Deep Learning-Driven Beam and Blockage Forecasting for Reliable mmWave Vehicular Networks

Muhammad Enayetur Rahman[1], Md. Shirajum Munir[2], and Sachin Shetty[3]

[1, 3]Department of Electrical and Computer Engineering, Old Dominion University, Norfolk, VA, 23529, USA
[2]School of Computing, Analytics, and Modeling, 1601 Maple St, Carrollton, GA 30118, USA.

***Abstract*—In the rapidly advancing field of millimeter-wave (mmWave) communications, ensuring a stable and efficient connection poses a substantial challenge due to the dynamic environmental conditions. Predicting beam direction and potential blockage is crucial for maintaining uninterrupted service, particularly in mobile scenarios. This study introduces an innovative deep-learning approach for predicting beam direction and blockage events in vehicular networks. We propose a unique technique centered on the Gated Recurrent Unit-based Beam and Blockage Prediction (GBBP) model to improve prediction accuracy. Our multivariate GBBP model anticipates the optimal serving beam and forecasts future blockage events, outperforming baseline models such as Long Short-Term Memory (LSTM), Recurrent Neural Networks (RNNs), and Attention-BiLSTM. The proposed GBBP model demonstrates superior performance in modeling the complex dynamics of mmWave communication channels, achieving high accuracy across various evaluation metrics, including Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and blockage accuracy** $(\sim 99\%)$**.**

***Index Terms*—Vehicular Communication, Millimeter Wave, Blockage Prediction, Beam Prediction, Deep Learning, Machine Learning.**

I. Introduction

Millimeter-wave (mmWave) technology, operating in the 30-300 GHz range (including 28 GHz), is key to advancing 5G and future wireless networks. Its shorter wavelengths enable higher data speeds, capacity, and support for emerging applications like virtual reality, high-resolution radar, and low-latency communications, addressing the growing demand for mobile broadband services [1], [2].

Yet the propagation properties and characteristics of mmWave frequencies provide additional difficulties. Greater attenuation of the higher frequencies results in a smaller coverage area and increases the likelihood of signal blockages by structures and other obstructions. The challenges associated with mmWave communication include significant attenuation against physical objects and interference issues caused by user movement [3], high power directional antennas being required to counteract attenuation, and vulnerability to blocking [4].

Email: mrahm011@odu.edu[1], mmunir@westga.edu[2], sshetty@odu.edu[3].
This work is supported in part by DoD Center of Excellence in AI and Machine Learning (CoE-AIML) under Contract Number W911NF-20-2-0277 with the U.S. Army Research Laboratory, National Science Foundation under Grant No. 2219742 and Grant No. 2131001

To ensure stable communication in vehicles using mmWave technology, it's essential to align the beams from both the base station and vehicle using highly directional antennas. Nevertheless, mmWave signals are quickly attenuated and have a short range, making beamforming in dynamic vehicle environments challenging. The ideal mmWave transmission distance is greatly reduced by air and molecular absorption, particularly in the 60 GHz range. As a result, Line of Sight (LOS) paths are crucial to mmWave communication [5]. The beam alignment process is influenced by factors like vehicle location, signal delay, power, arrival and departure angles, and whether the transmission is LOS or not [6].

Conventional methods of beam alignment, including exhaustive beam sweeping, search in every direction for the optimal beam pairing. This approach has a large overhead because of the intricacy of the search procedure. In recent studies, many learning methods have been proposed to resolve the beam/blockage problem in Beyond 5G vehicle networks. These models show significant promise in reducing the overhead and search complexity linked to conventional methods. Recently, various research have been conducted using machine learning (ML) approaches to assure the beam direction process from the base station to the mobile user, solving the limits given by high mobility in V2X situations. For instance, the authors [7] have created an attention-based AI model for predicting the Reconfigurable Intelligent Surface (RIS) beam for the Unmanned Aerial Vehicle (UAV) users located in 3D space for terahertz (THz) communication.

In this study, we propose a novel deep learning (DL) method to predict both beam direction and potential blockages for a moving vehicle. We specifically approach the beam direction task as a regression problem, utilizing inputs such as the distance between the vehicle and the millimeter-wave base station (mmBS), the departure and arrival angles of azimuth and elevation, vehicle location data, path loss, time delay, and power in our proposed Gated Recurrent Unit (GRU) based Beam and Blockage Prediction (GBBP) model.

Contributions of the paper:

- We formulate the beam direction and blockage problem by framing it regression and binary problem respectively, aiming to maximize the received signal power at the mobile user.

- We introduced and developed a multivariate GBBP deep learning (DL) model, that can predict the forthcoming angles of mmWave beams and blockages for a vehicle user.
- Our innovative deep learning model was thoroughly compared with other models: Long Short-Term Memory (LSTM), Attention-BiLSTM, and Recurrent Neural Network (RNN) models. This evaluation involved a detailed assessment of the models' Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), and Root-Mean Squared Error (RMSE) values. Our GBBP DL model performs well than all other models.

Paper outline: The relevant work is presented in the Introduction section. Section II discusses the system model, channel model, and problem statement. In Section III, we describe our proposed GBBP model for vehicular mobile networks. In Section IV, the simulation and its outcomes are examined and evaluated. Finally, we conclude our investigation with closing thoughts and recommendations for further research.

Notations: The set of natural numbers represented as $\mathbb{N}$, the $x \times y$ dimensional real and complex space numbers are represented as $\mathbb{R}^{x\times y}$ and $\mathbb{C}^{x\times y}$. With mean μ and variance σ^2 the normal distribution of the complex number is $\mathcal{CN}\left(\mu, \sigma^2\right)$. Absolute value or cardinality of a set represented as $|\cdot|$. Boldface symbols are denoted for matrices and vectors, e.g., $\mathbf{A}$, $\mathbf{a}$, and a represented as matrix, vector, and scalar values. $\mathbf{A}^T, \mathbf{A}^*$ are mentioned as transpose and conjugate transpose (Hermitian) of $\mathbf{A}$.

II. System Model, and Problem Statement

We consider an outdoor mmWave cellular network with multiple Base Stations (BSs) and moving vehicular users (VUs). The BSs act as transmitters, handling signaling and control processes using AI-assisted algorithms. Downlink data transmission operates via mmWave Multiple Input Multiple Output (MIMO) systems. VUs move in straight-line trajectories, and VUs are positioned at the same height. Uniform Planar Array (UPA) antennas are installed on every BS, whereas a single omnidirectional antenna is attached to each VU. A key challenge in this scenario is the occurrence of blockage events, where obstacles obstruct the LOS path. During such events, NLOS signals may still be present. We aim to predict these blockages using a Deep Learning (DL) algorithm to reduce the computational complexity of beam recalculations.

A. System Model

As illustrated in Fig. 1, the network consists of multiple BSs, each equipped with M_t UPA antennas, and V moving vehicular users (VUs), each with a single omnidirectional antenna. The BSs are statically positioned along both sides of the main street, while VUs move at a constant speed v_m, with their positions updated dynamically using cartesian coordinates. At each time step, every VU is connected to a BS for mmWave communication. The BSs ensure reliable communication by mitigating data loss through their multi-antenna configurations. Although each VU is modeled with a single antenna for simplicity, the system can be extended to support multi-antenna VUs, where signals from each antenna are aggregated to form the overall received signal. The VUs are assumed to recognize the locations of the BSs for efficient handover and communication. To address the challenge of blockage events, where obstacles obstruct the direct LoS path, we employ a simple but powerful DL algorithm to predict and mitigate these events. This approach reduces the computational burden of continuously recalculating beamforming strategies during NLOS conditions, thus improving overall system performance.

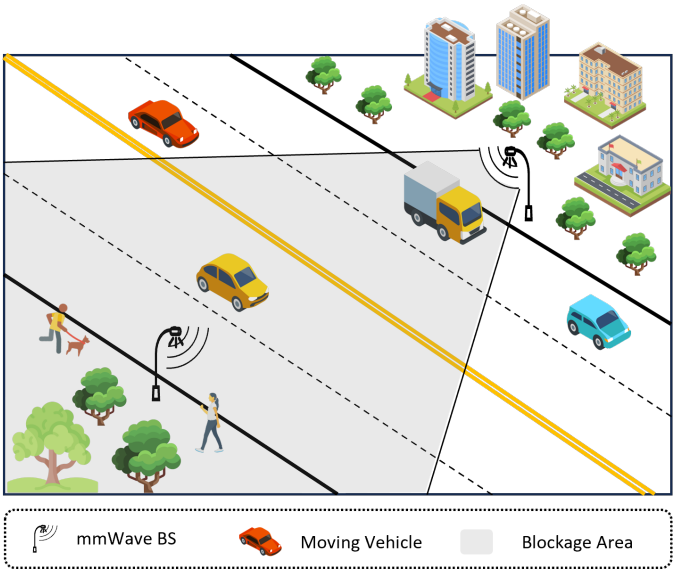

Fig. 1. System model for a mmWave vehicular scenario.

B. Channel Model

We are considering a downlink mmwave-BS (mmBS) automobile communication system where mmBS is equipped with $M_{bs} := M_{bs}^x \times M_{bs}^y \times M_{bs}^z$ antennas. Here, M_{bs}^x, M_{bs}^y, and M_{bs}^z are the x, y, and z direction number of antennas of BS-mm. Similarly, for moving vehicle users (VU), we apply the same assumption, i.e., $N_{vu} := N_{vu}^x \times N_{vu}^y \times N_{vu}^z$ antennas.

We create the mmWave channel between the mmBS and the VU, which are located at various locations inside the vehicle environment, thru ray-tracing. We can obtain essential channel data using this approach, including the azimuth and elevation angles at the departure and arrival sites, the arrival time, the received signal strength, and the LOS and NLOS paths that link a VU to the mmBS.

The channel matrix $\mathbf{H} \in C^{M_{bs}\times N_{vu}}$, consisting of L paths can be described as signal transmission from a mmBS to a VU in a vehicular environment. To include the Doppler effect in our model, we adjust the channel matrix $\mathbf{H}$ by introducing a time-varying complex gain factor for each path, accounting for frequency shifts due to the relative motion between the Vehicle Unit (VU) and the mmWave Base Station (mmBS). This adjustment introduces a timing disparity influenced by the frequency, the velocity, and the signal arrival direction. By

considering the Doppler effect, the adjusted channel matrix $\mathbf{H}$ [8] provides a more precise depiction of the channel in vehicular communication contexts, represented as follows:

$$\mathbf{H} = \sum_{\ell=1}^{L} \alpha_\ell \mathbf{a}_{\mathrm{vu}}(\theta_{\mathrm{vu},\ell}, \phi_{\mathrm{vu},\ell}) \mathbf{a}_{\mathrm{bs}}{}^*(\theta_{\mathrm{bs},\ell}, \phi_{\mathrm{bs},\ell}). \quad (1)$$

Here, for each path ℓ, α_{vu} is the complex gain factor, $\mathbf{H}$ is the channel matrix at time t and frequency f. $\mathbf{a}_{\mathrm{vu}}(\theta, \phi)$, $\mathbf{a}_{\mathrm{bs}}(\theta, \phi)$ are the steering vectors at the VU and BS respectively, where $(\theta, \phi) \in [-\pi/2, \pi/2]$ are the azimuth and zenith (elevation) angle pair.

For the UPA antenna with M_t antenna, it's beam steering vector is limited to the azimuth angle. So, it can be expressed as [9]:

$$\mathbf{a}_{\mathrm{bs}}(\theta_\ell) = \frac{1}{\sqrt{M_t}} [1, e^{j\frac{2\pi}{\lambda} d \sin\theta_\ell}, ..., e^{j(M_{bs}-1)\frac{2\pi}{\lambda} d \sin\theta_\ell}]^T, \quad (2)$$

where λ is the wavelength, and d is the distance between antenna elements. d is calculated as $d = \lambda/2$.

Similarly, the array response vector at the VU, $\mathbf{a}_{\mathrm{vu}}(\phi_\ell)$ can be found as follows:

$$\mathbf{a}_{\mathrm{vu}}(\phi_\ell) = \frac{1}{\sqrt{N_{vu}}} [1, e^{j\frac{2\pi}{\lambda} d \sin\phi_\ell}, ..., e^{j(N_{vu}-1)\frac{2\pi}{\lambda} d \sin\phi_\ell}]^T. \quad (3)$$

The complex gain factor can be written as:

$$\alpha_{\ell,k}(t) = \alpha_{\ell,k} \cdot e^{j2\pi f_d k T_s t}, \quad (4)$$

where k is the sample index at time t, f_d is the doppler-shift frequency and T_s is the sampling interval. The dopper-shift frequency f_d can be modified by

$$f_d = \frac{2 v_{vu}}{\lambda}, \quad (5)$$

where v_{vu} is the VU's speed or velocity. The resulting downlink mmwave signnal received at the VU k can be found as:

$$y_k = f_k^H \mathbf{H} \omega_k s_k + z_k, \quad (6)$$

where $f_k \in \mathbb{C}^{N_{vu} \times 1}$ is the beamforming vector at vehicle k, $w_k \in \mathbb{C}^{M_{bs} \times 1}$ is the transmit beamforming vector, $\mathbf{H} \in \mathbb{C}^{M_{bs} \times N_{vu}}$ represents channel matrix, and z_k is the additive white Gaussian noise (AWGN) with zero mean and variance of σ^2.

C. Problem Statement

Our study primarily aims to forecast future beamforming trajectories and blockage status. To accomplish this objective, we develop a Deep Learning (DL) model designed to efficiently guide the future beam and blockage for vehicle users. The base station adapts its beam, denoted as $\mathbf{b}$, and blockage denoted as $\mathbf{L}$ to align with the user's movements and maintain a consistent connection. Beam width, user velocity, and the quantity of antennas at the base station are a few examples of the variables that might affect beam coherence time. The beam coherence time approximation can be written as [10]:

$$T_{\mathrm{Bn}} = \frac{D}{v_{\mathrm{vu}} \sin(\alpha)} \frac{\Theta_n}{2}, \quad (7)$$

where v_{vu} is the vehicle velocity, D is the distance between BS and VU, angle α represents the relationship between the primary reflector's direction and the travel direction. Θ_n is the beam-width from the nth-BS. When t is the beam coherence time, the beam that is being utilized to communicate with the VU at a given time is represented as $\mathbf{b_n}^{(t)}$. This t-step is used to build a series of beams.

$$\mathcal{B}_t = \left\{ \mathbf{b_n}^{(1)}, \mathbf{b_n}^{(2)}, \ldots, \mathbf{b_n}^{(t)} \right\}. \quad (8)$$

Let, $l^{(t)}$ represent the presence of a LOS connection at time step t between the mmBS and VU:

$$\mathcal{L}_t = \left\{ \mathbf{l}^{(1)}, \mathbf{l}^{(2)}, \ldots, \mathbf{l}^{(t)} \right\}. \quad (9)$$

Let, $\mathbf{d}^{(t)}$ denote the VU location at time t when the beam $\mathbf{b_n}^{(t)}$ was selected. Then, we represent the location sequence as:

$$\mathcal{D}_t = \left\{ \mathbf{d}^{(1)}, \mathbf{d}^{(2)}, \ldots, \mathbf{d}^{(t)} \right\}. \quad (10)$$

For each time step, we also can use the pathloss information p_t.

$$\mathcal{P}_t = \left\{ \mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \ldots, \mathbf{p}^{(t)} \right\}. \quad (11)$$

We demonstrate a solution to the task of predicting the beam and blockage at time $t+1$ using above sequences: $\mathcal{B}_t$, $\mathcal{L}_t$, $\mathcal{D}_t$, $\mathcal{P}_t$. This is accomplished through the application of a multivariate GBBP deep learning framework, which effectively maps the beam and blockage sequences from $\mathcal{B}_t$ and $\mathcal{L}_t$ to $\mathcal{B}_{t+1}$ and $\mathcal{L}_{t+1}$. Our system excels at accurately predicting the optimal beam and blockage.

III. Solution Approach

In our proposed GBBP model, we implemented a multivariate sequence technique to predict the beam angles and blockage status. Our GBBP model gives us the capabilities to more accurately predict beam and blockage. The overall GBBP architecture is shown in Fig. 2 which provides the architecture for both beam angles and blockage prediction.

A. Foudational building blocks of GBBP

The baseline of our proposed GBBP model's architecture lies using the Gated Recurrent Unit (GRU) based model, which was designed to improve the learning of long-range dependencies in sequential data and it can solve vanishing gradient problem [11]. Below is a detailed explanation of the GRU structure.

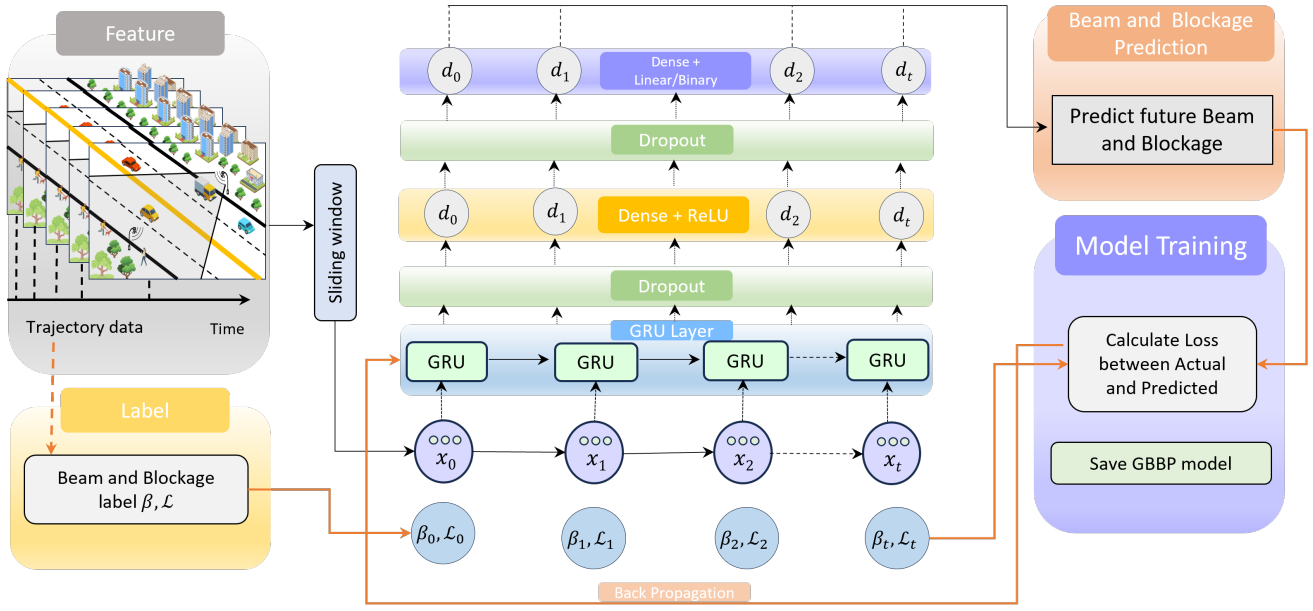

Fig. 2. The Proposed GBBP model architecture for multivariate Beam and Blockage Prediction.

1) Reset gate (r_t): The reset gate defines how much of the previous hidden state needs to be forgotten. When the reset gate is close to zero, it allows the network to learn short-term dependencies by erasing most of the prior state.

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r). \tag{12}$$

Where, r_t is the reset gate, σ is the sigmoid function, W_r is the weight for the reset gate, x_t is the input at time t, U_r is the self-weight, h_{t-1} is the hidden state at previous time step, and b_r is the bias term.

2) Update gate (z_t): The update gate controls how much of the previous hidden state is carried forward and how much of the new information is used to update the hidden state. This gate allows the GRU to retain long-term dependencies by determining how much of the past information is retained.

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z). \tag{13}$$

Where, z_t is the update gate, W_z is the weight matrix, U_z is the self-weight, and b_z is the bias term for the update gate at time t.

3) Candidate hidden state ($\tilde{h}_t$): This state contains the candidate information that can be used to update the hidden state. The reset gate controls how much of the previous hidden state contributes to this new candidate hidden state.

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h). \tag{14}$$

Where, $\tilde{h}_t$ is the candidate hidden state, W_h is the weight matrix, $\odot$ is the element wise multiplication, that allows selective forgetting of the previous state. $\tanh$ is the hyperbolic tangent activation function, b_h is the bias term.

4) Hidden state (h_t): The candidate hidden state and the prior hidden state are combined to create the final hidden state, which is managed by the update gate. The update gate decides whether to retain the old hidden state or update it with the new candidate state.

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t. \tag{15}$$

Where, h_t is the hidden state, and $(1 - z_t)$ controls the retention of the old hidden state, and z_t controls the influence of the new candidate state.

B. GBBP architecture description

In our mmWave vehicular scenario, at each position of the vehicle, we can extract the environment data that can be input data to our proposed model. The architecture of the GBBP model is designed to effectively capture temporal dependencies and non-linear patterns in multivariate time series data. It begins with a single GRU layer consisting of 32 units. This layer processes sequential input data and preserves information across time steps. To prevent overfitting, dropout layers with a rate of 20% are applied after both the GRU and fully connected (Dense) layers. The dense layer is defined with 20 units and ReLU activation that can introduce non-linearity, followed by an output layer with 4 units (or 1 unit for blockage) and linear (sigmoid for blockage prediction) activation to match the dimensionality of the target variables. This architecture is suitable for handling multivariate time series data, balancing model complexity and performance through the use of dropout and appropriate activation functions. A summarized algorithm is described in Algorithm 1.

Algorithm 1 GBBP Algorithm for Beam and Blockage Prediction.

Input: Data collection, $I(n)$ for each moving vehicle using equation $8-11$
Output: Beam and Blockage prediction model $\mathcal{B}_{t+1}$, and $\mathcal{L}_{t+1}$
Generate feature set $M(k)$ and beam and blockage label of VU from $I(k)$
Generate the training, validation and test data set X_{train}, X_{val} and X_{test} from $M(k)$
for mini batch $X \in X_{train}$ **do**
  for time window, T data X_T in X **do**
    for time $t = 1$ to T **do**
      Encode h_t using GRU layer
    end for
    Apply a Dropout layer with rate 20%
    Pass the output through a Dense layer, 20 units and ReLU activation
    Apply a Dropout layer, a rate 20%
    Compute Beam/Blockage prediction from a Dense layer with Linear/Sigmoid activation
  end for
  Train the model H with back propagation
end for
Store the best prediction model H

IV. Experimental Analysis and Discussion

This section provided a thorough evaluation and comparison of our suggested GBBP DL model for mmWave mobile users against existing RNN, Attention-BiLSTM, and traditional LSTM models.

A. Experimental Settings

Scenario Description: We experimented our DL model along with other models, by utilizing a publicly available framework [12] that can produce a NextG mmWave dataset for a set of given input parameters. Within this context, we used an outdoor mmWave scenario consisting of two streets and one crossroads in a downtown location, specifically designed to simulate a blockage situation. In this outdoor scenario, one surface acts as a blockage-surface that prevents signals being propagated through the environment. Two more surfaces act as a reflector surfaces to provide additional NLOS routes inside the system. Fig. 3 illustrates the system. For further information can be found in [12].

Scenario Input Parameters and Dataset Generation: In our experimental setup, we utilized the Python based script combined with the framework [12] to generate channel parameters of every transmitter-receiver pair. Next, we provide the system parameters to the framework and the framework constructs the dataset.

We are considering one active base stations (BS-3) for simplicity with $1 \times 8 \times 8$ antenna elements with isotropic antenna radiation pattern. For simplicity, a single antenna user is considered with isotropic radiation pattern. Starting from the 700 row to the 1300 rows user are equally spaced with 20 cm apart with all users heights are 2 m. Each row contains 181 users resulting a total of 108,781 users in the system. The full parameter values are mentioned in Table I.

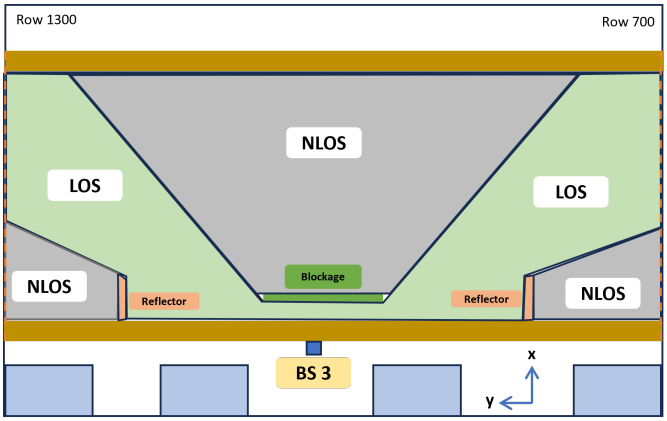

Fig. 3. Experimental scenario top view.

TABLE I
Parameters for Dataset Generation.

Data Generation Parameters	Value [13]
Active Base Station	3
User first and last row	700, 1300
Base Station Antenna (x, y, z)	1, 8, 8
User Antenna (x, y, z)	1, 1, 1
Center Frequency (GHz)	28
Space between Antennas	0.5
OFDM Subcarriers	512
Sampling Factor (OFDM)	1
Bandwidth (GHz)	0.5
OFDM Range	32
Number of Paths	1

Data Pre-processing Steps:

The dataset contains beam vector data such as power, time delay, and direction angle of arrival and departure for both azimuth and zenith. We deleted that beam information in the event if there was no communication channel. Ultimately, beam information was integrated with the VU position, BS, and LOS. We generated a dataset for VU trajectories that has 25-step sequences. It contains the LOS from the BS, VU locations, and beamforming vectors from the BS. There are 106,783 rows in the dataset. For every 25-step sequence, we predicted one row of vectors using 11 feature vectors as input. The experimental dataset was split randomly with a split ratio of 10% for testing and 10% for validation into train, validation, and testing subsets since we have sufficient data. Thus, the number of rows in the training, testing, and validation datasets is 85,427, 10,678, and 10,678 respectively.

Model Description and Training:

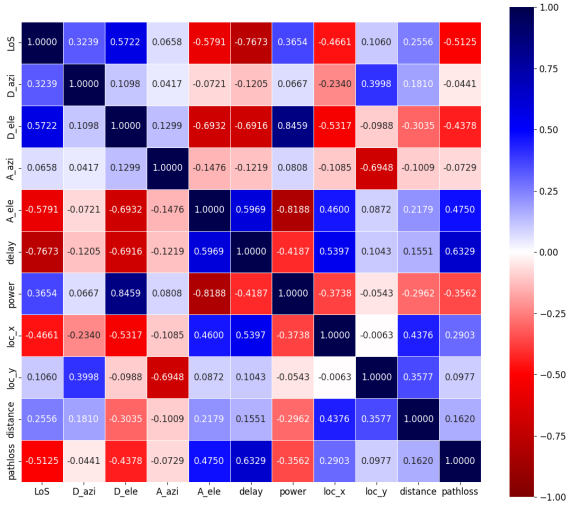

Fig. 4. The heatmap displays the correlation matrix, showcasing values ranging from -1 to 1.

There are no significant outliers in our dataset, and the characteristics have various scales. As depicted in Fig. 4, there is no extreme high correlation feature pair values which is greater than 0.85 or lower than -0.85. After that, we convert the features into a normalized range from 0 to 1. The dataset has many scales for distinct features, and our objective is to convert them while preserving their relation. The following equation for min-max scaling is used to convert the characteristics:

$$X_i' = \frac{X_i - X_{\min}}{X_{max} - X_{min}} \tag{16}$$

To improve our machine-learning model's performance, we apply feature scaling to the input dataset. The scaling parameters are obtained by fitting the scaler to the training set. To guarantee uniform uniformity throughout all data, these learnt parameters are then uniformly applied to the datasets used for training, validation, and testing. We prevent data leaking by limiting the scaling fit to the training set, guaranteeing a more precise assessment of our model. Table II provides a full breakdown of our suggested model's parameters.

TABLE II
PROPOSED GBBP MODEL'S HYPERPARAMETERS.

Parameters	Value
Optimizer	Adam
Dropout	0.2
Learning Rate	0.001
Batch size	256
Epochs	100
Data size	106,783
Data Split (Train:Val:Test)	80:10:10

We conducted a comparative analysis between our novel model and established LSTM, RNN, and one hybrid Attention-BiLSTM, architectures. These models are specifically designed for sequential neural network frameworks, where they process input sequences comprising 11 features per time step. To effectively capture sequential dependencies within the data, GBBP, LSTM, RNN models incorporate 32 units. Additionally, to address potential overfitting issues, a dropout layer with a 20% dropout rate is incorporated. Subsequently, a fully connected layer consisting of 20 neurons and employing a rectified linear unit (ReLU) activation function is introduced to introduce non-linear transformations. We try to maintain almost similar architecture for all of the models so that, a fair comparison can be concluded.

Lastly, to facilitate regression tasks, a fully connected layer comprising 4 neurons and employing a linear activation function is included. To ensure a fair comparison of our proposed model against LSTM, Attention-BiLSTM, RNN counterparts, we carefully constructed those architectures with identical total parameter counts, utilizing the same Adam optimizer with a consistent learning rate of 0.001, and conducting training over an equal number of epochs (100).

B. Result Analysis and Discussion

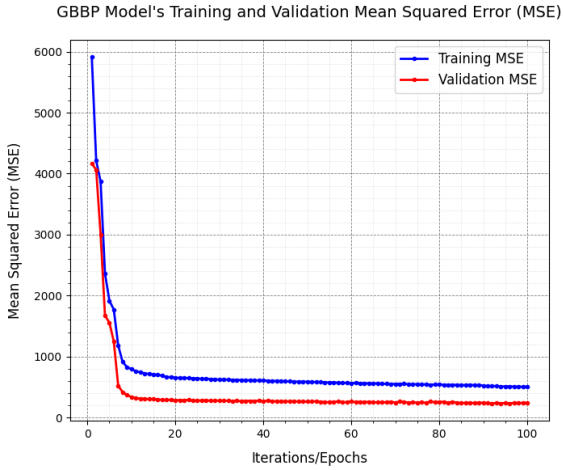

Fig. 5. Iteration vs Mean Squared Error for Beam Prediction

Fig. 5 describes, the learning curve plot of for our proposed GBBP Beam prediction model. We can see that, the validation curve converges with the training curve and there is no overfitting and it indicates that the model is well trained.

Our proposed GBBP model is simple yet powerful to predict simultaneously all departure and arrival angles of the Beam. It can also accurately predict blockage status with almost 99% accuracy. The actual and the predicted values are plotted for the departure (in Fig. 6) and the arrival angles (in Fig. 7). We can clearly see our model almost accurately predicts all the angles.

We compute four metrics, as listed in Table III, to evaluate the model's performance in Beam direction prediction:

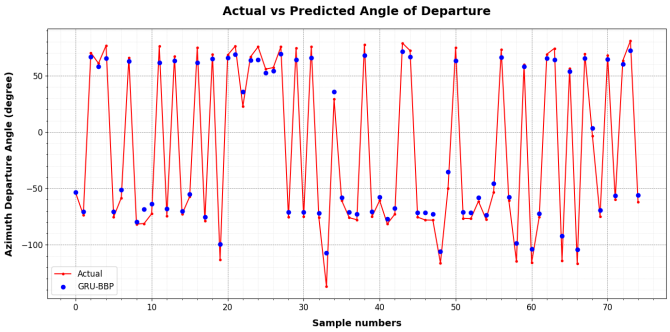

Fig. 6. Actual vs predicted value for Departure Angle

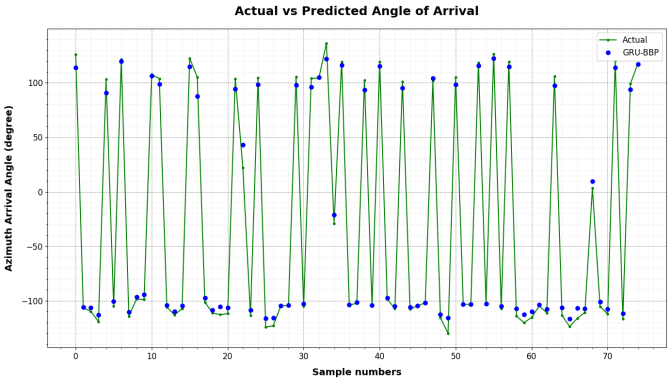

Fig. 7. Actual vs predicted value for Arrival Angle

Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). Our GBBP model demonstrates strong performance in comparison to other models. We also obtained the blockage prediction accuracy is almost 99% with our model. Moreover, our GBBP model is comparatively low complex and total parameters are less than the LSTM and Attention-BiLSTM.

V. Conclusion

In this work, we introduced the Gated Recurrent Unit-based Beam and Blockage Prediction (GBBP) model to address the challenges of predicting beam direction and potential blockage events in mmWave vehicular networks. The proposed multivariate model demonstrated superior performance across multiple evaluation metrics compared to baseline models, including LSTM, Attention-BiLSTM, and RNNs. As shown in the results, our GBBP model achieved the lowest Mean Absolute Error (MAE) of 5.9249, Mean Squared Error (MSE) of 248.5155, Root Mean Squared Error (RMSE) of 15.7644, and Mean Absolute Percentage Error (MAPE) of 0.4054. These outcomes reflect the model's ability to capture and predict the dynamic variations in mmWave communication channels more effectively than traditional deep learning approaches. Additionally, the proposed model's high blockage accuracy of 99% further emphasizes its potential for enhancing mmWave communication reliability in vehicular networks. These findings suggest that the GBBP model is a promising tool for improving beam management and maintaining stable connections in future vehicular communication systems.

TABLE III
Different Models Performance Comparison.

Model	MAE	MSE	RMSE	MAPE	%Acc.
LSTM	6.1713	220.1950	14.8390	0.4461	99.03
RNN	6.2561	251.8392	15.8694	0.4614	98.97
Att-BiLSTM	18.1306	509.6526	22.5755	0.4894	98.82
Our Model	**5.9249**	248.5155	15.7644	**0.4054**	98.80

References

[1] Y. Fujii, T. Iye, K. Tsuda, and A. Tanibayashi, "A 28 ghz beamforming technique for 5g advanced communication systems," in *2022 24th International Conference on Advanced Communication Technology (ICACT)*, 2022, pp. 96–99.

[2] W. Y. Yong, A. Vosoogh, A. Bagheri, C. V. D. Ven, A. Hadaddi, and A. A. Glazunov, "An overview of recent development of the gap-waveguide technology for mmwave and sub-thz applications," *IEEE Access*, vol. 11, pp. 69 378–69 400, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:259721556

[3] J. Li, Y. Niu, H. Wu, B. Ai, S. Chen, Z. Feng, Z. Zhong, and N. Wang, "Mobility support for millimeter wave communications: Opportunities and challenges," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1816–1842, 2022.

[4] F. Yang, J.-B. Wang, H. Zhang, M. Lin, and J. Cheng, "Intelligent reflecting surface assisted mmwave communication using mixed timescale channel state information," *IEEE Transactions on Wireless Communications*, vol. 21, no. 7, pp. 5673–5687, 2022.

[5] M. N. A. Siddiky, M. E. Rahman, and M. S. Uzzal, "Beyond 5g: A comprehensive exploration of 6g wireless communication technologies," *Preprints*, May 2024. [Online]. Available: https://doi.org/10.20944/preprints202405.0715.v1

[6] J. Ye and X. Ge, "Beam management optimization for v2v communications based on deep reinforcement learning," *Scientific Reports*, vol. 13, no. 1, p. 20440, Nov. 2023. [Online]. Available: https://doi.org/10.1038/s41598-023-47769-3

[7] M. E. Rahman, M. S. Munir, and S. Shetty, "An attention-based ai model for 3d beam prediction in thz unmanned aerial vehicle communication," in *2024 International Conference on Computing, Networking and Communications (ICNC)*, 2024, pp. 761–766.

[8] R. W. Heath, N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave mimo systems," *IEEE journal of selected topics in signal processing*, vol. 10, no. 3, pp. 436–453, 2016.

[9] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE journal of selected topics in signal processing*, vol. 8, no. 5, pp. 831–846, 2014.

[10] V. Va, J. Choi, T. Shimizu, G. Bansal, and R. W. Heath, "Inverse multipath fingerprinting for millimeter wave v2i beam alignment," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4042–4058, 2018.

[11] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," 2014. [Online]. Available: https://arxiv.org/abs/1406.1078

[12] A. Alkhateeb, "DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications," in *Proc. of Information Theory and Applications Workshop (ITA)*, San Diego, CA, Feb 2019, pp. 1–8.

[13] N. Abuzainab, M. Alrabeiah, A. Alkhateeb, and Y. E. Sagduyu, "Deep learning for thz drones with flying intelligent surfaces: Beam and handoff prediction," in *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2021, pp. 1–6.