2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC),
18-22 September 2022, Portsmouth, Virginia, USA
DOI:10.1109/DASC55683.2022.9925774

Traffic Flow Prediction for UTM Application: A Deep Learning Approach

Abdulrahman Alharbi, Ivan Petrunin and Dimitrios Panagiotakopoulos

School of Aerospace, Transport and Manufacturing, Cranfield University
Cranfield ,UK
Email: {Abdulrahman.A.Alharbi, i.petrunin, D.panagiotakopoulos}@cranfield.ac.uk

***Abstract* - Over the past few years, the research community has focused greatly on predicting air traffic flows, yielding remarkable outcomes. We found that existing literature in the field mainly covers prediction of air traffic flows for conventional aircraft. However, there is limited research about prediction of air traffic flows for Uncrewed Aircraft Traffic Management (UTM). This research study proposes a deep learning-based approach to predict air traffic congestion in the context of UTM over a period of three minutes. The use of the model aims to address congestion considering air traffic uncertainties instead of addressing the conventional issues of trajectory prediction or conflict detection and resolution. Our model also considers the influence of recreational users who fly UAVs at random times, during the execution of the above essential missions. Further, the effects of airspace structure configurations like static No-Fly Zones (NFZ), airfields with variable availability for drone flights, recreational areas, emergency UTM operation and environmental factors such as weather conditions have also been studied. The proposed model shows better performance compared to other approaches such as the Shallow neural networks and regression models.**

Keywords - complexity metrics, long short-term memory (LSTM) networks, uncrewed aerial vehicle (UAVs), uncrewed aircraft traffic management (UTM).

I. INTRODUCTION

The strength and numerous uses of uncrewed aerial vehicles (UAVs) have the potential to be of significant advantage to both commercial and industrial sectors. Worldwide, sales of UAVs are forecast to increase by an annual rate of 15.88% over the next four years [1]. In order to fulfil the expected rise in required services encompassing delivery, surveillance, and aerial photography [2], restricted airspace resources are coming under intense pressure to expand, leading to a lack of demand-capacity equilibrium and associated flight backlogs that have the capacity to create possible safety problems and inefficacy within air traffic control spheres. In order to remedy these issues, a precise and timely forecast of traffic flow provides a major contribution to pertinent managerial judgements, administrative development strategies and functional proficiency enhancement. Additionally, traffic flow data facilitate more effectual and resourceful decisions by air traffic controllers.

The definition of air traffic density, also referred to as congestion, is the frequency of airplanes travelling through a specific region over a certain period of time [3]. Congestion gives rise to the possibility of air traffic incidents and thus necessitates monitoring. Air traffic complexity reflects the degree of challenge that an actual traffic scenario poses to air traffic control [4]; this parameter has a negative association with the capacity of a controller to make choices and exacerbates the rate of judgment aberrations [5]. These so-called "hot spots" of air traffic necessitate close scrutiny from controllers so as to determine whether any trajectories need modification in order to avoid any conflict.

Complex and increasingly dense UAV traffic adds a notable duty to the administration of air traffic, metropolitan plans and the assignment of resources. Given this context, a number of key questions are typically voiced, such as 'do we know if an appropriate route in terms of airspace safety and energy efficiency for a specific mission at a specific time can be found ahead of time? Is it necessary to postpone the launch of some UAVs in order to accommodate a higher priority mission scheduled at a specific time'? Responding to these queries and having the ability to predict the traffic characteristics in advance would facilitate more effectual preparation and regulation.

Due to the increasing demand for UAVs, studies have been conducted to investigate the many noteworthy issues related to UAS traffic management. However, research about prediction of air traffic flow for Uncrewed Aircraft Traffic Management (UTM) systems is limited. Most researchers focused on predicting future traffic density based upon historical data [6]. They assumed a static environment with fixed start and destination points of vehicles, as well as fixed airspace constraints regarding no-fly zones (NFZ). A static environment is not directly applicable to UTM systems due to the dynamicity of the UAV's operational environment. Hence, there is a need to design practical scenarios that consider uncertainties due to weather conditions and static and dynamic obstacles, especially

In order to model and predict UTM airspace traffic flow while considering air traffic dynamics and uncertainties, this paper proposes a learning-based model to predict air traffic congestion over a period of three minutes based upon existing machine learning techniques. The use of the model aims to address issues related to UAV traffic management. It focuses on predicting congested regions by considering air traffic uncertainties instead of addressing the conventional issues of trajectory prediction or conflict detection and resolution. It applies a deep learning approach, a long short-term memory network (LSTM), to analyze air traffic patterns based upon time series. These patterns depend upon defined parameters and predict congested regions in UTM systems. The overall architecture of the proposed model is presented in Fig. 1.

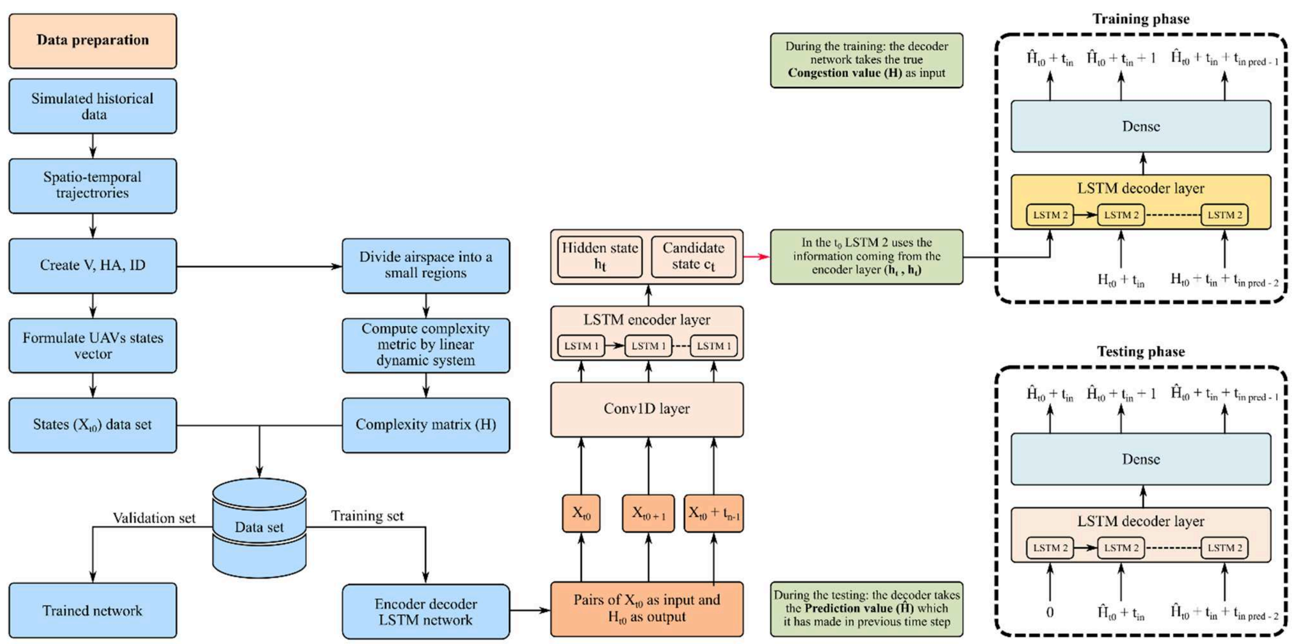

Fig. 1. UTM traffic flow prediction model.

The remainder of this paper is as follows: Section II details associated research and offers the background for the current proposed techniques; Section III describes the research methods; Section IV presents the proposed model results and discusses the performance of the model comparing with existing prediction approaches; Section V indicates the conclusions of the work and recommendations for further studies.

II. Related Works

Within the domain of air traffic research, air traffic flow prediction (ATFP) forms a vital process, with the objective of gauging traffic characteristics for a particular airspace at a later timepoint according to the real-time traffic context and previous operating information. ATFP has become an area of interest for scientists worldwide; numerous superlative results have been attained within this sphere [7]. A range of methods, essentially classified into four cohorts, have been innovated and dedicated to ATFP: flight plan-based algorithms and time series, probabilistic and machine learning algorithms, respectively. The following sections offer an abridged précis of these methods with some related literature.

A. Flight Plan-based Algorithms

Flight plan-based algorithms are a two-fold ATFP method in which a four-dimensional trajectory prediction is employed in order to gauge the flight over a period of individual waypoints according to the flight plan. This is then utilized to anticipate the air traffic flow of the relevant airspaces [8]. The drawback of this technique is that it relies significantly on the precision of the four-dimensional trajectory projections. Furthermore, this method fails to incorporate information regarding the real-time traffic context. Corrections applied to the prediction through the additional use of real-time assessment of flight positions have been proposed [9]. However, the forecast outcome demonstrates notable divergence from the ground situation as the trajectory predictions for flights are unable to acquire the altering worldwide traffic scenario in a timely manner.

B. Time Series Algorithms

A number of reviews in relation to time series algorithms deployed in the assessment of the utilities of varying domains have been published [10], [11], [12]. In the current work, linear stochastic paradigms for prediction have been widely used, e.g., autoregressive, moving average, autoregressive integrated moving average (ARIMA), seasonal ARIMA, autoregressive fractionally integrated moving average, autoregressive conditional heteroscedasticity (ARCH), and generalized ARCH. The models founded on ARIMA are a more useful and straightforward strategy than alternative models as the only information required is prior data. The principal disadvantage, however, is that the impact of the traffic scenario in neighboring airspaces is overlooked. Similarly, another dynamic air traffic flow paradigm derived from a network failed to take note of the time-reliance of historical traffic contexts [13].

C. Probabilistic and Stochastic Algorithms

Rather than providing the deterministic frequency of aircraft within an area over a period of time, techniques founded on probabilities measure the ambiguities in establishing ATFP and offer numerical output as well as associated probability. Such models were researched in order to establish the stochastic properties of air traffic flow. This method is utilized in order to predict traffic flow in more detail [14]; ambiguities are considered in order to emphasize probabilistic demand prediction. This technique was noted in a further study [15], which took into account the divergences induced by random phenomena, e.g., an alteration of a departure or arrival time, a non-permanent deviation from a trajectory or flying height owing to adverse weather or an unscheduled cancellation. These data infer that probabilistic forecasting of traffic flow according to the properties of error distribution provides information which is comparable to the true data. Despite the more precise forecasting data, the output of this technique is influenced by handcrafted models and flaws within the paradigm; therefore, it does not illustrate the entire real traffic scenario, which diminishes the accuracy of the prediction.

D. Machine Learning Algorithms for ATFP

With successful applications of machine learning and, specifically, deep learning-based models for prediction purposes in other fields such as road traffic prediction, it is also studied to solve the existing problems of air traffic flow management (ATM).

It was observed that the majority of publications are related to road or highway traffic flow or ATFP for aircrafts. UAV navigation, obstacle avoidance, and/or control approaches are some of the issues investigated by the current works [6], [16].

The work in [7] suggested an aggregate machine learning model that takes into account the entire air traffic flow situation by considering both spatial and temporal dependencies due to adjacent areas, historical traffic situation and flight levels. A novel combined machine learning model was trained with the integration of convolutional neural networks and recurrent neural network (ConvLSTM) for the traffic flow prediction using traffic flow matrix (TFM) data. The results of this study exhibit better performance compared to existing approaches due to the fact that the model predicts the flow distribution at different flight levels, thus helping to improve the operational efficiency of ATM.

In order to enhance the precision of air traffic predictions, a predictive autoregression model was developed utilizing support vector machine (SVM) merged with a polynomial model and a strong autoregression model so as to generate a coalesced predictive paradigm [17]. The latter was trialed with true air traffic information obtained from the Beijing ATC area. Greater ATFP precision of nearly 3% was achieved with the combination model when judged against SVM only. An innovative ATFP technique, founded on deep learning methods, took both spatial and temporal considerations into account [18] by utilizing a stacked auto-encoder paradigm trained in a greedy layer-wise manner to learn empirical properties of traffic flow.

The work in [19] developed a deep learning model to predict road traffic flows. Its main contribution is the development of an architecture that combines a linear model with a sequence of tanh layers. This work shows that deep learning architectures can capture nonlinear spatio-temporal effects. The first layer was used to identify spatio-temporal relations among predictors and the other layers were used to model nonlinear relations. It was shown that deep learning provides precise short-term traffic flow predictions.

A further study [20] used a complexity metric for ATFP, which was autonomous of any system of traffic control and deployed an innovative encoder-decoder LSTM neural network. A mean absolute error of 0.08 was obtained in predicting the air traffic complexity value 40 minutes in advance.

The work presented in [6] used the image-based trajectory data as input to convolutional neural network (CNN) and LSTM cascaded deep neural network and predicted the UAV instantaneous density using the segmentation method that relies on historical data. Although this work used correlation as the metrics for the evaluation of the proposed network and established a continuous prediction time horizon of one hour with good correlation scores, it did not take into account the realistic or practical missions as considered in this current work, nor did it take into account the effect of dynamical airspace structural constraints such as recreational areas and airfields or the effects of UAV prioritization; there is always a need to set the priority list for different missions The effects of weather constraints such as adverse wind, rain and extreme weather conditions were also not considered in this work.

The review of traditional ATFP techniques implies that contemporary strategies are bottom-up, i.e., predicting the traffic flow for a particular airspace rather than the general traffic dispersal within the region of interest. Additionally, a number of techniques focus on empirical airplane numbers within a region, therefore lacking the capacity to differentiate between scenarios of greater and lesser complexity for equivalent air traffic populations.

However, one of the challenges in forecasting and predicting UAVs' air traffic flow is the unavailability of historical data [21], a state of affairs which compelled the development of simulation frameworks for data generation that could take all the real-time scenarios and anomalies into account. It can be debated that there is a strong need to design practical scenarios that must take uncertainties into account, such as adverse weather conditions and static and dynamic obstacles due to the usage of UAVs in urban environments where the safety of assets may be of prime interest. Also, the emergency operations in UTM domains pose additional constraints on the development of simulation scenarios that could provide more accurate replicas against real historical data.

III. Methodology

A. Historical Data generation and Pre-processing

There is special need to develop scenario-driven planning approaches that aim to optimize plans for UAVs and enable the selection of who to serve, which routes to take and how much to deliver [22]. Due to limited resources and other restrictions, planning a relief operation is difficult, particularly for last mile delivery activities (from distribution centers to beneficiaries) [23]. Moreover, the unavailability of historical UTM traffic data prompts the generation of data for the prediction of UAV traffic flow patterns by using a simulation framework. The simulation uses particle swarm optimization (PSO)-based optimization algorithm to provide optimal paths from a UAV service start point to its delivery point.

The proposed model has been validated using a drone delivery system for the essential delivery of Covid-19 test samples, package delivery services and emergency fire-surveillance tasks, with different priority levels assigned to them. The model also considers the influence of recreational users who fly UAVs at random times during the execution of the above essential missions.

Further, the effects of airspace structure configurations like static NFZ, airfields with variable availability for drone flights, recreational areas and environmental factors such as weather conditions have also been studied.

The suggested method was evaluated and verified by running a simulation over the airspace of Bedfordshire, UK. Places that may be restricted for flights are airfields, recreational areas, and prisons. The test area included four airfields: Luton, Cranfield, Halton, and Old Warren (orange); four recreational areas: Dunstable, Sandy, Cardington, and Graveley (yellow); and Milton Keynes Prison (Blue). Fig. 2 depicts the simulation environment and the missions.

To make the simulation more realistic, the following simulation parameters were taken into consideration:

1) Fixed start and end position: As part of the simulation, the UAV missions' start, and end positions were set to mimic typical everyday operations and emergency services.
2) Priority levels: Each flight was assigned a service priority level, ranging from Level 1 (highest priority) to Level 4 (lowest priority). The following is a description of each level of priority:

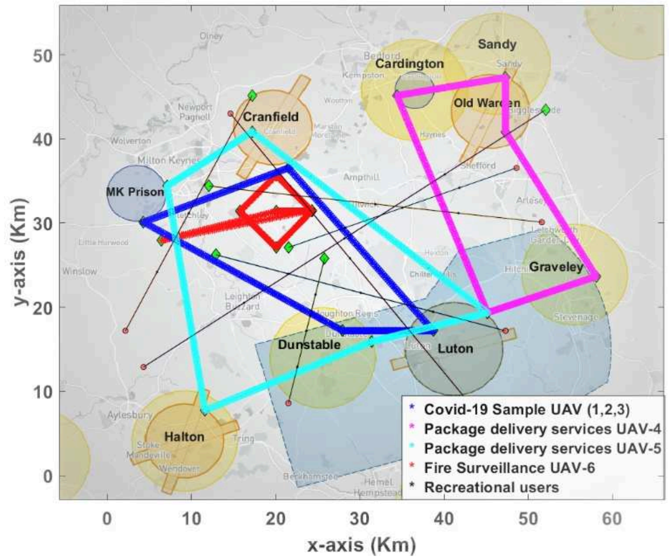

Fig. 2. Environment of simulation scenarios and the UAV missions map.

   a. Fire surveillance services (emergency services).
   b. Covid-19 test samples delivery service to multiple clinics.
   c. Package delivery service to multiple post offices.
   d. Recreational users who fly UAVs at random times (single-leg missions).

3) Dynamic NFZs: At some hours, airfields and recreation areas considered for simulation are dynamic in nature. This dynamism is random, making some areas available and others unavailable for the whole duration of the hour(s).
4) Random departure time: To make the system more realistic, the exact time at which each hobbyist's UAV departs is randomly set between one and 10 minutes in a 1-hour simulation scenario period.
5) Weather ambiguity: Different weather conditions classified as adverse and severe were considered. The details of the weather effects' implementations were presented in [24].
6) Deconfliction strategy: When two or more UAVs are in the same location at the same time, it is called a conflict. Literature [25] discusses some de-confliction strategies to resolve any UAV conflict that appears in the scenarios. In this simulation setting, de-confliction strategies such as ground delay and slow speed were used.

To simulate more complex dynamic airspace, 100 UAV trajectories were considered, 73 of which belonged to special UAV missions and 27 to random flying by hobbyists. A multi-mission scenario was also simulated between 9:00 am to 12:00 pm for the Bedfordshire area, where four missions were taking place. Three different sub-scenarios were created based on these hours as follows:

1) **First simulation scenario:** This simulation ran between 9:00 am and 10:00 am. In this scenario, all nine NFZs were static with no dynamic obstacles and

no weather constraints. As a result, no UAV could fly above them during this hour.

2) **Second simulation scenario:** This simulation ran between 10:00 am and 11:00 am. The difference from the first scenario is that any NFZ corresponding with recreational areas were dynamic, making some areas available and others unavailable at this hour. This scenario also incorporated severe weather effects.
3) **Third simulation scenario:** This simulation ran between 11:00 am and 12:00 pm. In this scenario, airfields are dynamic, while all four recreational areas and the prison were kept static. Among the four airfields, Luton and Cranfield airfields were available, and therefore, UAVs of flying hobbyists could fly over Luton and Cranfield areas at some points. The effects of adverse rain and wind were considered in this scenario.

For simplicity's sake, the figure of one of these scenarios is presented Fig. 3.

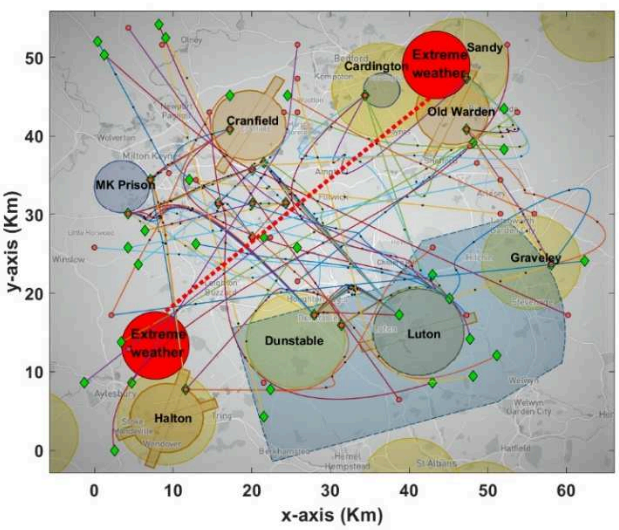

Fig. 3. Scenario 2:100 UAVs with extreme weather effects.

Once the PSO simulation had been completed, the UAV's states were formulated by arranging the five parameters: longitude (x), latitude (y), timestamps (z), velocity (V) and head direction (HA) for each UAV. Then, the whole airspace was divided into small cubes to identify the neighboring UAVs. The complexity matrix, which contains complexity information for each cube in the airspace, was evaluated by using negative eigenvalues of UAVs, as discussed in the next section.

B. Complexity Metric Formulation

In this work, we adapted the inherent complexity metric for a linear dynamic system model published previously [26] to address the prediction task of the ATFP in UTM domain. It calculates a complexity parameter in the neighborhood of an aircraft at a certain time (see Fig. 4). The proposed complexity metric was formulated to capture the dynamic behavior of neighboring UAVs in a reference window of airspace A filter is used in order to take into account the flights with a likelihood of engagement with the reference flight. For example, a UAV at a distance of 50 m [27] from the reference aircraft will not interact with the reference aircraft, and so, would not be calculated in the metric computation.

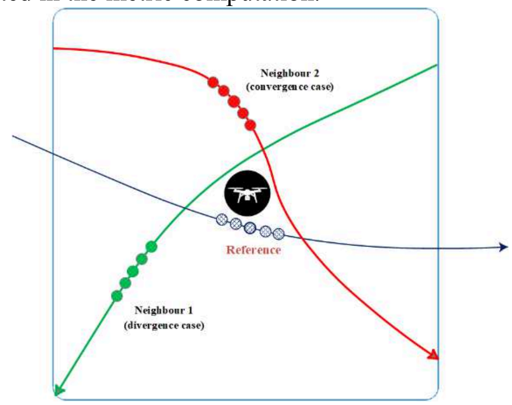

Fig. 4.The neighborhood of a reference UAV to calculate the complexity metric [26].

The dynamic behavior of a system can be represented with the following linear equation:

$$\dot{x} = Ax + b \tag{1}$$

The complexity metric can be formulated by including the motion of neighboring UAVs. Motion of the UAVs through the airspace can be presented using the following linear equation:

$$\dot{U} = A_U U + b \tag{2}$$

In the above equation, $\boldsymbol{A_U}$ is the state transition matrix representing the dynamics of UAVs in the neighborhood of the reference UAV, whereas $\boldsymbol{b}$ is the input matrix based on the speed of the respective neighboring UAV. If the matrices $\boldsymbol{P}$ and $\boldsymbol{V}$ were such that:

$$P = \begin{bmatrix} x_1 & \dots & x_n \\ y_1 & \ddots & y_n \\ z_1 & \dots & z_n \end{bmatrix} \quad and \quad V = \begin{bmatrix} V_{x_1} & \dots & V_{x_n} \\ V_{y_1} & \ddots & V_{y_n} \\ V_{z_1} & \dots & V_{z_n} \end{bmatrix} \tag{3}$$

where x_1, y_1 *and* z_1 represent the spatial-temporal coordinates, and V_{x_1}, V_{y_1} *and* V_{z_1} represent the velocities for the first UAV in the neighborhood of the reference UAV. The state transition matrix $\boldsymbol{A_U}$ can be evaluated as follows:

$$min_{A,b} \| V_U - (A_U P + b) \|^2 \tag{4}$$

When the characteristic information of the state transition matrix was scrutinized, it was evaluated that the positive real parts of the eigenvalues correspond to divergence from the reference neighborhood along the associated eigendirections while negative real parts correspond to convergence on a reference point. Therefore, the complexity in a neighborhood can be mathematically presented as:

$$c(A_u) = \sum_{Re(\lambda(A_u)) < 0} |Re(\lambda(A_u))| \tag{5}$$

Following diagonalization of the matrix $\boldsymbol{c(A_u)}$, it was observed that the system's asymptotic activity relies only on A eigenvalues. Positive and negative true eigenvalue components equate to divergence along the related eigenvectors and convergence toward a key coordinate, respectively (Fig. 5).

Thus, as the dynamic system is closed, the metric $\boldsymbol{c}(A_u)$ is a parameter of the magnitude of the system's diverging or converging activity.

Given that this system has the tightest approximation of a linear dynamic system with a good fit for the present air traffic loci and speeds, a robust converging activity equates to quickly joining trajectories, which are linked with a significant level of complexity.

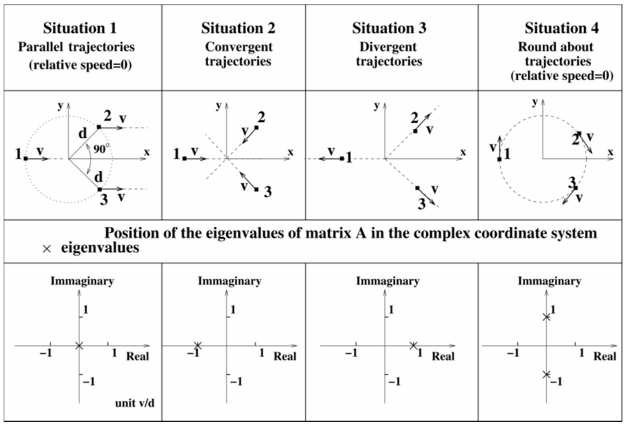

Fig. 5. Eigenvalues of the dynamical system [26].

C. *Architecture of the Prediction model*

1) One-dimensional (1-D) Convolutional Layer:
Described as a single-dimension convolutional layer, the initial encoder network layers process the input sequence that reflects the UAV states. Each such layer is made up of different f filters, which encompass a kernel of learnable values parameters with the dimension $d * m$, where m and d indicate the input sequence size and a convolutional layer hyperparameter, and the breadth of the kernel, respectively. The individual filters amass the result of the input sequence relating to the distance and the individual output series elements all include data from a number of serial time increments. This can be represented more accurately as:

$$y_{ij} = \emptyset\left(\sum_{k=-d}^{d}\sum_{l} w_{kl}^{j} x_i + k, l\right) \qquad (6)$$

where y_{ij} is the j^{th} vector element of the i^{th} sequence term, x_{kl} is the l^{th} vector element of the k^{th} sequence term of the input, w_{jkl} is the k^{th} weight of the kernel of the j^{th} filter associated with the l^{th} dimension of the input sequence, and Φ is an activation function.

Convolution operations are carried out only along the input sequence's time dimension; thus, it is referred to as being one dimensional (1-D). Given that the convolutional strata are of a single dimension, it was anticipated that the encoder network would recognize dependencies within an extremely brief time period. Thus, the LSTM strata could undertake processing of an intermediate sequence whereby the individual time increments incorporated dynamic data from the earlier and upcoming time intervals. This would enable calculation of the embedding for the entire sequence. In the absence of the convolutional strata, the LSTM layers could only process each time interval consecutively without any data from subsequent time increments.

2) Encoder-Decoder LSTM:
A category of networks, recurrent neural networks (RNN), were engineered in order to manage serial data, e.g., text streams, audio extracts and time-series information [28]. Nevertheless, the empirical RNN stratum experiences challenges in learning long-term dependencies when the linear dimensions of the sequence are sizeable. This issue has been termed the vanishing gradient problem [29]. Its resolution has been attempted with a number of RNN layer configurations with long-term memory, e.g., LSTM [30]. The latter is an enhanced block of RNN which includes four regulatory gates, i.e., input, forget, cell and output, respectively, together with concealed units. The overall configuration of a lone LSTM block is depicted in Fig. 6. The mathematical description is as follows:

$$\begin{aligned} I^t &= f(W_{ix}x^t + W_{ih}h^{t-1} + W_{ic}C^{t-1} + b_i) \\ F^t &= f\big(W_{fx}x^t + W_{fh}h^{t-1} + W_{fc}C^{t-1} + b_{if}\big) \\ C^t &= F^t \circ C^{t-1} + I^t \circ g(W_{cx}x^t + W_{ch}h^{t-1} + b_c) \\ O^t &= f(W_{ox}x^t + W_{oh}h^{t-1} + W_{oc}C^t + b_o) \\ h^t &= O^t \circ g(C^t) \end{aligned} \qquad (7)$$

where the respective activations of the input, forget, cell and output gates are denoted by I^t , F^t , C^t and O^t , and h^t indicates the hidden unit. The *t* superscript indicates the forecasting moment. *W* represents the weight tensor; the vector of the information transmission is given by the subscripts. The bias within the equivalent part is shown by *b.*

Sequence-to-sequence prediction issues are difficult to resolve due to the fact that input and output sequences comprise differing item populations. The encoder-decoder LSTM, is engineered in order to deal with such situations. Its configuration is formed by a pair of models. The first encodes the input sequence into a vector of fixed length, and the second decodes the latter and provides an output of the forecast sequence [20].

The novelty of this configuration is the utilization of a fixed dimensional intrinsic representation within the model's center, which is termed sequence embedding [31]. The encoder and decoder both fall under the description of RNNs, i.e., LSTMs.

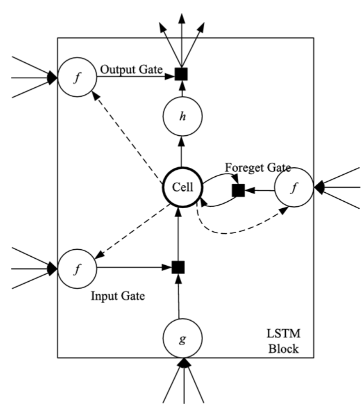

Fig. 6.The general architecture of LSTM [7].

3) Model training:

A sequence-to sequence regression assignment forms the goal of the current supervised learning model. In order to accomplish this, a training set has to be constructed utilizing training input/output pairings made up of sequences of UAV states matched with complexity value sequences for the whole airspace. In order to engineer the training outputs, an $\boldsymbol{n} \times \boldsymbol{n}$ complexity metric matrix was described $\boldsymbol{H_t}$ for each time increment t. For a specific time t, the input vector $\boldsymbol{X_t}$ was obtained by the concatenation of all UAV states existing within the airspace at that time. Such states from the database were entered into the 1-D convolution layer versus the timestep. The consequent output was then introduced into a number of LSTM layers.

The encoding sequence is formed by the hidden states of the final LSTM layer. This is the only encoder data that will be conveyed to the decoder network, which is itself made up of a number of LSTM layers, of which the hidden states are stimulated by the encoding sequence, together with numerous dense layers. The output vector from the latter has an $\boldsymbol{n^2}$ dimension, which is equivalent to the complexity metric's matrix.

Two elements form the decoder input, namely, the encoder's output and the output sequence term which was predicted or decoded previously. In training, the coder network is instigated by a method referred to as teacher forcing, which indicates that in training, the true output sequence is used as a further input into the decoder network, but translocated by a single time increment, i.e., the additional input into the decoder network is the output it should have forecast at the preceding time juncture. At the inference mechanism, the prediction determined at the earlier time interval was thus taken into account.

The root mean squared error (RMSE) represents the loss function and can be expressed as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(C_{ac} - C_{pr}\right)^2} \tag{8}$$

Where N represents the total number of airspace (x, y) coordinates. C_{ac} represents the actual amount of congestion at any specific point and C_{pr} presents the predicted congestion at that specific (x, y) coordinate.

The list of model hyperparameters are shown in TABLE *1*.

TABLE 1. LIST OF HYPERPARAMETER VALUES

Parameter	Value
Batch size	128
(1-D) kernel width d	3
(1-D) filter f	512
Hidden layers (LSTM)	128
Activation	ReLU
Optimizer	Adam optimizer [32]
Learning rate	0.001
Epochs	500
Loss function	RMSE -Equation (8)

IV. RESULTS AND DISCUSSION

This section presents the results of the performance of the proposed architecture in the scenarios explained in section III above. These three scenarios perturb the ideal mission plans with the introduction of various dynamic random factors such as recreational areas, airfields, and various uncertain weather conditions. The proposed model is trained on 90% of the dataset and then tested on the rest of 10% of this data.

A. Prediction result for First Scenario of Simulations:

A scenario of 100 UAVs with a no-fly zone with no weather conditions has been considered in this case to evaluate the air space congestion. A heatmap of the congestion matrices for this scenario is presented in Fig. 7. The congestion in the Bedfordshire airspace due to neighboring UAVs has been presented in the heat map. The lowest and highest congestion has been presented with blue and red colors respectively.

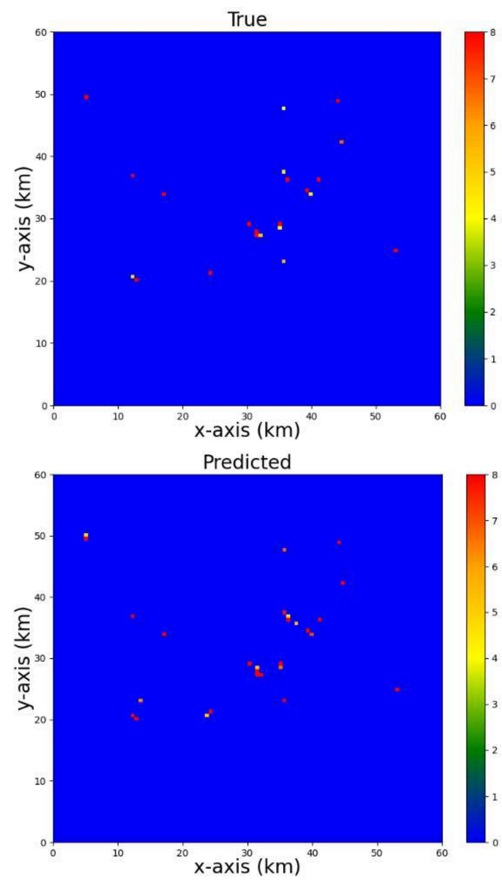

Fig. 7. Actual and predicted complexity in the airspace for the first scenario.

It can be seen from the figure above that the proposed model can capture most of the highly complex regions appearing in the future timestamp. The performance of the LSTM trained architecture can be evaluated by the RMSE value of the prediction. In this scenario, an RMSE reached a value of 0.30. This small value of the RMSE metric indicates the high accuracy of our model.

B. Prediction result for Second Scenario of Simulations:

To add more complexity to the Bedfordshire airspace, the second scenario of 100 UAVs with extreme weather conditions and the recreational areas being dynamic was considered. The congestion heatmap is presented in Fig. 8 below.

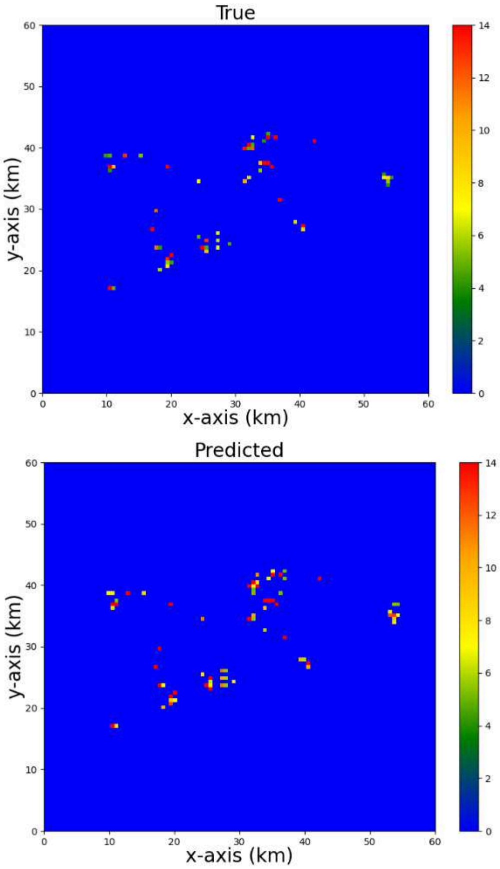

Fig. 8. Actual and predicted complexity in the airspace for the second scenario

It can be seen from the above heatmap that the encoder-decoder model has captured a major complexity in the airspace by observing the dynamic behavior of UAVs, their speed, and areas of congestion. To quantify the performance of this model, the loss value for training and validation at each epoch has reached the value of 0.39.

C. Prediction result for Third Scenario of Simulations:

Similarly, to understand airspace density behavior under different scenario parameters, the third scenario with 100 UAVs in airspace was considered with adverse rain and wind as weather uncertainties. In addition, the airfields' areas were dynamic, while all the four recreational areas and the prison were kept static. A heatmap of the congestion for this scenario is in Fig. 9.

The below figure shows that the proposed model learns the spatial and temporal transition patterns of flight flow in air traffic. The key characteristics of air traffic flow are captured in order to forecast future air traffic flow. In this scenario, a RMSE reached the value of 0.34. It can be seen from the RMSE that the prediction of the trained model architecture is sufficiently close calculated true congestion value based on dynamical model discussed earlier in section III-B, which reflects the optimal performance of the proposed air traffic flow prediction architecture.

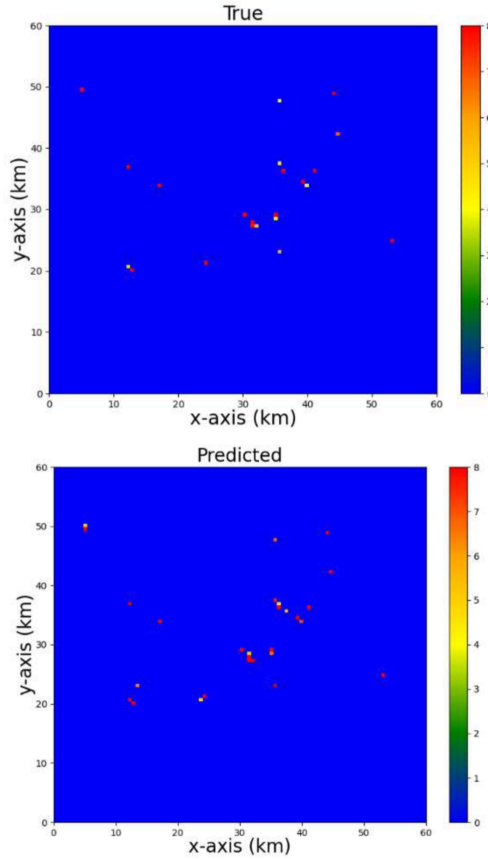

Fig. 9. Actual and predicted complexity in the airspace for the third scenario.

D. Comparison between the Proposed model with the Existing approaches

To evaluate the performance of the proposed encoder-decoder LSTM architecture for the prediction of air traffic flow patterns, a comprehensive comparative analysis is presented in this section. The proposed model has been compared with two prediction approaches, which were discussed in the literature. All three scenarios were used to evaluate the performance of each approach. The selected architectures for the comparative analysis with the proposed model are : a shallow NN-based model with dense network connections that was proposed in [33] and a regression architecture-based Nonlinear Auto Regression with External input (NARX) model, which has been presented in [12]. The performance of all the selected approaches has been evaluated on some of the key performance metrics as follows:

$$Mean\ (\mu) = \frac{1}{N}\sum_{k=1}^{N}\left|C_{ac} - C_{pr}\right| \quad (9)$$

$$Standard\ deviation\ (\delta) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\left(C_{ac} - C_{pr}\right) - \mu\right)^2} \quad (10)$$

$$AMPE\ (\eta) = \frac{1}{N}\sum_{k=1}^{N}\left|\frac{\left(C_{ac} - C_{pr}\right)}{C_{ac}}\right| \times 100\% \quad (11)$$

Where N denotes the total number of airspace trajectories (x, y). C_{ac} represents the true congestion values evaluated using dynamic linear model of UAVs presented earlier in section III-B and C_{pr} represents the predicted congestion at that specific airspace (x, y) coordinate using LSTM encoder-decoder architecture. The standard deviation has been used to evaluate how much the predicted point is deviating from the mean of the data. Each predicted point variation has been evaluated and summed up to present the total deviation in the predicted values from the mean value.
Absolute mean percentage error (AMPE) measures prediction accuracy of the model as a percentage. Equation (8) presents the RMSE which used to evaluate the prediction model performances for all three prediction approaches.

The results of the comparison of different prediction approaches are presented in Fig. 10 , Fig. 11 and Fig. 12.

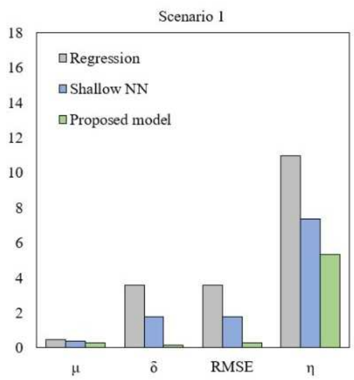

Fig. 10. Comparison of different prediction approaches for the first scenario.

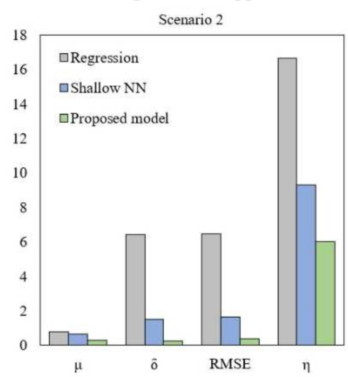

Fig. 11. Comparison of different prediction approaches for the second scenario.

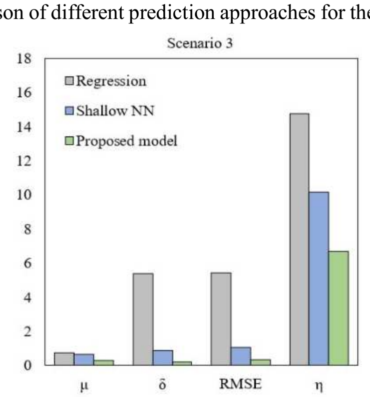

Fig. 12.Comparison of different prediction approaches for the third scenario.

According to the comparative results, we can see that the deep learning model shows the superior performance over the exiting approach.
Regression and Shallow networks are easy to construct as the toolbox is available, but they do not capture the complex relation of states and complexity metrics.

The encoder-decoder LSTM predictive model is a better tool not just for better prediction but also due to its input and output structures, which help provide predictions of airspace complexity with a consecutive step in time. Moreover, this helps evaluate the airspace complexity more intuitively by presenting the airspace congestion on a spatial and temporal scale, which helps the UTM operator to look into the future with a perspective of how the complexity will be changing in the airspace concerning different slots of time.

Due to modelling of both spatial and temporal dependencies, the proposed model showed better prediction performance compared to the other approaches with different weather ambiguities simulated in the three scenarios above.

The regression model only presents the influence of historical data (temporal with the absence of spatial correlations). This means that it is highly dependent on the size of the data and look-ahead horizon. Furthermore, the regression model is susceptible to any unexpected changes in air traffic flow such as flow patterns or weather conditions.

The Shallow NN used with a dense network connection is better for capturing system behavior due to its advanced architecture. The spatial coordinates of airspace complexity were better predicted than by the regression model. However, the proposed model has performance superiority over the Shallow NN.

V. Conclusion & Future work

To meet the critical need for accurate congestion prediction in ATFM systems, we have proposed a learning-based model to predict air traffic congestion over a period of three minutes based on the existing machine learning techniques. It applies a deep learning approach, a long-short term memory network (LSTM). In this model, we designed practical scenarios which consider uncertainties due to adverse weather conditions, and static and dynamic obstacles, especially in urban environments, where the safety of people and assets is of prime interest. Also, emergency UTM operations pose additional constraints in developing simulation scenarios.

The encoder-decoder model has the ability to extract effective features from simulated historical congestion, for a broad range of UAV missions' trajectories. Moreover, it can predict the UTM airspace complexity three minutes in advance and with acceptable accuracy.

Thus, the proposed model enables the UTM operator to regulate and reconfigure UAV paths based upon complexity prediction graphs representing air traffic hotspots. It can also be used to mitigate congestion in predicted UAV traffic hotspots and suggest appropriate conflict-free trajectories by changing head directions, ground delay, or speed. Our model mainly seeks to reduce the workload of the air traffic controller, by predicting congested areas in advance and enabling appropriate action to prevent their formation.

In future work, we will propose a recommendation system that assists the UTM controller to predict the safe actions to mitigate the congestion hotspots, which may lead to conflict, thus, threatening the airspace safety. Keeping in view of the safety-critical aspects and certification, our proposed advisory system will seek to meet the requirements of the trust for such an automation system such as explainability and accuracy.

REFERENCES

[1] M. Intelligence, "DRONES MARKET - GROWTH, TRENDS, COVID-19 IMPACT, AND FORECASTS (2021 - 2026).pdf." 2020.

[2] H. Hildmann and E. Kovacs, "Review: Using Unmanned Aerial Vehicles (UAVs) as Mobile Sensing Platforms (MSPs) for Disaster Response, Civil Security and Public Safety," *Drones*, vol. 3, no. 3, p. 59, 2019, doi: 10.3390/drones3030059.

[3] B. Hilburn, "Cognitive complexity in air traffic control : a literature review COGNITIVE COMPLEXITY IN AIR TRAFFIC CONTROL : A LITERATURE REVIEW Center for Human Performance Research," no. November, 2017.

[4] M. Prandini, L. Piroddi, S. Puechmorel, and S. L. Brázdilová, "Toward air traffic complexity assessment in new generation air traffic management systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 809–818, 2011, doi: 10.1109/TITS.2011.2113175.

[5] E. M. Pfleiderer, C. A. Manning, and S. Goldman, "Relationship of complexity factor ratings with operational errors," no. May, 2007, [Online]. Available: http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA467731.

[6] Z. Zhao *et al.*, "Machine learning-based traffic management model for UAS instantaneous density prediction in an urban area," *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, vol. 2020-Octob, 2020, doi: 10.1109/DASC50938.2020.9256471.

[7] Y. Lin, J. wei Zhang, and H. Liu, "Deep learning based short-term air traffic flow prediction considering temporal–spatial correlation," *Aerosp. Sci. Technol.*, vol. 93, p. 105113, 2019, doi: 10.1016/j.ast.2019.04.021.

[8] Y. Lin, J. wei Zhang, and H. Liu, "An algorithm for trajectory prediction of flight plan based on relative motion between positions," *Front. Inf. Technol. Electron. Eng.*, vol. 19, no. 7, pp. 905–916, 2018, doi: 10.1631/FITEE.1700224.

[9] W. Tian and M. Hu, "Study of air traffic flow management optimization model and algorithm based on multi-objective programming," *ICCMS 2010 - 2010 Int. Conf. Comput. Model. Simul.*, vol. 2, pp. 210–214, 2010, doi: 10.1109/ICCMS.2010.20.

[10] P. Hendikawati, Subanar, Abdurakhman, and Tarno, "A survey of time series forecasting from stochastic method to soft computing," *J. Phys. Conf. Ser.*, vol. 1613, no. 1, 2020, doi: 10.1088/1742-6596/1613/1/012019.

[11] S. Mehrmolaei and M. R. Keyvanpour, "Time series forecasting using improved ARIMA," *2016 Artif. Intell. Robot. IRANOPEN 2016*, pp. 92–97, 2016, doi: 10.1109/RIOS.2016.7529496.

[12] E. Cadenas, W. Rivera, R. Campos-Amezcua, and C. Heard, "Wind speed prediction using a univariate ARIMA model and a multivariate NARX model," *Energies*, vol. 9, no. 2, pp. 1–15, 2016, doi: 10.3390/en9020109.

[13] D. Chen, M. Hu, H. Zhang, J. Yin, and K. Han, "A network based dynamic air traffic flow model for en route airspace system traffic flow optimization," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 106, pp. 1–19, 2017, doi: 10.1016/j.tre.2017.07.009.

[14] C. Wang and L. Yang, "Probabilistic methods for airspace sector flow and congestion prediction," *Xinan Jiaotong Daxue Xuebao/Journal of Southwest Jiaotong University*, vol. 46, no. 1. pp. 162–166, 2011, doi: 10.3969/j.issn.0258-2724.2011.01.026.

[15] W. Tian, H. Xu, Y. Guo, B. Hu, and Y. Yao, "Probabilistic En Route Sector Traffic Demand Prediction Based upon Statistical Analysis of Error Distribution Characteristics," *J. Adv. Transp.*, vol. 2018, 2018, doi: 10.1155/2018/8184513.

[16] Z. Sándor, "Challenges caused by the unmanned aerial vehicle in the air traffic management," *Period. Polytech. Transp. Eng.*, vol. 47, no. 2, pp. 96–105, 2019, doi: 10.3311/PPtr.11204.

[17] H. H. Zhang, C. P. Jiang, and L. Yang, "Forecasting traffic congestion status in terminal areas based on support vector machine," *Adv. Mech. Eng.*, vol. 8, no. 9, pp. 1–11, 2016, doi: 10.1177/1687814016667384.

[18] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang, "Traffic Flow Prediction with Big Data: A Deep Learning Approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, 2015, doi: 10.1109/TITS.2014.2345663.

[19] N. G. Polson and V. O. Sokolov, "Deep learning for short-term traffic flow prediction," *Transp. Res. Part C Emerg. Technol.*, vol. 79, pp. 1–17, 2017, doi: 10.1016/j.trc.2017.02.024.

[20] L. Shi-garrier *et al.*, "Predicting Air Traffic Congested Areas with Long Short-Term Memory Networks To cite this version : HAL Id : hal-03344406 Predicting Air Traffic Congested Areas with Long Short-Term Memory Networks," 2021.

[21] S. Bijjahalli, R. Sabatini, and A. Gardi, "Advances in intelligent and autonomous navigation systems for small UAS," *Prog. Aerosp. Sci.*, vol. 115, no. June, p. 100617, 2020, doi: 10.1016/j.paerosci.2020.100617.

[22] G. Radzki, P. Golinska-Dawson, G. Bocewicz, and Z. Banaszak, "Modelling Robust Delivery Scenarios for a Fleet of Unmanned Aerial Vehicles in Disaster Relief Missions," *J. Intell. Robot. Syst. Theory Appl.*, vol. 103, no. 4, 2021, doi: 10.1007/s10846-021-01502-2.

[23] T. Larrabee, H. Chao, M. Rhudy, Y. Gu, and M. R. Napolitano, "Wind field estimation in UAV formation flight," *Proc. Am. Control Conf.*, pp. 5408–5413, 2014, doi: 10.1109/ACC.2014.6859266.

[24] A. Alharbi, I. Petrunin, and D. Panagiotakopoulos, "Identification and Characterization of Traffic Flow Patterns for UTM application," *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, vol. 2021-Octob, 2021, doi: 10.1109/DASC52595.2021.9594494.

[25] A. Alharbi, A. Poujade, K. Malandrakis, I. Petrunin, D. Panagiotakopoulos, and A. Tsourdos, "Rule-based conflict management for unmanned traffic management scenarios," *AIAA/IEEE Digit. Avion. Syst. Conf. - Proc.*, vol. 2020-Octob, 2020, doi: 10.1109/DASC50938.2020.9256690.

[26] A. García, D. Delahaye, and M. Soler, "Air Traffic Complexity Map based on Linear Dynamical Systems," no. Dd, 2020, [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-02512103.

[27] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, "Analysis of conflict resolution methods for manned and unmanned aviation using fast-time simulations," *SESAR Innov. Days*, no. December, 2019.

[28] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 4, no. January, pp. 3104–3112, 2014.

[29] J. Rosindell and Y. Wong, "On the difficulty of training recurrent neural networks," *Phylogenetic Divers. Appl. Challenges Biodivers. Sci.*, no. 2, pp. 41–71, 2018, doi: 10.1007/978-3-319-93145-6_3.

[30] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[31] L. Lu, X. Zhang, and S. Renais, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 2016-May, pp. 5060–5064, 2016, doi: 10.1109/ICASSP.2016.7472641.

[32] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[33] B. Geng, R., Cui, D. and Xu, "Support vector machine-based combinational model for air traffic forecasts. Journal of Tsinghua University (Science and Technology), 48(7), pp.1205-1208..pdf." .