A WiSARD Network Approach for 5G MIMO Beam Selection

Joanna C. Manjarres[1][0000−0003−1124−4601], Douglas O. Cardoso[2,3][0000−0002−1932−334X] Aldebaro Klautau[4][0000−0001−7773−2080], and José Ferreira de Rezende[1][0000−0002−5660−6488]

[1] PESC, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
joanna@land.ufrj.br, rezende@land.ufrj.br
[2] Centre of Linguistics, Faculty of Arts and Humanities, University of Porto, Porto, Portugal
[3] Smart Cities Research Center, Polytechnic Institute of Tomar, Tomar, Portugal
docardoso@letras.up.pt
[4] LASSE, Federal University of Para, Belém, Brazil
aldebaro@ufpa.br

Abstract. The integration of context information and machine learning techniques can enhance the capabilities of 5G/6G networks when dealing with the beam selection problem. This paper proposes the use of a Weightless Neural Network (WiSARD) with multimodal data as input to address this problem. The performance of the WiSARD is compared to classic machine learning algorithms (KNN, Decision Tree, SVC, Random Forest) based on the top-k accuracy in a vehicular network. The simulation results indicate that the WiSARD is a competitive method for this scenario and can be a valuable asset for future cellular networks.

Keywords: 5G · Mm-wave · Beam selection · AI · machine learning · WiSARD network.

1 Introduction

Over the decades, cellular systems continuously evolved. In 1990, data services were available for around 12 million mobile subscriptions. However, this number grew significantly, and in 2021, there were more than 8 billion users [12]. This surge in mobile traffic has highlighted the need to improve the capacity of cellular networks. The use of higher frequencies such as millimeter waves (mmWaves) are expected to fulfill this requirement. However, mmWaves pose several challenges, including high attenuation, loss propagation, path loss, poor foliage penetration, and others [21]. To overcome these challenges, MIMO techniques with a large number of antenna elements have been adopted, for instance to increase the directivity of beams generated by these systems [4].

Among the adopted MIMO algorithms, some rely on accurate channel estimation techniques to establish precoding and/or combining vectors that depend on the user channel and improve the communication between the users and the base station and vice-versa [4]. This process involves the periodic transmission by the user equipment of a pilot sequence that allows the base station to continually estimate the channel. While this method optimizes the achieved SINR, it requires complex processing on both devices and relatively high overhead due to the periodic exchange of significant channel state information (CSI).

A strategy to decrease the required CSI is to previously define a codebook of these beamforming vectors, and embed it in each device. Such codebook can be designed based on the environment in which the system will be used, or using a heuristic as collecting the basis functions from the discrete Fourier transform (DFT). For the sake of simplicity, each vector is said to provide a beam that can be used both in transmission and reception. Thus, in the brute-force approach, each pair

of transmitting and receiving devices needs to scan all the available beams to determine the best pair of beams to be used. This procedure is called beam selection. This strategy of beam selection has the advantage of low channel overhead, but it can be quite expensive depending on the number of beams in the codebook.

There are various proposals in the literature to simplify the process of beam selection and decrease the latency involved. Some research utilizes machine learning techniques that incorporate context information to enhance the characteristics of the environment for beam selection. The authors of [14], [9], [15], and [8] have used machine learning techniques such as Linear SVM, AdaBoost, Decision Tree, Random Forest, Deep Neural Network (DNN), Deep Q Learning (DQN), KNN, Naive Bayes, and Artificial Neural Network (ANN). In [27], a lightweight neural network (NN) and a convolutional neural network (CNN) were used.

This research proposes the utilization of Weightless Neural Networks (WiSARD) for beam selection. This technique offers a lightweight classification model based on random access memory (RAM) [1]. It has been applied in several applications with competitive accuracy scores, low processing time, and easy implementation in embedding hardware [10].

This paper presents a performance comparison of the beam selection accuracy obtained by the WiSARD network and machine learning algorithms such as KNN, Decision Tree, SVC, and Random Forest. The selection process is carried out using multimodal data, including coordinates of the receiver positions and data from the LiDAR sensor.

The following text is organized as follows. Section 2 explains the WiSARD model and how it works. Section 3 describes the dataset used and how the data was prepared to be used by the WiSARD. Section 4 presents the performance of the WiSARD method and compares it to other classical approaches. Finally, the Section 5 draws the conclusions and future works.

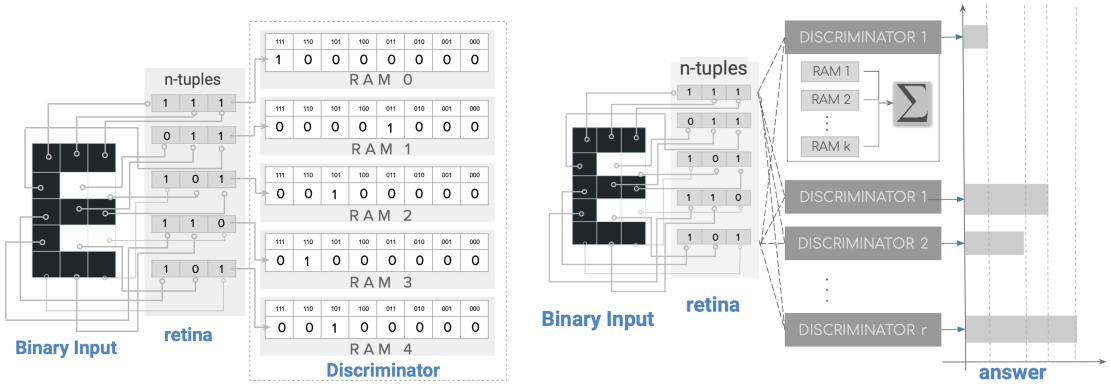
2 Machine Learning Solution

The Weightless Neural Networks (WNNs) represent a class of machine learning models characterized by RAM-based neural networks, where these RAMs serve as artificial neurons. Various WNN models exist in the literature, such as WiSARD (Wilkes, Stonham, and Aleksander Recognition Device), GSN (Goal Seeking Neuron), GRAM (Generalization RAM), and VG-RAM (Virtual Generalization RAM) [18]. For this study, we employed a WiSARD network.

The structure of such a model comprises sets of RAMs which compose discriminators, with each discriminator being responsible for a specific class in a classification problem. All knowledge is stored in those RAM memories, so that inputs are converted to a binary format to serve as RAM addresses. This happens in order to record which addresses were accessed during training or to take the value stored at some address as the output of a neuron. Unlike traditional neural networks, where learning primarily occurs in connection weights, WNNs learning depends on changes in memory contents.

In order to start training, all RAM positions of all discriminators of a given WiSARD instance are initialized to zero, and inputs binarized and then randomly mapped into n-tuples (n is a model hyperparameter). Once formed, these tuples remain fixed throughout the lifespan of each WiSARD instance. Subsequently, examples of each class are presented to the discriminator, and RAM memories are accessed using addresses formed by the corresponding tuples. In the most basic training regime, a value of '1' is then written in the memory position indicated by the address: alternatively, the stored value can be an increment counter [5] or even a timestamp [6], for example. Figure 1a illustrates an example of a toy character recognition problem, in which a hypothetical discriminator records an example of the class 'E' while training.

In the classification phase, new data undergo the same input mapping as in the training phase. Inputs address the RAM memories to read their contents, and if the reading points to a '1', the RAM generates the stored value as output. The discriminator then provides a degree of similarity by summing all the bits generated by each RAM node. The discriminator with the highest value indicates the class to which the input likely belongs, as depicted in Figure 1b.



(a) Training phase (b) Classification phase

Fig. 1: An illustration of WiSARD operation

Despite their remarkable simplicity, RAM-based neural networks are highly effective in pattern recognition tasks, offering rapid training and testing along with adjustable hardware implementation [2] and low energy consumption [20]. However, this technique is sensitive to input quantization, which directly affects the WiSARD model's performance. WNNs have been applied in various fields, including natural language interpretation, image recognition [19], target tracking [7], text mining [22], GPS trajectory classification [3], and traffic sign recognition [11], among others. However, there are currently no references in the literature regarding their application in beam selection for cellular networks.

3 Dataset and Preprocessing

This work uses Raymobtime[1] dataset collection, which was created using a simulation methodology that aims to create realistic data. It uses a set of software to construct these datasets, such as: Cadmapper[2] for generating a location map; OpenStreetMaps[3] for modeling realistic 3D outdoor scenarios; SUMO[4] for mobility simulation of vehicles, pedestrians, drones, etc; and Wireless InSite[5] for ray-tracing simulation. Raymobtime has 13 datasets, S000 through S012 and this work uses

[1] https://www.lasse.ufpa.br/raymobtime/
[2] https://cadmapper.com
[3] https://www.openstreetmap.org/#map=4/-15.13/-53.19
[4] https://sumo.dlr.de/userdoc/
[5] https://www.remcom.com/wireless-insite-em-propagation-software/

the S008 dataset. S008 is composed of 2086 episodes and each episode has 10 mobile receivers and one stationary transmitter with communication in the 60 GHz frequency band. This dataset includes ray-tracing, LiDAR (Light Detection and Ranging), images, and receiver's position. This work makes use of the following context information for improving beam selection:

- receiver's position. Composed by 9 features: valid channel indicator; episode, scene, and receiver IDs; Vehicle's name; X, Y, and Z Coordinates; and LOS (Line of Sight) or NLOS (Non Line of Sight) indicator;
- LiDAR sensor. Collection of x, y and z coordinates in every episode. This collection represents a cloud of points obtained by the LiDAR sensor. The information is mapped as a binary grid in a 3D histogram, where, 1 correspond to obstacle and 0 to non obstacle;
- Signal characteristics (ray-tracing). Composed by 8 parameters: received power, time of arrival, elevation and azimuth angles of departure, elevation and azimuth angles of arrival, ray phase, flag "1" for an LOS ray, and "0" for NLOS ray.

Concerning the ray-tracing, the Wireless InSite tool provides for each pair Tx-Rx the L rays with the larger gain magnitude. From the information about these rays, it is possible to estimate the channel between each pair of devices by using the mmWave geometric channel model as follows:

$$\boldsymbol{H} = \sqrt{N_t N_r} \sum_{\ell=1}^{L} \alpha_\ell \boldsymbol{a}_r(\theta_\ell^A, \phi_\ell^A) \boldsymbol{a}_t^*(\theta_\ell^D, \phi_\ell^D) \quad , \tag{1}$$

where, α_ℓ is the complex gain of all users in the ℓ-th path, θ_ℓ^A and θ_ℓ^D is the elevation of the AoA (Angle of Arrival) and the AoD (Angle of Departure), ϕ_ℓ^A and ϕ_ℓ^D is the azimuth of the AoA and the AoD of the ℓ-th ray, respectively, $\boldsymbol{a}_r(\theta_\ell^A, \phi_\ell^A)$ is the beamforming vector of Rx, and, $\boldsymbol{a}_t^*(\theta_\ell^D, \phi_\ell^D)$ is the conjugate transpose of the beamforming vector of Tx, and, N_t and N_r are the number of transmitter and receiver antennas, respectively [24].

The received signal when the (t,r) beam pair is used can be written as:

$$y_{(t,r)} = \boldsymbol{f}_t \boldsymbol{H} x \boldsymbol{w}_r^* + \boldsymbol{w}_r n \quad , \tag{2}$$

where, $\boldsymbol{H}$ represents the channel matrix between Tx-Rx, x is the transmitted signal, $\boldsymbol{f}$ is the precoder vector of Tx, $\boldsymbol{w}$ is the combining vector of Rx, and $n \sim N_c(0, \sigma_n^2 I)$ is the noise. We assume beam codebooks fixed. $C_t = \{f_1, ..., f_{|C^t|}\}$ at the Tx and $C_r = \{w_1, ..., w_{|C^r|}\}$ at the Rx. And, considering the set of all possible beam pairs (t,r) into a unique index $i \in \{1, 2, 3, ..., m\}$, where $m \leq |C_t||C_r|$, $|C_t|$ is the number of Tx codebook elements and $|C_r|$ is the number of Rx codebook elements, the objective is to select the optimal beam pair index i that maximizes the received signal:

$$i = \underset{i \in \{1,2,3,...,m\}}{\operatorname{argmax}} (y_i) \quad . \tag{3}$$

In classical beam selection, such as the approach defined in the IEEE 802.11ad and 5G-NR standards, the transmitter and receiver sweep all beam pairs. Although an exhaustive search through all candidate options ensures the beam alignment, this task quickly increases the complexity because depends on the geometry and number of antenna elements used. A strategy proposed in the literature is to define a subset k of candidate pair-beams, which are subsequently swept to select the one that maximizes the received signal.

In this work, we used DFT codebooks, both in the Tx and Rx, whose main characteristic is to cover all space around the antenna array. According to the geometry of the array and the number

of antenna elements used, a different set of beamforming vectors is generated for each codebook. In our DFT codebooks, the number of codebook elements is given by the number of antenna elements in the array, i.e. $|C_t| = N_t$ and $|C_r| = N_r$. We used the 25 strongest rays to calculate the MIMO sub-channel for each ray considering 32 elements in the Tx antenna and 8 elements in the Rx antenna with a spacing between elements of half wavelength. We can observe a unbalanced distribution of the pair of beams over the 256 possible combinations (32×8): 50.19% of beams are associated to the indexes between 154 and 159 in the train set, and 61,32% of beams for the test set.

The WiSARD model, detailed in Section 2, is designed to process inputs that are exclusively in binary form. This necessitates a crucial preprocessing step to convert the data into a suitable format to the model. Specifically, when dealing with GPS data, which is represented by floating-point numbers, a transformation is required to convert these values into binary patterns.

Some of the binary encoding methods found in literature include thresholds, mean threshold, Marr–Hildreth filter, Laplacian filter, and thermometer [13]. However, since most of these filters are optimized for detecting edges in images, they are not suitable for this scenario. Therefore, we have employed the thermometer, which utilizes multi-bit unary encoding, comparing features against progressively increasing thresholds. It is not unique to our study but is also adopted in tackling similar challenges [17]. It effectively translates the continuous values of GPS data into a binary format, enabling the WiSARD model to interpret and process the information accurately.

In this approach, the thermometer encoding represents the relative distance between a given coordinate and the minimum coordinate value. Specifically, when the receiver position aligns with the minimum coordinate (and assuming a resolution of 1), only a single bit is activated to signify this minimum value. Conversely, if the receiver is positioned at the maximum coordinate, all bits within the encoding are activated, as illustrated in Figure 2. This encoding process is applied independently to both the x and y coordinates. As a result, the binary representation for each coordinate varies in size, capturing the unique spatial relationships inherent to the dataset.

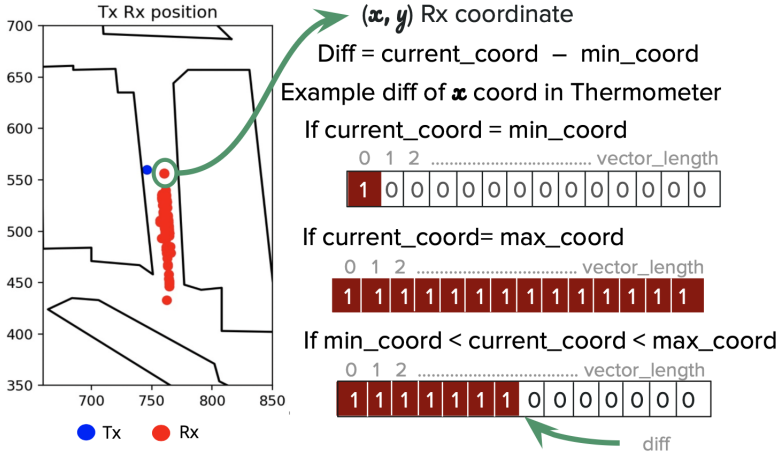


Fig. 2: GPS data preprocessing

The choice of resolution for thermometer encoding critically influences the WiSARD model's effectiveness. As detailed in Section 4, the quest for optimal parameters is pivotal because the input

size to the model increases linearly with resolution. This section explores how varying resolutions impact the model's accuracy, guiding the selection of an ideal resolution that balances performance with resource utilization.

The LiDAR data undergoes an initial quantization step to make it compatible with our model's requirements. To address this, we have devised a unique 3D-thermometer encoding method that captures the receiver's spatial position within the environment. This innovative approach is depicted in Figure 3. Essentially, the input from the LiDAR data includes not just the receiver's exact location but also a comprehensive map of all quantized obstacles within the scene. To prepare this data for our model, we transform the three-dimensional information into a flattened vector by resizing the two matrices that represent the receiver's position and the obstacle map. These vectors are then concatenated to create a single vector that encapsulates both the receiver's location and the surrounding obstacle information.

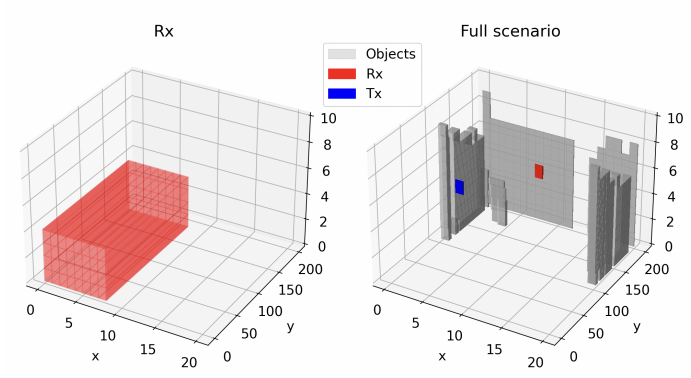


Fig. 3: LiDAR data preprocessing, rx like 3D-thermometer add LiDAR quantized

4 Simulation Results

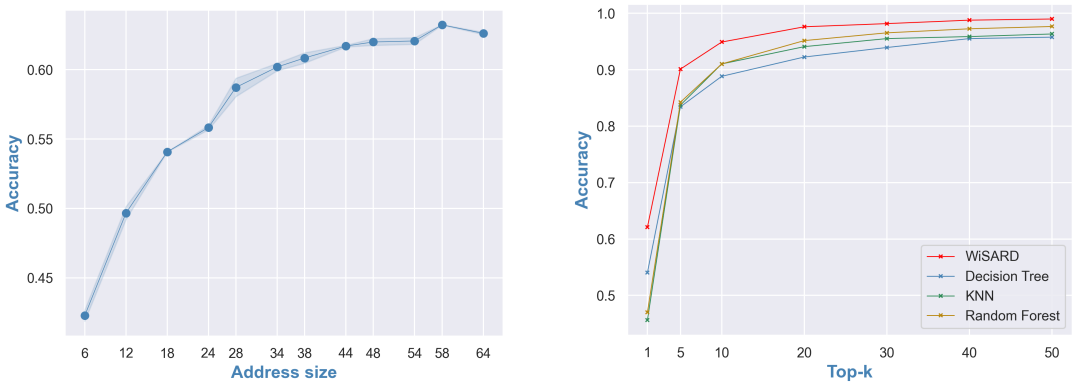
In this paper, we approach the task of beam selection as a classification problem and introduce the use of the WISARD network as a novel machine learning technique for this purpose. Our dataset comprises 20,860 samples, of which 11,194 are deemed valid for experimentation. We allocated 80% of this valid dataset for training and the remaining 20% for testing. For our implementation, we have utilized the wisardpkg[7] library.

4.1 Beam selection using LiDAR data

Hyperparameters tuning: Initially, we have assessed the performance of the behavior network using various address sizes when using LiDAR data only. The input dimension for this dataset is $20 \times 200 \times 10 \times 2$, which suggests 80.000 bits for the WiSARD model's retina input. In this setup,

[7] https://github.com/IAZero/wisardpkg

the WiSARD network identifies the optimal beam pair among 256 possible choices with accuracy superior to 60%, as shown in the Figure 4a.



(a) Behavior of WNN with address size variations

(b) Top-k classification for beam selection

Fig. 4: Performance of WiSARD with LiDAR data only

Comparison: We have conducted a comparative performance analysis of the WiSARD model (using 58 bits of address size) against well-established machine learning algorithms, such as: Decision Trees, Random Forests, and k-nearest neighbors. These specific algorithms were selected for comparison due to their frequent use in the evaluations performed in related works [14,8].

It was necessary to preprocess the data for the WiSARD model since it requires binary input. In contrast, we used real-valued numeric data as input for the remaining models due to better performance with this kind of data.

Our analysis delved into the performance implications of varying 'k' values (Section 3). The network will recommend a subset of top-k pair of beams. For this particular task, k can be 1, 5, 10, 20, 30, 40, or 50 out of the total 256 possible pairs. The transmitter can then use an exhaustive search within this subset to determine the best pair of beams. Furthermore, we have provided a comprehensive overview of the parameters applied across all techniques for processing LiDAR data. These details are summarized in Table 1.

Figure 4b shows the top-k accuracy obtained by the different techniques when using LiDAR data only. Here, the WiSARD model outperforms the other evaluated techniques by an average of 13% for the top k=1 scenario and 3% for the top k=20 scenario.

4.2 Beam selection using GPS coordinates

Hyper-parameters tuning: When using GPS data as input, the coordinates are represented using the thermometer approach. The thermometer resolution not only defines the precision of the input data but also determines the size of the input for each coordinate, as detailed in Table 2.

Also, the accuracy of the model is significantly influenced by the thermometer resolution and address size. This relationship is graphically represented in Figure 5, which displays a heat map of

Technique	Parameters used in models		
	GPS	LiDAR	Type
KNN	40	40	Number of neighbors
SVC	0.1	-	Learning rate
	1	-	c
Decision Tree	70	65	Minimum number of samples at a leaf
	10	15	Maximum depth of the tree
Random Forest	5	7	Number of trees in the forest.
	10	11	Minimum number of samples at a leaf

Table 1: Default parameters of benchmark techniques for LiDAR and GPS data

Resolution		1	2	4	8	16	32	64	128	256	512
Input size	**X coord**	20	40	80	160	320	640	1.280	2.560	5.120	10.240
	Y coord	245	490	980	1.960	3.920	7.840	15.680	31.360	62.720	125.440

Table 2: Impact of the resolution thermometer over the input size

the WiSARD's performance across different settings. Additionally, Figure 6a illustrates the performance along a specific line of the heat map, corresponding to a thermometer resolution of 8. Based on the outcomes of this analysis, we have selected a thermometer resolution of 8 and an address memory size of 34 for the next experiments.

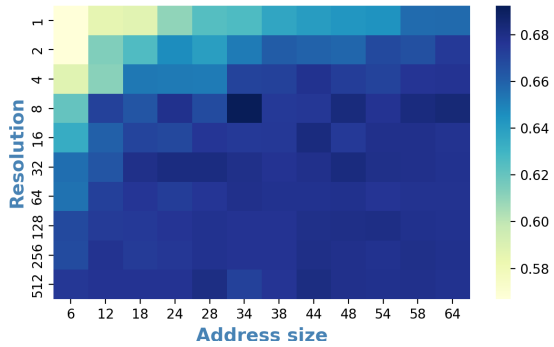
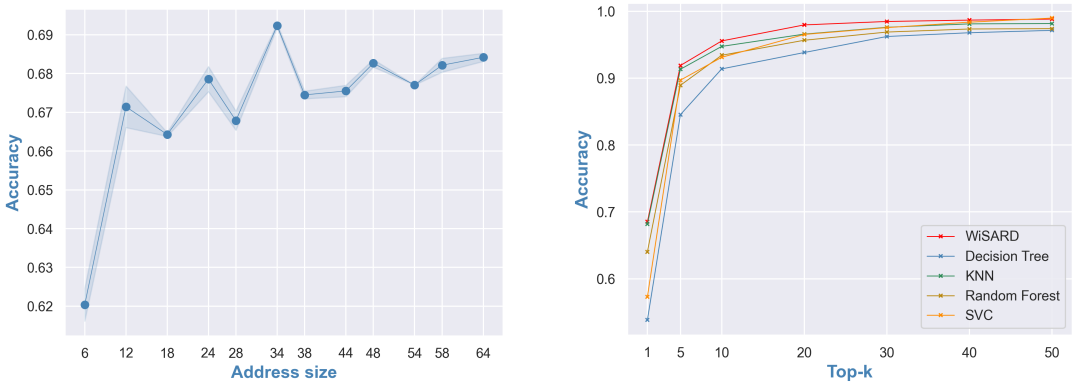


Fig. 5: Heat map of WiSARD accuracy performance

Comparison: Once again, to ensure a fair and consistent comparison with the other approaches, we have selected some values to the parameters and standardized them across all techniques. The specific parameters employed in our analysis are those already detailed in Table 1.

Our evaluation methodology was consistent across all models, focusing on how variations in the 'k' value affect each model's accuracy. This approach allowed for a direct comparison under identical conditions. The results of this comparison are illustrated in Figure 6b.



(a) Behavior of WNN with address size variations

(b) Top-k classification for beam selection

Fig. 6: Performance of WiSARD with GPS data only

The analysis revealed that, on average, the WiSARD model outperformed the other techniques by 7% for the top 'k=1' scenario and by 2% for the top 'k=20' scenario. This indicates that the WiSARD model is effective at predicting the most relevant outcomes when considering a limited number of top choices.

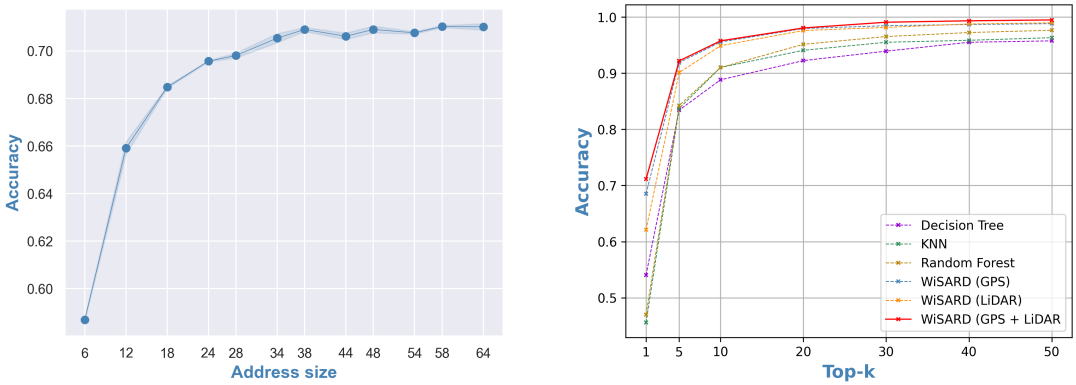
4.3 Beam selection using both GPS and LiDAR data

Hyper-parameters tuning: The integration of GPS and LiDAR data required a careful adjustment of the model's address size to accommodate the merged dataset effectively. During the preprocessing phase, we noted that the LiDAR data consisted of 80,000 bits. To ensure that the WiSARD model assigns equal weights to both GPS and LiDAR inputs, it was crucial to balance the size of the inputs. Consequently, we have adjusted the resolution of the GPS data to 512 bits, resulting in a combined (x,y) input size of 135,680 bits for the GPS data alone. When merged with the LiDAR data, the total input size for the WiSARD's retina increased to 215,680 bits.

Figure 7a illustrates how the WiSARD model's performance varied with different address memory sizes, using this combined input. The results highlight a notable improvement in the WiSARD network's accuracy, achieving scores around 70%. This demonstrates the effectiveness of integrating GPS and LiDAR data and fine-tuning the address size for the combined scenario. The comparison with the other techniques are presented in Figure 7b.

Comparison: The integration of both GPS and LiDAR information yielded a significant improvement in beam prediction accuracy for the top choice (k=1) scenario. Specifically, the accuracy increased by 9% over the use of LiDAR data alone and by 3% over the use of GPS data alone. This comparison emphasizes the value of combining multiple types of context information to bolster the model's performance.

It is important to highlight that the proposed model significantly enhanced the accuracy of beam selection. The greatest accuracy achieved was an impressive 71% (Table 3). This comparison not only emphasizes the effectiveness of our algorithm but also situates our findings within the broader context of current research.



(a) Behavior of NN with address size variations

(b) Top-k classification for beam selection

Fig. 7: Performance of WiSARD with LiDAR and GPS data

Technique	**Input**	**Antenna config.**	**Accuracy [%]**			**Ref**
		[Tx-Rx]	Top-1	Top-10	Top-20	
Linear SVM	GPS	UPA [4X4]	33.2	-	-	[14]
AdaBoost			55.0	-	-	
Decision Tree			55.5	-	-	
Random Forest			63.2	-	-	
DNN			63.8	-	-	
DQN			67.5	-	-	
DNN	LiDAR		20.5	33.0	49.5	[9]
DL	LiDAR GPS		42.0	90.0	98.0	[15]
Conv3D	LiDAR		63.3	92.5	-	[27]
KNN	GPS		43.0	-	-	[8]
Naive Bayes			8.70	-	-	
ANN			40.50	-	-	
Lightweight NN	LiDAR, GPS		59.5	-	87.0	[26]
F-DL	LiDAR	ULA [32X8]	46.2	89.9	96.1	[23]
DNN	LiDAR	UPA [8X72]	32.0	80.0	-	[25]
Dense Neural Network	LiDAR	ULA [64X8]	43.0	82.0	85.0	[16]
WiSARD (this work)	LiDAR, GPS	ULA [32X8]	71.0	95.0	98.0	

Table 3: Performance comparison with ML techniques from state-of-the-art

5 Conclusions and future works

This paper proposes the use of the WiSARD network for beam selection in a mmwave MIMO system. The method was explored with each unimodal feature (GPS and LIDAR data) and both features simultaneously. The simulation results provide a fair comparison between the proposed method and other machine learning techniques commonly used in the literature. It is evident that the WiSARD network outperforms the baseline methods, with a 3% increase in performance compared to state-of-the-art methods for the worst task, top-1.

Overall, the WiSARD network scores are competitive, and since it is a lightweight method in terms of processing, it could be a useful tool for future generations of cellular networks. In addition, the process of selecting the appropriate beam for data transmission can reduce energy consumption in smart mobile devices. For future work, it is recommended to conduct a complexity analysis of the algorithms, an experimental evaluation with real-world data and to assess the feasibility of embedding WiSARD-based algorithms.

Acknowledgments

This work was partially supported by CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), RNP with resources from MCTIC (under the grant 01250.075413/2018-04 from the Radiocommunication Reference Center project of the National Institute of Telecommunications, Brazil) and the MCTI/CGI.br and the São Paulo Research Foundation (FAPESP) (under grants 2018/23097-3 (SFI2), 2020/05127-2 (SAMURAI) and 2020/05152-7 (PROFISSA)). Douglas O. Cardoso acknowledges the financial support by the portuguese Foundation for Science and Technology (Fundação para a Ciência e a Tecnologia, FCT) through grants with the following DOIs: 10.54499/UIDB/00022/2020, 10.54499/UIDP/00022/2020, 10.54499/UIDB/05567/2020, and 10.54499/UIDP/05567/2020.

References

1. Aleksander, I., Gregorio, M.D., França, F.M.G., Lima, P.M.V., Morton, H.: A brief introduction to Weightless Neural Systems. In: 17th European Symposium on Artificial Neural Networks, ESANN 2009, Bruges, Belgium, April 22-24, 2009, Proceedings (2009)
2. Alves, D.S., Cardoso, D.O., Carneiro, H.C., França, F.M., Lima, P.M.: An Empirical Study of the Influence of Data Structures on the Performance of VG-RAM Classifiers. In: 2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence. pp. 388–393 (Sep 2013)
3. Barbosa, R., Cardoso, D.O., Carvalho, D., França, F.M.G.: Weightless neuro-symbolic GPS trajectory classification. Neurocomputing **298**, 100–108 (Jul 2018)
4. Brilhante, D.d.S., Manjarres, J.C., Moreira, R., de Oliveira Veiga, L., de Rezende, J.F., Müller, F., Klautau, A., Leonel Mendes, L., P. de Figueiredo, F.A.: A literature survey on ai-aided beamforming and beam management for 5g and 6g systems. Sensors **23**(9), 4359 (2023)
5. Cardoso, D.O., Carvalho, D.S., Alves, D.S., Souza, D.F., Carneiro, H.C., Pedreira, C.E., Lima, P.M., França, F.M.: Financial credit analysis via a clustering weightless neural classifier. Neurocomputing **183**, 70–78 (Mar 2016)
6. Cardoso, D.O., França, F.M.G., Gama, J.: WCDS: A Two-Phase Weightless Neural System for Data Stream Clustering. New Generation Computing **35**(4), 391–416 (Oct 2017)
7. Cardoso, D.O., Gama, J., Gregorio, M.D., França, F.M.G., Giordano, M., Lima, P.M.V.: WIPS: the wisard indoor positioning system. In: 21st European Symposium on Artificial Neural Networks, ESANN 2013, Bruges, Belgium, April 24-26, 2013 (2013)
8. Chatzoglou, E., Goudos, S.K.: Beam-Selection for 5G/B5G Networks Using Machine Learning: A Comparative Study. Sensors **23**(6), 2967 (Jan 2023)
9. Dias, M., Klautau, A., González-Prelcic, N., Heath, R.W.: Position and LIDAR-Aided mmWave Beam Selection using Deep Learning. In: 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC). pp. 1–5 (Jul 2019)

10. Ferreira, V.C., Nery, A.S., Marzulo, L.A.J., Santiago, L., Souza, D., Goldstein, B.F., França, F.M.G., Alves, V.: A Feasible FPGA Weightless Neural Accelerator. In: 2019 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 1–5 (May 2019)
11. Forechi, A., De Souza, A.F., Oliveira Neto, J.d., Aguiar, E.d., Badue, C., d'Avila Garcez, A., Oliveira-Santos, T.: Fat-Fast VG-RAM WNN: A high performance approach. Neurocomputing **183**, 56–69 (Mar 2016)
12. GSM Association: The mobile economy 2022, https://tinyurl.com/3u5426yx, visited on 2024-02-29
13. Kappaun, A., Camargo, K., Rangel, F., Firmino, F., Lima, P.M.V., Oliveira, J.: Evaluating Binary Encoding Techniques for WiSARD. In: 2016 5th Brazilian Conference on Intelligent Systems (BRACIS). pp. 103–108 (Oct 2016)
14. Klautau, A., Batista, P., González-Prelcic, N., Wang, Y., Heath, R.W.: 5G MIMO Data for Machine Learning: Application to Beam-Selection Using Deep Learning. In: 2018 Information Theory and Applications Workshop (ITA). pp. 1–9 (Feb 2018)
15. Klautau, A., González-Prelcic, N., Heath, R.W.: LIDAR Data for Deep Learning-Based mmWave Beam-Selection. IEEE Wireless Communications Letters **8**(3), 909–912 (Jun 2019)
16. Klautau, A., de Oliveira, A., Pamplona Trindade, I., Alves, W.: Generating Mimo Channels for 6G Virtual Worlds Using Ray-Tracing Simulations. In: 2021 IEEE Statistical Signal Processing Workshop (SSP). pp. 595–599 (Jul 2021)
17. Lello, G.C.D., Caldeira, J.F., Aredes, M., França, F.M.G., Lima, P.M.V.: Weightless Neural Networks Applied to Nonintrusive Load Monitoring. In: 2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). pp. 844–851 (May 2020)
18. Ludermir, T.B., Carvalho, A.C.P.d.L.F.d., Braga, A.P., Souto, M.C.P.d.: Weightless neural models: a review of current and past works. Neural Computing Surveys (1999)
19. Lyrio, L.J., Oliveira-Santos, T., Badue, C., De Souza, A.F.: Image-based mapping, global localization and position tracking using VG-RAM weightless neural networks. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 3603–3610 (May 2015)
20. Miranda, I.D., Arora, A., Susskind, Z., Villon, L.A., Katopodis, R.F., Dutra, D.L., De Araújo, L.S., Lima, P.M., França, F.M., John, L.K., Breternitz, M.: Logicwisard: Memoryless synthesis of weightless neural networks. In: 2022 IEEE 33rd International Conference on Application-specific Systems, Architectures and Processors (ASAP). pp. 19–26 (2022). https://doi.org/10.1109/ASAP54787.2022.00014
21. Rahimi, M., Singh, H., Prasad, R.: Mm-waves Promises and Challenges in Future Wireless Communication: 5G. In: 5G Outlook – Innovations and Applications. River Publishers (2016)
22. Rangel, F.M., de Faria, F.F., Lima, P.M.V., Oliveira, J.: Semi-supervised classification of social textual data using wisard. In: 24th European Symposium on Artificial Neural Networks, ESANN 2016, Bruges, Belgium, April 27-29, 2016 (2016)
23. Salehi, B., Reus-Muns, G., Roy, D., Wang, Z., Jian, T., Dy, J., Ioannidis, S., Chowdhury, K.: Deep Learning on Multimodal Sensor Data at the Wireless Edge for Vehicular Network. IEEE Transactions on Vehicular Technology **71**(7), 7639–7655 (Jul 2022)
24. Va, V., Choi, J., Shimizu, T., Bansal, G., Heath, R.W.: Inverse Multipath Fingerprinting for Millimeter Wave V2I Beam Alignment. IEEE Transactions on Vehicular Technology **67**(5), 4042–4058 (May 2018)
25. Xu, W., Gao, F., Jin, S., Alkhateeb, A.: 3D Scene-Based Beam Selection for mmWave Communications. IEEE Wireless Communications Letters **9**(11), 1850–1854 (Nov 2020)
26. Zecchin, M., Mashhadi, M.B., Jankowski, M., Gündüz, D., Kountouris, M., Gesbert, D.: LIDAR and Position-Aided mmWave Beam Selection With Non-Local CNNs and Curriculum Training. IEEE Transactions on Vehicular Technology **71**(3), 2979–2990 (Mar 2022)
27. Zheng, Y., Chen, S., Zhao, R.: A Deep Learning-Based mmWave Beam Selection Framework by Using LiDAR Data. In: 2021 33rd Chinese Control and Decision Conference (CCDC). pp. 915–920 (May 2021)