

# Trajectory Design for RSMA Networks Assisted by AIRS with LSTM and Transformers

Brena Kelly S. Lima<sup>⊙\*</sup>, João P. Matos-Carvalho<sup>⊙\*†</sup>, Rui Dinis<sup>⊙‡</sup>, Daniel Benevides da Costa<sup>⊙¶</sup>,  
Marko Beko<sup>⊙\*†||</sup> and Rodolfo Oliveira<sup>⊙‡</sup>

<sup>\*</sup>COPELABS, Lusófona University, Campo Grande 376, 1749-024 Lisbon, Portugal

<sup>†</sup>Center of Technology and Systems (UNINOVA-CTS) and LASI, 2829-516 Caparica, Portugal

<sup>‡</sup>Instituto de Telecomunicações, Department of Electrical Engineering, NOVA University of Lisbon, Caparica, Portugal

<sup>¶</sup>King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia

<sup>||</sup>Instituto de Telecomunicações, Instituto Superior Técnico, University of Lisbon, Portugal

**Abstract**—This paper investigates rate-splitting multiple access (RSMA) networks assisted by aerial intelligent surfaces (AIRS) by employing deep-learning approaches to solve trajectory problems for unmanned aerial vehicles (UAVs). Specifically, two models for predicting positions using long-short term memory (LSTM) and Transformers are developed. Training results show that both proposed frameworks can capture temporal features to determine the UAV’s position for tracking user mobility. However, simulation results indicate that the proposed Transformer-based model demonstrates robustness against variations in user locations, providing superior prediction accuracy and consequently yielding higher performance gains in terms of sum rate when compared with the LSTM-based model. Additionally, it is demonstrated that the AIRS-RSMA scheme outperforms AIRS-NOMA systems due to its ability to effectively handle residual successive interference cancellation (SIC) errors.

**Index Terms**—Intelligent reflecting surface (IRS), long-short term memory (LSTM), rate-splitting multiple access (RSMA), trajectory design, transformers, unmanned aerial vehicle (UAV).

## I. INTRODUCTION

Unmanned Aerial Vehicles (UAV)-assisted wireless networks have gained significant attention for providing wider coverage and reliable communication links to ground devices due to their high mobility, cost-effectiveness solution, ability to perform complex tasks in different scenarios, and flexible configuration [1]. In a parallel avenue, intelligent reflecting surfaces (IRSs) have been viewed as an effective technology to customize the propagation environment. In particular, an IRS represents a sophisticated solution, comprising a planar metasurface replete with large reflective elements. These elements can be intricately programmed, operating independently to manipulate both phase and amplitude to reflect and steer impinging waves towards any desired direction [2].

Recently, several studies have exploited UAVs integrated with IRSs [3]–[5]. By exploiting this integration, it is possible to achieve numerous benefits compared to terrestrial deployment. In particular, [5] provides a comprehensive discussion

about the integration of IRS and UAV, called aerial IRS (AIRS) non-orthogonal multiple access (NOMA) networks, as well as its architecture, functionality principles, and performance gains.

Considering the intrinsic requirements expected in beyond-5G networks, multiple access techniques play a crucial role in supporting connectivity demands and spectrum management. In particular, the rate-splitting multiple access (RSMA) scheme has received significant interest as a promising enabling technique in beyond-5G networks, owing to its resilient transmission framework and attractive performance enhancements [6], [7]. The interplay between IRS and RSMA can provide numerous benefits to the networks [8]. Consequently, the integration of AIRS with RSMA systems can further improve the performance of the wireless networks as compared to the terrestrial IRS (TIRS) framework [9].

In response to the complexities of dynamic aerial environments, extensive research has been conducted in the field of machine learning (ML) over the years. To learn and extract resources from the dynamic environment, the long short-term memory (LSTM) [10] and Transformers [11] techniques have been explored due to their features to predict sequential patterns by recursively forwarding the previous information to the memory.

Despite the numerous studies carried out, there remains a research gap regarding the application of LSTM and/or Transformers in networks that specifically investigate the integration of AIRS in RSMA systems. In contrast to existing works, this study addresses the trajectory problem and proposes two frameworks based on LSTM and Transformers, respectively, to predict the trajectory of UAVs for a downlink AIRS-RSMA multiuser network, exploring scenarios with dynamic users. A performance analysis, utilizing training and test results, is presented to demonstrate the effectiveness of the proposed schemes. Additionally, a detailed comparison of the frameworks is carried out, wherein the Transformers outperform LSTM in terms of sum rate. This is because models using the attention layer, like Transformers, can establish a greater relationship between past values (context) to more accurately predict the future.

This work is funded by Fundação para a Ciência e Tecnologia under the project UIDB/50008/2020, UIDB/04111/2020, CELL-LESS6G 2022.08786.PTDC, CEECINST/00147/2018/CP1498/CT0015, European Union’s Horizon Europe Research and Innovation Programme under the Marie Skłodowska-Curie grant agreement No. 101086387, as well as Instituto Lusófono de Investigação e Desenvolvimento (ILIND) under Project COFAC/ILIND/COPELABS/1/2022.

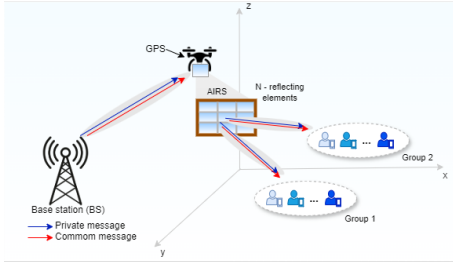


Fig. 1: Illustration of a system model: AIRS-RSMA network with two user groups.

## II. SYSTEM MODEL

Consider a downlink MISO network in which a BS with  $M$  antennas communicates with  $U$  single-antennas users, in which  $m = 1, \dots, M$ . It is assumed that there is no direct link between the BS and users. Then, an aerial IRS (AIRS) with  $N$  reflecting elements, with  $n = 1, \dots, N$ , is employed to enable the communication between the BS and users, as depicted in Fig. 1. It is assumed that the users are distributed within a region of radius  $R$  and clustered into  $G$  groups based on their minimum distance from the neighbor user. For simplicity, each group contains  $K$  users, i.e.,  $U = GK$ . In addition, it is assumed that the UAV continuously flies at a fixed altitude  $H$ .

Based on three-dimensional Cartesian coordinates, the location of the BS, AIRS, and  $k$ -th user within the  $g$ -th group at the  $t$  time step can be represented, respectively, by  $(x_0, y_0, z_0)$ ,  $(x^t, y^t, H)$ , and  $(x_{gk}^t, y_{gk}^t, 0)$ . Based on this, the distances between BS→AIRS,  $d_n^t$ , and AIRS→ $k$ -th user within the  $g$ -th group,  $d_{gk}^t$ , can be given, respectively, by  $d_n^t = \sqrt{(x_0 - x^t)^2 + (y_0 - y^t)^2 + (z_0 - H)^2}$ , and  $d_{gk}^t = \sqrt{(x^t - x_{gk}^t)^2 + (y^t - y_{gk}^t)^2 + (H)^2}$ .

It is important to mention that the mobility of the users is taken into account, where for each moment  $t$ , the users assume a new position. Based on this, the scenario will be referred by dynamic users.

Since the channels between BS→AIRS and AIRS→ $k$ -th user can be seen as LoS (line-of-sight) links, their channel power gain can be expressed as

$$\mathbf{H}_{mn} = \sqrt{\beta_{mn}} \mathbf{G}_{mn} \in \mathbb{C}^{M \times N}, \quad (1)$$

and

$$\mathbf{h}_{gk} = \sqrt{\beta_{gk}} \mathbf{g}_{gk} \in \mathbb{C}^{N \times 1}, \quad \forall k \in \mathcal{K}, \quad (2)$$

where  $\beta_{mn} = \beta_0 d_n^{-\nu}$  and  $\beta_{gk} = d_{gk}^{-\nu}$  denote, respectively, the large-scale average channel power gain between BS→AIRS and AIRS→ $k$ -th user, in which  $\beta_0$  symbolizes the gain parameter. The parameter  $d_j$ , with  $j \in \{n, gk\}$ , means the distance between BS→AIRS and AIRS→ $k$ -th user, and  $\nu$  refers to the pathloss exponent. In addition,  $\mathbf{G}_{mn}$  and  $\mathbf{g}_{gk}$  denote the small-scale fading, modeled by the Nakagami distribution.

It is assumed that the AIRS is equipped with a controller that can adjust the AIRS's phase shifts. To characterize the phase shifts, it is considered that the phase-shift matrix of the AIRS is diagonal and can be written as  $\Phi = \text{diag}[\kappa_1 e^{j\theta_1}, \kappa_2 e^{j\theta_2}, \dots, \kappa_N e^{j\theta_N}] \in \mathbb{C}^{N \times N}$ , denotes the phase-shift occurring at  $n$ -th element of the AIRS, in which  $|\kappa_l e^{j\theta_n}| = 1$  for  $\theta_n \in [0, 2\pi)$  and  $n \in \mathcal{N}$ , and  $\kappa_n$  refers to the fixed amplitude reflection coefficient. We assume that  $\kappa_n = 1$  with  $n = 1, \dots, N$ , as commonly adopted in the literature, i.e., [12]. For practical implementation purposes, we consider that discrete phase shifts uniformly quantizing the interval  $[0, 2\pi)$  are adopted at each element of the AIRS, which can be given as [13]  $\theta_n \in \left\{ \frac{2\pi i}{2^{\check{b}}}, i = 0, 1, 2, \dots, 2^{\check{b}} - 1 \right\}$ ,  $\forall n \in \mathcal{N}$ , where  $\check{b}$  denotes the bit resolution. Based on that, the effective end-to-end (e2e) channel gain from the BS to  $k$ -th user of the  $g$ -th group via AIRS is expressed by

$$\mathbf{h}_k = \mathbf{H}_{mn} \Phi \mathbf{h}_{gk}^H \in \mathbb{C}^{M \times 1}. \quad (3)$$

Since the RSMA strategy is employed to transmit data to the users, the message intended for  $u$ -th user is split into two messages: common and private [6]. Based on that, the transmission can be viewed as two-stage process. The first one consists of separating the data to serve multiple groups of users, then the stream is sent to serve the user within each group. In order to achieve this, the messages intended for users of a specific group are encoded in a common stream  $s_g^c = (s_{g1}^c, \dots, s_{gK}^c)$ , using a codebook shared by the  $k$ -th user of the  $g$ -th group, while the private parts are encoded independently into private streams  $s_{gk}^p$ . At the transmitter, the data streams are linearly precoded and superimposed in the power domain, then simultaneously transmitted. As a result, the signal transmitted by the BS can be expressed as

$$\mathbf{x} = \sum_{g=1}^G \left( \mathbf{c}_g \sqrt{\alpha_g^c} s_g^c + \sum_{k=1}^K \mathbf{p}_{gk} \sqrt{\alpha_{gk}^p} s_{gk}^p \right) \in \mathbb{C}^{N \times 1}, \quad (4)$$

where the vector of symbol streams to be transmitted to the  $g$ -th group can be denoted by  $\mathbf{s}_g = [s_g^c \alpha_g^c, s_{g1}^p \alpha_{g1}^p, \dots, s_{gK}^p \alpha_{gK}^p]^T \in \mathbb{C}^{(K+1) \times 1}$ , where  $\alpha_g^c$  and  $\alpha_{gk}^p$  are, respectively, the power allocation coefficients for the common and private messages in which  $\alpha_g^c + \sum_{k=1}^K \alpha_{gk}^p = 1$ ,  $\mathbf{c}_g \in \mathbb{C}^{N \times 1}$  denotes the precoding vector of the common message serving all users within  $g$ -th group and  $\mathbf{p}_{gk} \in \mathbb{C}^{N \times 1}$  represents the precoding vector of the private message for the  $k$ -th user of the  $g$ -th group, in which  $\|\mathbf{c}_g\| = 1$  and  $\|\mathbf{p}_{gk}\| = 1$ . For notation simplicity, the precoding vectors intended for the  $g$ -th group are reexpressed by  $\mathbf{W}_g = [\mathbf{c}_g, \mathbf{p}_{g1}, \dots, \mathbf{p}_{gK}] \in \mathbb{C}^{N \times (K+1)}$ , where  $\mathbf{W}_g$  stands for the precoding matrix of the  $g$ -th group. The conventional zero-forcing (ZF) precoder is considered.

Without loss of generality, the AIRS can only serve one group within each time slot to mitigate inter-group interference. Then, at the users' side, the SIC technique is used once in each receiver to separate the common and private messages. The common message is detected first, considering the private

message of the other users of the  $g$ -th group as interference. Based on this, the corresponding signal-to-interference-plus-noise ratio (SINR) to decode the common message at the  $k$ -th user of the  $g$ -th group is given by

$$\gamma_{gk}^c = \frac{\rho |\mathbf{h}_k \mathbf{c}_g|^2 \alpha_g^c}{\rho \sum_{i=1}^K |\mathbf{h}_k \mathbf{p}_{gi}|^2 \alpha_{gi}^p + 1}, \quad (5)$$

where  $\rho$  means the transmit signal-to-noise ratio (SNR). Since imperfect SIC is considered and the common message is not decoded perfectly, the  $k$ -th user of the  $g$ -th group must decode its private message taking into account the other users' private messages and the common message as interference. Thus, the SINR to decode the private message at the  $k$ -th user of the  $g$ -th group is given by

$$\gamma_{gk}^p = \frac{\rho |\mathbf{h}_k \mathbf{p}_{gk}|^2 \alpha_{gk}^p}{\rho |\mathbf{h}_k \mathbf{c}_g|^2 \alpha_g^c \xi + \rho \sum_{i \neq k} |\mathbf{h}_k \mathbf{p}_{gi}^p|^2 \alpha_{gi}^p + 1}, \quad (6)$$

in which  $\xi \in [0, 1]$  denotes the coefficient of imperfect SIC. Therefore, the instantaneous achievable rate of the  $k$ -th user of the  $g$ -th group in decoding the common message can be expressed as  $R_{gk}^c = \log_2(1 + \gamma_{gk}^c)$ . Nevertheless, the achievable rate of the  $k$ -th user in decoding its intended private message is given by  $R_{gk}^p = \log_2(1 + \gamma_{gk}^p)$ .

To ensure that all users of the  $g$ -th group can successfully decode the common message, the rate of the common message should be formulated as  $\bar{R}_{gk}^c = \min_{k \in \mathcal{K}} \{R_{gk}^c\}$ . Finally, based on the achievable common rate  $\bar{R}_{gk}^c$  and achievable private rate  $R_{gk}^p$ , the total achievable rate of the  $k$ -th user can be expressed as

$$R_{gk}^{\text{tot}} = \bar{R}_{gk}^c + R_{gk}^p. \quad (7)$$

Next, it is formulated a trajectory design for the proposed AIRS-RSMA network.

### III. DEEP LEARNING-BASED TRAJECTORY DESIGN

#### A. Basics of LSTM

LSTM is part of a group of architectures called RNN, which are a family of neural networks widely used for models whose problems are characterized by data sequences [14], [15]. These two types of networks differ from other neural networks since the network parameters are adopted along several points of the sequence under analysis. The outputs of the previous elements influence each one of the output elements. This particularity of persistence of the characteristics present in the data allows the model to look at a sequence as a whole and not just at an individual element. Unlike other networks, RNNs use feedback from activation's from previous time slots as input from the network to make a decision for the current input. It should be noted that the sequences that the recurrent neural network (RNN) analyzes correspond to a time interval  $\delta$ , which may not refer precisely to the phenomenon of time duration in the real world.

Given the example of an LSTM cell and the complete internal structure illustrated in Fig. 2, where the input gate, the output, and the forget gate are represented by  $\hat{i}^{(t)}$ ,  $\hat{o}^{(t)}$  and  $\hat{f}^{(t)}$  respectively, and  $\hat{x}$ ,  $\hat{C}$  and  $\hat{h}$ , correspond to the input,

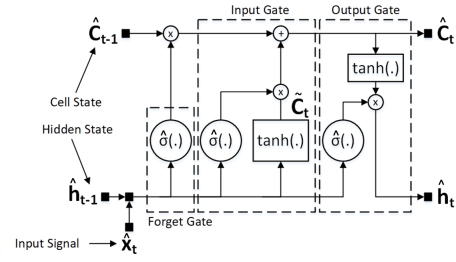


Fig. 2: Internal structure of the LSTM cell. Adapted from [16].

state and output of the current cell and the previous cell, that is, the time stamp  $t - 1$ . The variable  $\tilde{C}$  represents the vector of new values that can be added to the state  $\hat{C}$ . Then the final state  $\hat{C}$  of the previous LSTM cell may or may not be affected by the output of each of the three gates, creating a new state of the current cell.

#### B. Basics of Transformers

The transformers represent a neural architecture that has stood out for its ability to efficiently handle sequences of data, becoming essential for complex tasks such as automatic translation, text summarization, and natural language processing in general. The transformer architecture was initially introduced by [17] in 2017, marking a paradigm shift in how language models were designed and trained.

At the core of the [17] architecture lies multihead attention, a technique that allows the model to focus on different parts of the input simultaneously. This provides a richer understanding of complex contexts, enabling the model to capture long-range relationships between words in a sequence. The absence of direct sequential dependence, present in previous architectures, contributes to remarkable computational efficiency.

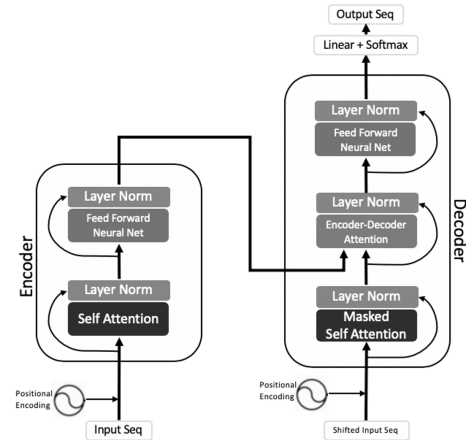


Fig. 3: Transformer Architecture. Adapted from [17].

As can be seen in Fig. 3 there are two core parts: encoder (left side) and decoder (right side) blocks.

The encoder blocks are responsible for processing the input sequence. It consists of multiple identical layers (referred to

as encoder layers), where each encoder block has two main sub-components:

- 1) Multi-Head Self-Attention Mechanism: This mechanism allows the encoder to focus on different parts of the input sequence simultaneously. It computes attention scores for each position in the input sequence, capturing relationships between different tokens;
- 2) Feedforward Neural Network: After the attention mechanism, the output is passed through a feedforward neural network. This helps capture complex, non-linear relationships in the data.

Both sub-components (attention and feedforward) have a residual connections to allow the model to learn incremental changes, and a layer normalization to apply to the sum of the input and the output of each sub-component.

The decoder block is responsible for generating the output sequence. Like the encoder, it consists of multiple identical layers (decoder layers), where each decoder block has three main sub-components:

- 1) Masked Multi-Head Self-Attention Mechanism: Similar to the encoder's attention mechanism, but with a mask applied to prevent attending to future positions in the sequence during training. This is crucial for autoregressive generation;
- 2) Encoder-Decoder Attention Mechanism: This attention mechanism allows the decoder to attend to different parts of the input sequence. It helps the model align the input and output sequences effectively;
- 3) Feedforward Neural Network: Similar to the encoder, the output from the attention mechanisms is passed through a feedforward neural network.

The tree main sub-components, as in the encoder, are followed by a residual connection and layer normalization.

For this work, only the decoder block was considered, following a similar approach to the Generative Pre-trained Transformer (GPT) [18].

### C. Trajectory Prediction Process

The observation space used to train the network is composed of the location of the UAV and the sum-rate experienced by each group at the  $t$ -th interval of time. Since the training process is performed by considering two groups, the observation space for trajectory optimization can be expressed as

$$S^\dagger = \left[ x^t, y^t, \sum_{k=1}^K R_{1k}^{\text{tot},t}, \sum_{k=1}^K R_{2k}^{\text{tot},t} \right]. \quad (8)$$

As shown in Fig. 1 the  $S^\dagger$  is represented by the sensor data where there is a GPS and also an AIRS (both are attached to the UAV). The information from these sensors will feed two AI algorithms for trajectory prediction: the LSTM and Transformers deep learning algorithms. To predict the UAV next position, it receives the UAV's current position and the rate of each user group and returns the next UAV position to be reached. To generate the UAV's next position, the parameters that were used to train both networks are shown in Tables I

and II, where  $N_n$  denotes the number of neurons in each layer, the dimension of the Transformer input layer is equal to 4, which represents the current 2D UAV position ( $x^t$ , and  $y^t$ ) and the sum-rate experienced by each group. The model was implemented using Matlab with Deep Learning Tool Box associated. The model was also trained on a laptop with NVIDIA GeForce GTX 1060, 16GB RAM and Intel Core i7-8750H CPU @ 2.20GHz x 12.

TABLE I: The proposed custom RNN architecture hyperparameters for AI trajectory model.

Layer	Output	Hyperparameters
Input	4x1	
LSTM_0	600	$L_n = 1452000$ ;
LSTM_1	600	$L_n = 2882400$ ;
FC Output	2	$L_n = 1202$ ; optm=ADAM loss=Mean Square Error

TABLE II: The proposed custom Transformer architecture hyperparameters for AI trajectory model.

Layer	Output	Hyperparameters
Input	4x1	
Embedding	768	$L_n = 3840$ ;
Positional Encoding	1x768	$L_n = 15360$ ;
Decoder Block_0	768	$L_n = 7085568$ ;
Decoder Block_1	768	$L_n = 7085568$ ;
Decoder Block_2	768	$L_n = 7085568$ ;
Decoder Block_3	768	$L_n = 7085568$ ;
Decoder Block_4	768	$L_n = 7085568$ ;
Normalization Layer	768	$L_n = 1536$ ;
FC Output	2	$L_n = 1536$ ; optm=ADAM loss=Mean Square Error

With regard to the transformer model, it is necessary to reinforce some important hyperparameters, namely:

- Context Length: 2;
- Embedding dimension: 768;
- Number of heads: 12;
- Number of encoders blocks: 5;
- Drop rate: 0.1.

## IV. RESULTS

In this section, simulation results are provided to evaluate the performance of the proposed method. It is considered a scenario with  $N = 4$  antennas at the BS and  $K = 3$  single-antenna users per group,  $G = 2$ , which are positioned in a cell of radius of 100 m. The UAV continuously flies, following the fixed height  $H = 20$  m, and an AIRS with  $N = 60$  reflective elements is fitted to the UAV. In particular, it is assumed  $\ddot{b} = 2$ ,  $\nu = 2.2$ ,  $\alpha_g^c = 0.9$ ,  $\alpha_{gk}^p = 0.03 \forall k \in \mathcal{K}$ , and  $\beta_0 = 1000$ . The parameter  $\beta_0$  is adjusted based on the desired performance of the receivers. For AIRS-NOMA scenario, it is considered  $\alpha_{g1} = 0.7$ ,  $\alpha_{g2} = 0.2$ , and  $\alpha_{g3} = 0.1$ . When not specified, the value for SNR is set as 30 dB. In addition, it is assumed that the power consumption of IRS is 3 mW per element.

### A. Training Progress

One of the crucial phases in the development of this system is the training phase for the models proposed in Sections III-A and III-B, as shown in Algorithm 1.

---

**Algorithm 1:** The proposed training LSTM and Transformer models phase.

---

**Require:**  $e = 1, \dots, N$ ,  $LSTM_1 = 600$ ,  $LSTM_2 = 600$ ,

- 1: //Build the Deep Learning Model
- 2: **for**  $e = 1, \dots, N$  **do**
- 3: //LSTM Model
- 4:  $model_{RNN}[0] \leftarrow input$
- 5:  $model_{RNN}[1:2] \leftarrow LSTM_1, LSTM_2$
- 6:  $model_{RNN}[3] \leftarrow FNN$
- 7: //Transformer Model
- 8:  $model_{TFR}[0] \leftarrow input$
- 9:  $model_{TFR}[1] \leftarrow DecoderBlock$
- 10:  $model_{TFR}[2] \leftarrow FNN$
- 11: //Compile Models
- 12:  $model_{RNN}, model_{TFR} \leftarrow Loss = MSE$
- 13:  $model_{RNN}, model_{TFR} \leftarrow Optimizer = Adam$
- 14: //Train Models
- 15:  $model_{RNN}, model_{TFR} \leftarrow Dataset$
- 16: //Save models evaluations scores
- 17:  $results_{RNN} \leftarrow model_{RNN}.save$
- 18:  $results_{TFR} \leftarrow model_{TFR}.save$
- 19: **end for**

---

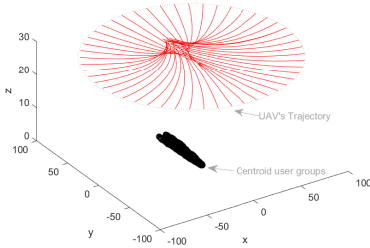


Fig. 4: Generating the observation space.

By exploring the scenario, Fig. 4 depicts the AIRS's trajectory in the training process based on the learning policy to find the centroid of the static user groups, as described in Sections III-A and III-B. In this figure, one can see that the UAV explores 30k starting points. The AIRS starts from one of these points and, at each time  $t$ , recalculates the distance to the center of the user groups based on linear equation. The coefficients of the equation of line are updated at each time  $t$ . Then, an observation space is generated by considering the data obtained for each starting point, as expressed in (8). The observation spaces are then used as input to the LSTM and transformers networks to predict the trajectory.

Fig. 5 shows the relationship between the loss value (y-axis) and each step (x-axis) during models training. Despite

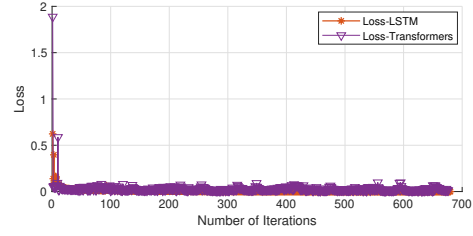


Fig. 5: Loss training results for trajectory model with the dynamic group movement.

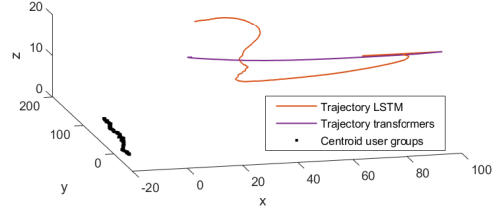


Fig. 6: Predicted trajectory.

the generic overlap in the loss values of both models, the LSTM model converges faster than the transformer model to predict the UAV trajectory. This is due to the complexity of the transformer network's values, where the LSTM model has around 4.3 million of learnable parameters, and the transformer model has approximately 35.5 million learnable parameters to optimize, which is 8 times larger than the LSTM model [16].

### B. Performance Analysis

Fig. 6 depicts the UAV trajectory obtained by applying LSTM and Transformers to predict the UAV's position. One can see that the Transformer-based approach predicts the UAV's optimal position faster compared to LSTM. The Transformer-based network discerns that achieving a high success rate doesn't necessarily require drastic movements or strict adherence to the user group's centroid position. Instead, it identifies a nuanced correlation between base stations (BS) and users, considering factors such as the transmitted power from the BS. To corroborate this insight, Fig. 7 presents a performance comparison in terms of sum rate versus episodes for Transformers and LSTM in AIRS-RSMA networks. It is observed that the Transformer-based model consistently outperforms LSTM across all time intervals. It is noteworthy that LSTM's performance suffers when the UAV strictly follows circular paths to track user groups, while the Transformer's model strikes a balance between sum rate and positioning between BS and UAV, as well as between UAV and user groups.

Fig. 8 plots the sum rate of the proposed AIRS-RSMA network using both the proposed LSTM and Transformer models. For comparison purposes, the AIRS-NOMA scheme is also included. By comparing Fig. 8a and 8b, it is evident that the proposed Transformer method significantly enhances the system performance by striking a balance between rate

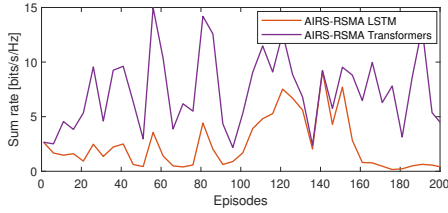
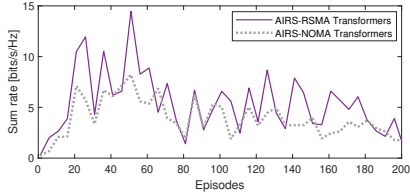
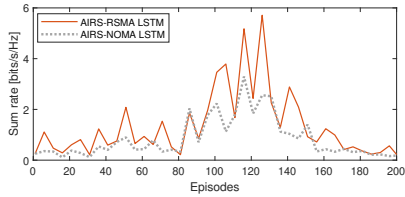


Fig. 7: Sum rate versus episodes for  $\rho = 30\text{dB}$ ,  $\epsilon = 0.01$ .



(a) Performance of Transformers model.



(b) Performance of LSTM model.

Fig. 8: Performance comparison AIRS-RSMA versus AIRS-NOMA  $\rho = 30\text{ dB}$  and  $\epsilon = 0.01$ .

and distance from the BS. Furthermore, AIRS-RSMA exhibits impressive sum rate values compared to AIRS-NOMA, owing to the additional degree of freedom achieved by intelligently utilizing common messages. Although NOMA leverages the advantages of the user channel, in dynamic scenarios where users may be close to each other, NOMA can achieve similar performance to RSMA in terms of sum rate. For example, in Fig. 8a, during episode 126, AIRS-RSMA achieves a performance of 8.67 bits/s/Hz, while AIRS-NOMA reaches 4.44 bits/s/Hz, representing a performance gain of approximately 51%. Similarly, for the LSTM model, as shown in Fig. 8b, AIRS-RSMA achieves 5.71 bits/s/Hz, while AIRS-NOMA reaches 2.55 bits/s/Hz, translating to a performance gain of 44%. Comparing Transformers to the LSTM model for AIRS-RSMA systems, Transformers can achieve gains of up to 43.5%

## V. CONCLUSIONS

In this work, MISO-RSMA networks assisted by AIRS with dynamic multiuser scenarios were investigated. We presented the system model, the fundamentals of LSTM and Transformers, and two models to predict the trajectory of UAVs. Training and testing results were presented to compare the performance of the proposed models. The simulation results demonstrated the robustness of the proposed Transformer model, significantly improving the sum rate compared to the LSTM schemes. For comparison purposes, AIRS-NOMA was

also simulated. The results underscored the robustness of AIRS-RSMA systems, achieving a superior performance gain compared to AIRS-NOMA systems.

## REFERENCES

- [1] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, 2016.
- [2] X. Yuan, Y.-J. A. Zhang, Y. Shi, W. Yan, and H. Liu, "Reconfigurable-intelligent-surface empowered wireless communications: Challenges and opportunities," *IEEE Wirel. Commun.*, vol. 28, no. 2, pp. 136–143, Feb 2021.
- [3] S. Li, B. Duo, M. D. Renzo, M. Tao, and X. Yuan, "Robust secure UAV communications with the aid of reconfigurable intelligent surfaces," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 10, pp. 6402–6417, 2021.
- [4] C. Zhao, X. Pang, W. Lu, Y. Chen, N. Zhao, and A. Nallanathan, "Energy efficiency optimization of IRS-assisted UAV networks based on statistical channels," *IEEE Wirel. Commun. Lett.*, vol. 12, no. 8, pp. 1419–1423, 2023.
- [5] B. K. S. Lima, A. S. De Sena, R. Dinis, D. Benevides Da Costa, M. Beko, R. Oliveira, and M. Debbah, "Aerial intelligent reflecting surfaces in MIMO-NOMA networks: Fundamentals, potential achievements, and challenges," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 1007–1024, 2022.
- [6] Y. Mao, B. Clerckx, and V. Li, "Rate-splitting multiple access for downlink communication systems: Bridging, generalizing and outperforming SDMA and NOMA," *EURASIP J. Wirel. Commun. Net.*, vol. 2018, 05 2018.
- [7] X. Liu, J. Feng, F. Li, and V. C. M. Leung, "Downlink energy efficiency maximization for RSMA-UAV assisted communications," *IEEE Wirel. Commun. Lett.*, vol. 13, no. 1, pp. 98–102, 2024.
- [8] M. Katwe, K. Singh, B. Clerckx, and C.-P. Li, "Rate splitting multiple access for sum-rate maximization in IRS aided uplink communications," *IEEE Trans. Wirel. Commun.*, vol. 22, no. 4, pp. 2246–2261, 2023.
- [9] S. K. Singh, K. Agrawal, K. Singh, B. Clerckx, and C.-P. Li, "RSMA for hybrid RIS-UAV-aided full-duplex communications with finite block-length codes under imperfect SIC," *IEEE Trans. Wirel. Commun.*, pp. 1–1, 2023.
- [10] F. Guo, L. Lu, Z. Zang, and M. Shikh-Bahaei, "Machine learning for predictive deployment of UAVs with multiple access," *IEEE Open J. Commun. Soc.*, vol. 4, pp. 908–921, 2023.
- [11] B. Zhu, E. Bedeer, H. H. Nguyen, R. Barton, and Z. Gao, "UAV trajectory planning for AoI-minimal data collection in UAV-aided IoT networks by transformer," *IEEE Trans. Wirel. Commun.*, vol. 22, no. 2, pp. 1343–1358, 2023.
- [12] X. Gao, X. Mu, W. Yi, and Y. Liu, "Intelligent trajectory design for RIS-NOMA aided multi-robot communications," *IEEE Trans. Wirel. Commun.*, pp. 1–1, 2023.
- [13] C. You, B. Zheng, and R. Zhang, "Channel estimation and passive beamforming for intelligent reflecting surface: Discrete phase shift and progressive refinement," *IEEE J. Sel. A. Commun.*, vol. 38, no. 11, pp. 2604–2620, 2020.
- [14] J. P. Matos-Carvalho, S. D. Correia, and S. Tomic, "Sensitivity analysis of LSTM networks for fall detection wearable sensors," in *2023 6th Conference on Cloud and Internet of Things (CIoT)*, 2023, pp. 112–118.
- [15] R. Santos, J. P. Matos-Carvalho, S. Tomic, M. Beko, and S. D. Correia, "A hybrid lstm-based neural network for satellite-less UAV navigation," in *2023 6th Conference on Cloud and Internet of Things (CIoT)*, 2023, pp. 91–97.
- [16] B. K. S. Lima, J. P. Matos-Carvalho, R. Dinis, D. B. Da Costa, M. Beko, and R. Oliveira, "LSTM-based trajectory and phase-shift prediction for RSMA networks assisted by AIRS," *IEEE Transactions on Communications*, pp. 1–1, 2024.
- [17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf)
- [18] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.