength division multiplexing (WDM) with polarization interleaving [1], [2], [6].

Although these traditional techniques have been used for many years, they face several limitations due to the dynamic nature of space and the need for precise alignment. Traditional hardware and control algorithms, such as proportional-integral-derivative (PID) controllers, do not account for all parameters and environmental factors necessary to minimize pointing errors [5].

On the other hand, over the last 10-15 years, advancements in machine learning models and modern algorithms have significantly improved. Simultaneously, the cost of computational resources, such as GPUs and edge processors, has decreased [7]. These scientific breakthroughs have made it possible to implement intelligent systems onboard satellites for real-time pointing error correction.

III. Overview of Machine Learning Mitigation Techniques

This literature survey provides a brief overview of three machine learning models and explains how they achieve better results.

A. *K-Nearest Neighbor (KNN) Model*: The paper [8] proposed a K-Nearest Neighbor based nonparametric approach to correct nonlinear pointing errors after linear errors have been compensated. The pointing error is defined as the difference between the *actual direction* and the *ideal guide direction* of the Line of Sight (LOS):

$$\Delta A = \hat{A} - A, \quad \Delta E = \hat{E} - E$$

where $\hat{A}$ and $\hat{E}$ are the azimuth and elevation guide values. A and E are the actual measured values.

Linear pointing errors are first corrected using a *parameter model*. The residual errors after this correction are considered nonlinear pointing errors. Parameter models can correct linear pointing errors, they cannot effectively handle nonlinear errors.

The KNN algorithm is a nonparametric method that estimates the value of a test sample based on its k nearest neighbors in the dataset. The process involves:

- *Distance Measurement*: The Euclidean distance between two points P_1 and P_2 (in azimuth-elevation space) is computed as:

$$d(P_1, P_2) = \sqrt{(dA)^2 + (dE)^2}$$

where:

$$dA = \min(|A_1 - A_2|, 2\pi - |A_1 - A_2|), \quad dE = |E_1 - E_2|$$

- *Selection of Neighbors*: The *k nearest neighbors* to the test sample (target pointing direction) are selected from historical calibration data (i.e., known star positions and pointing errors).
- *Weight Calculation*: A *Gaussian-like function* is used to assign weights to the nearest neighbors based on their distances:

$$WT_i = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{d_i^2}{2d_k^2}\right)$$

where d_k is the distance of the k-th nearest neighbor. The weight is normalized:

$$w_i = \frac{WT_i}{\sum_{j=1}^{k} WT_j}$$

- *Prediction of Nonlinear Pointing Error*: The predicted pointing error is computed as a *weighted average* of the pointing errors of the nearest neighbors:

$$y_{\text{pred}} = \sum_{i=1}^{k} w_i y_i$$

where y_i is the known pointing error of the i-th neighbor.

The optimal value of k (i.e., the number of neighbors) is determined using *Generalized Cross Validation (GCV)*:

$$GCV(k) = \frac{\sum_{i=1}^{n} (y_{t_i} - \hat{y}_{t_i})^2}{\left(1 - \frac{1}{n}\sum_{i=1}^{n} w_{i1}\right)^2}$$

where, y_{t_i} is the true pointing error of the i-th known sample and $\hat{y}_{t_i}$ is the predicted pointing error for the same sample. The value of k is chosen to minimize $GCV(k)$. The system is tested using an *optical communication terminal (OCT)* that tracks and measures star positions in various directions while mounted on a motion platform. The OCT records the azimuth (A) and elevation (E) angles, attitude measurements, and encoder readings. Before correction, the initial pointing error was large (1312.9 µrad). After Parameter Model correction, the pointing error was reduced to 87.3 µrad. Finally, after KNN correction, the pointing error was further reduced to *69.0 µrad* for calibration stars and *70.8 µrad* for target stars. The proposed KNN algorithm effectively corrects nonlinear pointing errors after linear errors have been compensated. Experimental results show significant improvement in pointing accuracy, demonstrating that a nonparametric KNN approach is well-suited for mitigating pointing errors in motion platforms.

B. *Convolutional Neural Network (CNN)-Based Closed-Loop Control System*: In [5], a 1D convolutional neural network (Conv1D) was employed to mitigate beam pointing errors in satellite-to-ground free-space optical (FSO) communication by predicting optimal gain values for a closed-loop feedback control system. The Conv1D model processes sequential input data, such as control and noise parameters, atmospheric turbulence indices, and system vibrations, to extract temporal features and estimate the gain matrix needed to stabilize beam displacement. The

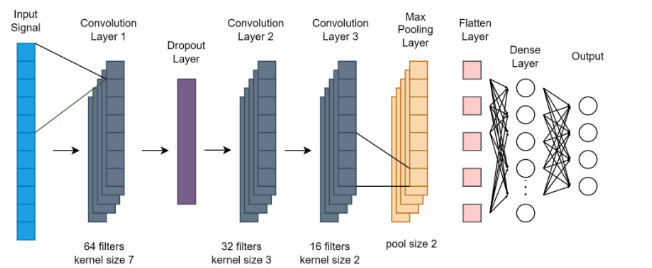

Fig. 1: Network architecture for the 1D convolution multi-output regression model [5].

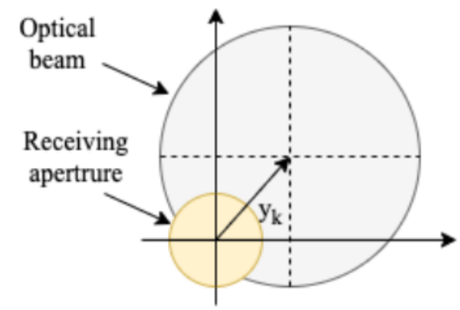

Fig. 2: Optical beam, receiving aperture and the displacement vector y_k [9]

architecture includes convolutional layers to capture patterns in time-series data, activation functions for non-linear mapping, and pooling layers to enhance robustness against noise. By continuously feeding real-time pointing error measurements into the model, it dynamically adjusts the transmitter's position to maintain beam alignment with the receiver.

The Conv1D model architecture used for multi-output regression in this study is designed to predict the gain matrix of the closed-loop system for free space optical (FSO) satellite-to-ground communications. It uses convolutional layers to capture temporal dependencies in the input data.The input features are temporal data that vary over time, such as the scintillation index, Gaussian noise, and the level of attenuation. These input features are fed into the Conv1D model. The initial Conv1D layer consists of 64 filters with a kernel size of 7. This large kernel size helps capture a wide range of local patterns in the input data and gather information from a broader span of neighboring time steps, making it suitable for capturing longer-term dependencies. The activation function used in the convolutional layers is ReLU.After the initial layer, there are additional Conv1D layers with smaller kernel sizes of 3 and 2, respectively. These layers are used to capture more detailed and localized patterns from the input data. While the initial Conv1D layer captures broader features, these subsequent layers extract finer-grained information. The number of filters is also reduced in these layers to balance complexity and prevent overfitting.A dropout layer with a rate of 0.5 is included in the Conv1D model to address overfitting and enhance generalization. This deactivates half of the units during training to achieve a balance between retaining information and effectively regularizing the model.A Max Pooling 1D layer is used to down-sample the feature maps, reducing their spatial dimensions.A flatten layer reshapes the output of the previous layers to create a compressed representation of the extracted features. The first dense layer is responsible for making predictions based on the learned representations from the preceding layers. By applying an activation function, this layer introduces non-linearity to the model, enabling it to learn complex and non-linear relationships between the flattened features and the target variables. The final dense layer outputs the predicted gain matrix, which consists of 4 values. The model produces multiple output variables that represent the elements of the 2 × 2 gain matrix K [5].

C. *Tree-based regression models*: In [9], authors proposed a multi-output regression model based on tree-based algorithms to predict and optimize the gain matrix K in a closed-loop feedback system, ensuring that the beam remains centered on the receiver aperture. The pointing error y_k is that the system is modeled as a displacement vector between position of the transmitted beam (θ_k) and the receiver aperture (α_k):

$$y_k = d(\theta_k - \alpha_k)$$

where d is the distance between the transmitter and the receiver. The closed-loop system aims to minimize the pointing error by adjusting the transmitted beam's position based on the observed displacement. The gain matrix K determines the corrective feedback applied to the system. The goal is to find an optimal K that stabilizes the closed-loop system and reduces the pointing error by ensuring:

$$\|y_k\| \leq \epsilon \|w_k\|$$

where w_k represents disturbances (e.g., noise, turbulence), and ϵ is the disturbance attenuation level.

To predict the optimal gain matrix K, three tree-based regression models are used:

- *Decision Tree Regressor*: A simple model that splits the dataset based on feature values to predict K. It works by recursively dividing the data space and assigning predictions based on averages of the data in each split.
- *Random Forest Regressor*: An ensemble of decision trees. Each tree is trained on a random subset of the data, and the final prediction is obtained by averaging the predictions of all trees. This helps reduce overfitting and improves robustness.
- *Gradient Boosting Regressor*: An ensemble learning method where trees are built sequentially. Each tree

is trained to correct the errors made by the previous trees. This process focuses on minimizing the overall prediction error.

The system uses multiple input parameters (features) to predict K, including the system matrix (a_p, a_l) of the transmitter and receiver, the control matrix (b_p) for adjusting the beam, noise parameters (r_p, r_l) related to disturbances, the scintillation index (σ^2) representing atmospheric turbulence, the iiradiance (I) of the signal, and white Gaussian noise components at both the transmitter and receiver. Feature importance analysis showed that the noise matrices and the scintillation index had a significant influence on predicting the gain values.

A synthetic dataset was generated for training the machine learning models. The dataset was created based on a stochastic state-space model using known system parameters and turbulence levels. Multiple input-output pairs were generated, where the inputs were the system features and the outputs were the gain matrix K.

The tree-based regression models were trained on the dataset to learn the mapping from the input features to the optimal gain values K. The performance of the models was evaluated using Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE).

The model was tested on open-loop and closed-loop systems. In the *open-loop system*, the pointing error was significantly higher due to the absence of feedback correction. In the *closed-loop system*, the predicted gain matrix K was used to dynamically adjust the beam position, resulting in a stable pointing error within allowable limits. The pointing error was reduced from significant values in the open-loop system to minimal alignment errors in the closed-loop system.

The tree-based regression model effectively mitigates pointing errors in FSO systems by predicting the optimal gain matrix for a closed-loop feedback system. The *decision tree regressor* provided a good balance of performance and simplicity, while *random forest* and *gradient boosting* improved robustness and accuracy.

IV. Conclusion

An overview of pointing errors in FSO communication and three machine learning-related mitigation methods is discussed in this article. The growing trend of satellite communication, along with the commercialization of satellites, is expected to be a significant boost for FSO communication. With the rise of edge computing and TinyML, the cost of implementing these machine learning methods is anticipated to become more affordable in the near future. These machine learning approaches demonstrate that the performance and cost efficiency of communication using FSO can improve significantly.

References

[1] H. Kaushal and G. Kaddoum, "Optical communication in space: Challenges and mitigation techniques," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 57–96, 2017, Conference Name: IEEE Communications Surveys & Tutorials, ISSN: 1553-877X. DOI: 10.1109/COMST.2016.2603518.

[2] G. Wang, F. Yang, J. Song, and Z. Han, "Free space optical communication for inter-satellite link: Architecture, potentials and trends," *IEEE Communications Magazine*, vol. 62, no. 3, pp. 110–116, Mar. 2024, Conference Name: IEEE Communications Magazine, ISSN: 1558-1896. DOI: 10.1109/MCOM.002.2300024.

[3] J. Zhang, Y. Cao, P. Guo, *et al.*, "Estimation of direction and zero errors of satellite laser terminals in low-light conditions based on machine learning," *JOSA A*, vol. 41, no. 12, pp. 2316–2326, Dec. 1, 2024, Publisher: Optica Publishing Group, ISSN: 1520-8532. DOI: 10.1364/JOSAA.533672.

[4] H. Kaushal, V. K. Jain, and S. Kar, "Acquisition, tracking, and pointing," in *Free Space Optical Communication*, H. Kaushal, V. Jain, and S. Kar, Eds., New Delhi: Springer India, 2017, pp. 119–137, ISBN: 978-81-322-3691-7. DOI: 10.1007/978-81-322-3691-7_4.

[5] N. Maharjan and B. W. Kim, "Machine learning-based beam pointing error reduction for satellite–ground FSO links," *Electronics*, vol. 13, no. 17, p. 3466, 2024, Num Pages: 3466 Place: Basel, Switzerland Publisher: MDPI AG. DOI: 10.3390/electronics13173466.

[6] A. Sharma, "Analysis and mitigation of receiver pointing error angle on inter-satellite communication," 2015.

[7] A. Ananthaswamy, *Why Machines Learn*. Jul. 16, 2024.

[8] C. Peng, D. He, Y. Huang, *et al.*, "Pointing-error correction of optical communication terminals on motion platforms based on a k-nearest neighbor algorithm," *IEEE Photonics Journal*, vol. 14, no. 6, pp. 1–7, Dec. 2022, Conference Name: IEEE Photonics Journal, ISSN: 1943-0655. DOI: 10.1109/JPHOT.2022.3220400.

[9] N. Maharjan and B. W. Kim, "Reduction of beam pointing error on free space optical communication links using tree based machine learning multi-output regression model," in *2023 Fourteenth International Conference on Ubiquitous and Future Networks (ICUFN)*, ISSN: 2165-8536, Jul. 2023, pp. 664–668. DOI: 10.1109/ICUFN57995.2023.10200700.