Edge Intelligence in 5G and Beyond Aeronautical Network with LEO Satellite Backhaul

Babak Mafakheri*, Chao Yan*, Kiran Narayanaswamy*, Isabelle Trang*, Tobias Betz*
Konrad Pientka*, Leonardo Goratti*
* Safran Passenger Innovations GmbH, Munich area, Germany;
emails: {name.lastname}@zii.aero

Abstract—**The vision of ubiquitous network connectivity to fuel uninterrupted services to any user has materialized with the Fifth-Generation (5G) of mobile technology and will probably find maturity on the way to developing 6G. To reach this goal, 5G technology and its evolution (B5G), as well as Multi-access Edge Computing (MEC), alongside Machine Learning (ML) will play pivotal roles. This work sheds light onto a test bed development and initial experimentation results obtained to enable airlines´ passengers on-board an aircraft with broadband connectivity as an advancement toward ubiquitous access. We detail our research and experimentation activity as part of the H2020 AI@EDGE research project around a 5G network and an edge-cloud built on top of aviation-certified hardware and off-the-shelf servers. The edge-cloud is used to develop and test MEC applications that can be seen as the next generation of services offered to airlines and to airlines' passengers and that rely on machine learning. The 5G network is integrated into a larger test-bed and connected to a 5G core on the ground by means of a Low Earth Orbit (LEO) satellite backhaul such as Starlink.**

Index Terms—**5G, MEC, ML, LEO satellite backhaul**

I. Introduction

Air travelers today desire to continue using their personal devices for an uninterrupted connected experience, thus requiring internet connectivity, unlike in the past when the available services were limited to content locally stored on the aircraft. Although access to the on-board Local Data Network (LDN) remains a key component of in-flight connectivity [1], it is equally important to provide broadband internet access in the cabin to ensure that passengers can ubiquitously access services. On the one hand, the traditional In-Flight Entertainment and Connectivity (IFEC) system utilizes the LDN, which can still benefit from new technologies such as Artificial Intelligence (AI) / Machine Learning (ML) and Multi-access Edge Computing (MEC) to improve maintenance and enhance the content selection. On the other hand, the requirement for internet connectivity on Commercial off-the-Shelf (COTS) user devices can be met through the integration of new generations of satcom technologies with cellular networks like 5G, Beyond 5G (B5G) and 6G in the future.

Edge computing, as an extension of cloud computing, is able to host many computationally intensive applications at the network edge and closer to the consumers. This capability makes edge computing one of the pillars of the 5G/B5G system, especially when low latency and bandwidth efficiency are concerned [2]. The European Telecommunications Standards Institute (ETSI) revised the concept of emerging edge computing and introduced it as MEC to broaden its scope [3]. The ETSI MEC architecture introduced the concept of MEC applications (Apps) that can reside inside a MEC host where they consume compute and storage provided by the edge Network Function Virtualization Infrastructure (NFVI). For example, ML algorithms that can figure out how to perform network management tasks by parsing large amounts of data can be deployed within a MEC platform as MEC Apps. Such a setup allows performing required tasks at the edge provided that enough resources and data are available without going all the way to the central cloud [4]. The 5G system, at the same time, has native support for MEC from Third Generation Partnership Project (3GPP) Release 15 with a focus on User Plane Function (UPF) (re)selection, access to both LDN and central data network (e.g., internet). Enhancements of edge computing for different vertical domains (e.g., industrial processes), however, have been investigated by both industry and academia in further releases of 3GPP [5].

The MEC concept is becoming also increasingly important for the future of the Non-Terrestrial Network (NTN), especially due to the expensive nature of the bandwidth available over GEO satellites. With the 5G enhancements in 3GPP Releases 16 and 17, the focus has shifted towards creating a three-dimensional network model by incorporating NTNs. Release 15 of 3GPP primarily dealt with frequency bands (e.g., Ku-bands, S-bands) and antennas, while Release 16 (as per 3GPP TR 38.821 [6]) put an emphasis on architecture, high-level protocols, and use case identification, with further advancements made in Release 17. In 3GPP TR 22.822 [7], two scenarios were identified: i) satellite being used as an access technology in 5G systems by a User Equipment (UE), and ii) satellite serving as the backhaul between a terrestrial gNB and the 5G core [8].

Inspired by the second use case outlined in 3GPP TR 22.822, this paper presents a scenario in which a moving platform, such as an aircraft, can make use of Starlink Low Earth Orbit (LEO) constellation satellites as a backhaul to connect to the central 5G core on the ground. In addition, this paper proposes a new paradigm for aircraft on-board connectivity utilizing the capabilities of the ETSI MEC architecture. To achieve this, we present the deployment of an edge-cloud that harnesses on-board localized information and performs analysis to enhance IFEC system capabilities.

The Aero Edge-Cloud presented in this paper serves as a validation of the H2020 AI@EDGE [9] project approach,

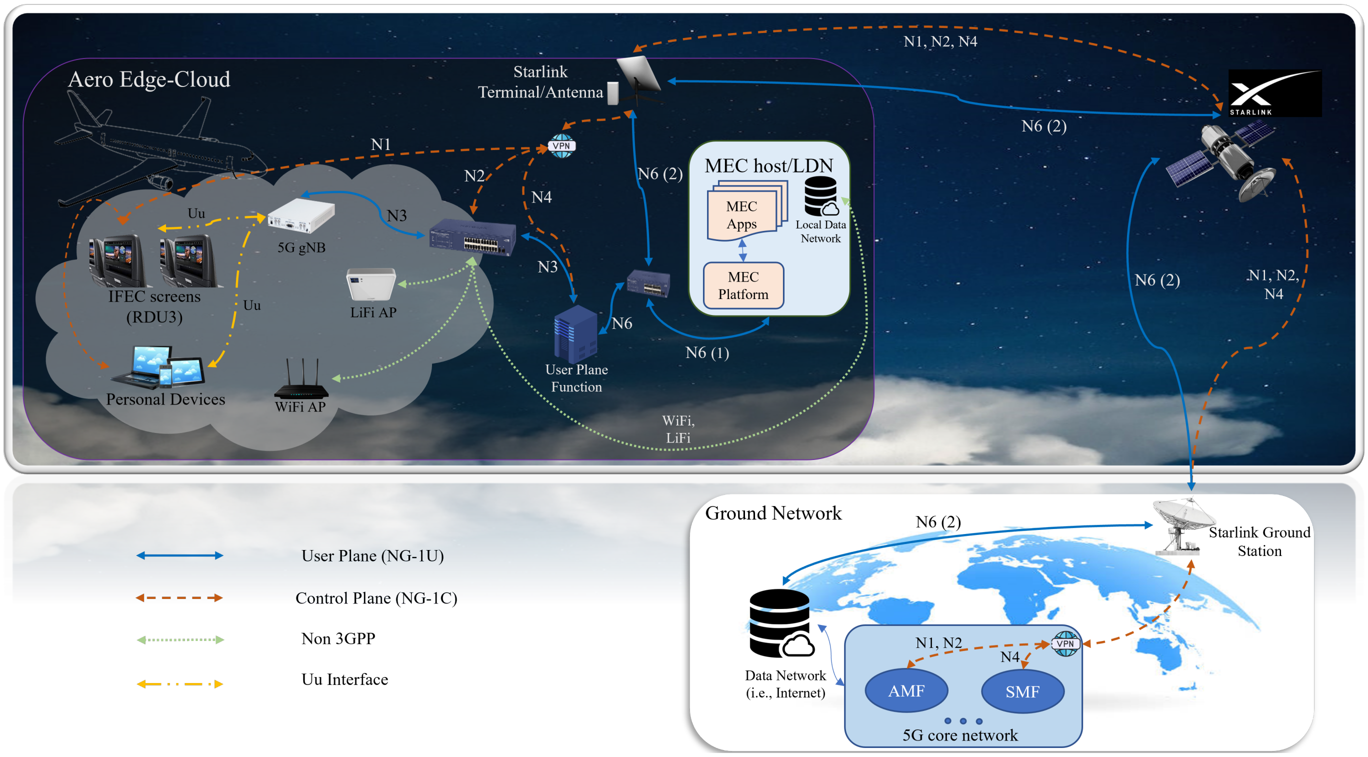

Fig. 1: 5G and beyond Aero Edge-Cloud platform and connectivity system.

a secure and reusable artificial intelligence platform for edge computing. Within the framework of AI@EDGE, we introduce two MEC applications: ML-based Screen Failure Prediction App and Content Recommendation App. The experimental implementation and findings of this work are discussed throughout the paper. The article is structured as follows. In Section II, we introduce the overall architecture of an aeronautical network with a focus on integrating 5G with Starlink for the on-board Aero Edge-Cloud. Section III discusses two applications that are served inside the Aero Edge-Cloud, followed by a discussion on the validation of these applications in Section IV. Finally, in Section V, we conclude the work and discuss future prospects.

II. Architecture of 5G aeronautical Network

The architecture of the proposed Aero Edge-Cloud and connectivity system on an aircraft is presented in Fig. 1. The figure provides an illustration of the 5G system architecture spanning across the edge site of an aeronautical network and a ground infrastructure by means of LEO satellite connectivity. The entire network is divided into three main parts:

- i Aero Edge-Cloud: which comprises the on-board network including the Radio Access Network (RAN), MEC host, and the LDN. The existing IFEC hardware, such as IFEC screens (or Removable Display Units (RDUs)), and servers are also included in this part.
- ii Satellite backhaul: which is assumed over Starlink LEO constellation.
- iii Ground network: consisting of the 5G core network and the central data network (i.e., the internet).

A. Edge-Cloud in an Aircraft

For the sake completeness, the RAN in Fig. 1 includes also two non-3GPP technologies like WiFi and Light Fidelity (LiFi)) together with 5G to provide simultaneous and efficient connectivity to the IFEC system (such as wireless RDUs) and COTS user devices (e.g., smartphones). In this paper, WiFi and LiFi are not addressed, but it is worth mentioning them since WiFi is nowadays the dominant technology on-board and LiFi is seen as a potential successor. From Fig. 1, the Control Plane (CP) traffic is always directed to the ground-based 5G core network through the satellite backhaul, while the UPF is deployed on-board the aircraft. The Access and Mobility Management Function (AMF) and Session Management Function (SMF) are reached by establishing the 3GPP-defined N1, N2 and N4 interfaces for user authentication and session management respectively, [10].

The User Plane (UP) traffic can be either directed to a ground network or on-board the aircraft to access LDN content like in-flight entertainment media. As a result, users are divided into two categories: i) IFEC users who access on-board content and ii) regular users who mainly require internet access. This differentiation is achieved through different Data Network Name (DNN) directing users to the required Data Network (DN). In practice, the 5G core assigns two IP address pools for the users after the attach procedure and the UPF creates two Protocol Data Unit (PDU) sessions for each pool of IPs, as depicted by N6(1) and N6(2) in Figs. 1 and 2. Similar to the CP signals, the N6(2) travels through the satellite connection with one notable difference: while the CP signals (i.e., N1, N2, and N4 interfaces) are routed to the corresponding Mobile Network Operator (MNO) on the ground via a Layer-3 Virtual private network (VPN), traffic over N6(2) is sent to the ground directly to reach the internet. In the system, the N6(1) interface sets up the PDU session for communication between the wireless IFEC devices and the on-board servers, specifically at the MEC host level and the LDN. As depicted in Fig. 2, the MEC host encompasses the MEC platform coupled with the NFVI to provide storage, compute and network resources to run MEC applications [3]. This placement of the MEC host at the edge enhances the accessibility of resources in close proximity to the on-board users. The paper further explores the two MEC applications designed for the Aero Edge-Cloud network in Section III, which run within containers on the virtual infrastructure managed by the Virtualized Infrastructure Manager (VIM).

The other components, indicated by the red boxes in Fig. 2, including the MEC Orchestration (MEO), MEC platform manager, monitoring, and database, are part of the AI@EDGE Network and Service Automation Platform (NSAP) and Connect Compute Platform (CCP) [11] that will eventually be integrated with the Aero Edge-Cloud. However, their specific details are beyond the scope of this article and are only shown in the figure for the sake of completeness and clarity as part of the AI@EDGE platform.

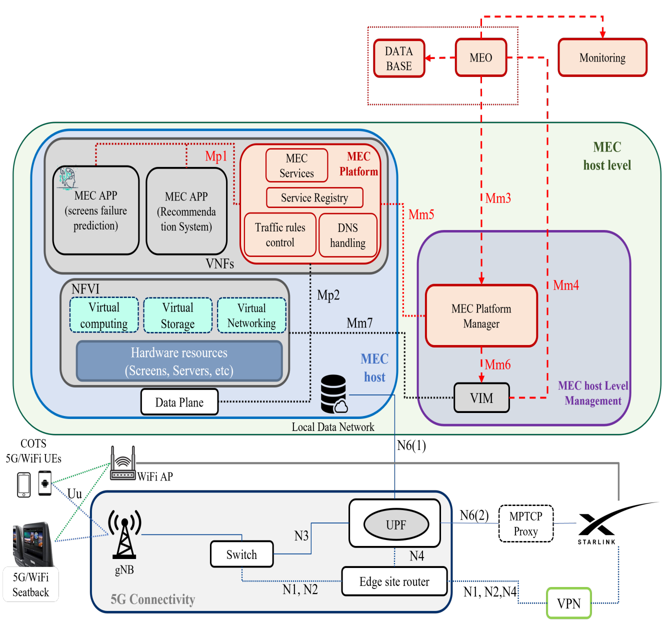

Fig. 2: MEC integration with the on-board RAN

B. 5G Network over LEO satellite Backhaul

Geostationary Orbit (GEO) satellites provide global coverage with a limited number of space segments. However, the high latency of +500ms makes them inadequate for broadband services that require also low latency and low-cost bandwidth. On the other hand, LEO satellites, which are located at a closer proximity to the earth (between 500-2000 Km), offer lower latency compared to their geostationary counterparts. Although the lower orbit of these satellites implies a much larger number for global coverage, different companies have recently started deploying LEO satellite constellations to provide high-speed internet access worldwide. In [12], the authors made a comprehensive comparison between three private LEO satellites (Starlink, OneWeb and Kuiper) and found that Starlink outperforms its competitors OneWeb and Kuiper. For our lab setup, we selected and measured a Starlink connection performance for a single user, showing average end-to-end latency of around 30ms and a data rate over 150 Mbps. The 3GPP release 17 [13] introduced a satellite backhaul scenario that is positioned between the 5G core network and the terrestrial access network to transport the N1, N2, and N3 interfaces. In contrast, the aircraft scenario proposed in this article considers the use of an on-board UPF to access to the LDN and the internet. Therefore, the 5G interfaces of N1, N2, and N4 are transported through Starlink, as well as the N6 interface that is used for access to an external data network such as the internet. In our aircraft network, the N3 interface remains on board between the 5G RAN and the UPF, as illustrated in Fig. 3.

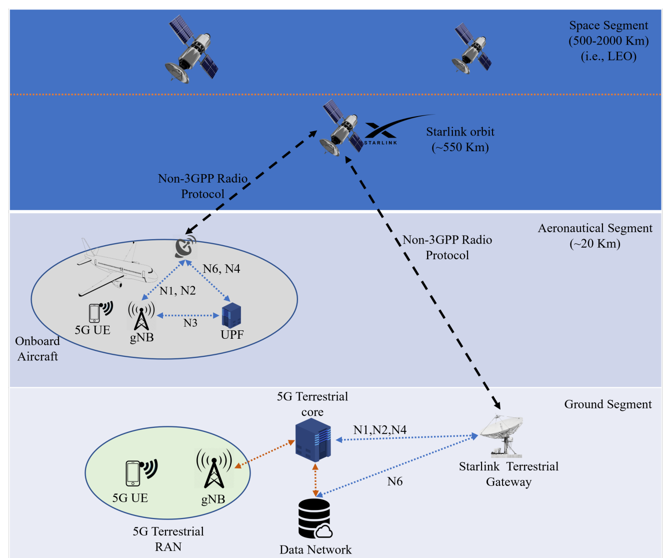

Fig. 3: Starlink as a backhaul for the 5G and beyond Aero Edge-Cloud system

III. AIFS over Aircraft Edge-Cloud

The Artificial Intelligence Functions (AIFs) are defined as AI-enabled end-to-end applications that are deployed across the AI@EDGE platform [9]. This section introduces two MEC applications, a.k.a. AIFS, for the Aero Edge-Cloud to validate the AI@EDGE platform concept.

A. ML-based predictive maintenance

1) IFEC screens failure prediction App: The prediction model we describe herein lies within ML-based predictive maintenance [14] and is designed based on data gathered from IFEC screens (i.e. RDU) in service. Normally, screens can be in a non-functional state due to several reasons internal to an RDU device such as temperature, aircraft type, software release and software update, hardware type, and others. By predicting such non-functional states, we aim at reducing maintenance time and avoiding downtime of RDUs affecting passengers' quality of experience.

2) Data Collection and Aggregation: ML models depend on the availability of reliable data sets for enabling algorithms to make also reliable predictions. We gathered the required data set from an existing commercial SQL database in which data (or logs) collected from different airline aircraft are systematically stored. Normally, such data is used by maintenance departments to analyze the performance of different aircraft systems offline, including IFEC devices. The multi-label historical failure data set for the RDU is obtained from the overall data stored by the repair team, and it thus involves the data-gathering process from different sources.

3) Data set overview: The gathered historical data comprise features that describe the state of an IFEC screen. These features are described by an n-tuple including, but not limited to, {*id, average temperature, flight duration, last software update,...*}. Each subset of the n-tuple in the database contains different RDU characteristics and identifies a unique screen. The database exhaustively contains different types of attributes described by raw data. Some data fields can be fed in input directly to the ML model, whereas others have to be dropped. In this way, the first step consists of pre-processing raw data to obtain a meaningful data subset. The data types of the remaining features are mainly categorical (textual) and numeric. Apart from this, the data set contains an important feature that describes whether the screen was already broken or replaced, and it is our target label class. This feature is represented by a binary variable with the value "0", indicating the class of normally working RDU and value "1", representing the class of RDU predicted to be defective by the ML model and that must be replaced. Thus, based on the historical values stored in the data set, we can categorize the use case as a binary classification machine learning problem and we conducted experiments using different algorithms [15].

B. Content recommendation App

In the literature, there are several types of content recommendation systems such as popularity-based, content-based, and collaborative filtering models [16]. Among those, the main advantage of the popularity-based model is that recommendations can be generated even for users (e.g., passengers) whose preference for IFEC content selection is unknown. For movie content, most of the existing popularity-based models use the ratings from the users to calculate a popularity score for each movie in the IFEC database [17]. The drawback of existing airline data sets is that they lack passengers' ratings of the movies. However, other features such as the watching ratio and the number of viewers for each movie can be used. Hence, in the popularity model we propose such features alongside the IMDb rating and the movie release year are taken into consideration to calculate the popularity score of a movie. The watching ratio represents the average proportion of a movie watched by the passengers. The watching ratio is calculated based on three types of records from the airline data sets: *video start*, *video stop*, and *video complete*. The video starts logging records when the user initiates to watch a video; the video stop logging records when the user stops the playback of a video; the video completes shows when the video plays to completion. With these records, the watching time can be calculated and divided by the total run time of that movie to obtain the watching ratio. The IMDb score is calculated with the average rating and the number of votes, which are two features available from the public IMDb data set. The linear formula used to calculate the popularity of each movie is as follows: $Popularity = w_1 \times (watching\, ratio + number\, of\, viewers) + w_2 \times release\, year + w_3 \times IMDb\, score$, where w_i, $i = 1, 2, 3$, are adjustable weight parameters used to emphasize/de-emphasize the contribution of each criterion selected to output the popularity score in the popularity-based method. We should mention that while the weights can be assigned empirically, they can also be the result of an optimization. According to the computed popularity, m movies rated with popularity score in decreasing order are displayed to the passengers' RDU during a flight. We also point out that the result of the recommendation may vary across different airlines and routes based on the historical data set.

Further, we conducted a series of experiments to understand the relationship between movies that are watched the most by passengers and their release years based on the historical data set gathered across three months. The results showed that passengers tend to watch the most movies released in recent years. However, some old movies can be still very popular and should not be excluded from the recommendation. Therefore, we empirically defined two different ways of assigning weights for old and new movies. Accordingly, in the case of old movies, a higher weight is assigned to the watching ratio and the number of viewers compared to the other ones. Vice versa, in the case of new movies, a higher weight is assigned to the IMDb score and release year. In the future, the popularity model can be tested by different airlines.

IV. Implementation and Validation

In this section, we present the test bed implementation of the system components that were discussed previously. We also point out that measurements of the different time intervals required to collect pre-process data and provide content recommendation to passengers based on the popularity-based model is still ongoing in our test facility. Therefore, the focus here on the ML-based model for RDUs failure prediction.

A. Aeronautical test rack development

In order to deploy the Aero Edge-Cloud network and integrate it with the central 5G core on the ground through Starlink, an aircraft edge-cloud test-rack with 21 RDUs was developed as it can be seen from Fig. 4. The test-rack reproduces the IFEC system commercially deployed on-board an airline aircraft and the Aero Edge-Cloud is a mapping into the aircraft edge-cloud according to a suitable network embedding. An RDU is the 3rd generation of aviation certified IFEC screens manufactured by Safran Passenger Innovations and they are used worldwide by aircraft passengers to consume media content from the in-flight entertainment media server, nowadays. Besides the screens, the test-rack includes also one Supermicro server that implements a LDN and stores

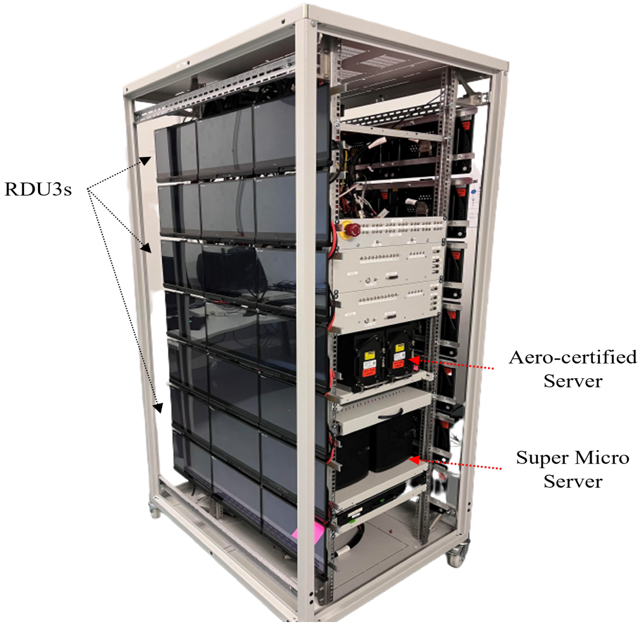

Fig. 4: The test-rack embodiment of the Aero Edge-Cloud

all the onboard media including music, movies, etc. This is a COTS server with a very compact form factor (similar to an aero-certified server) with 12 CPU cores and 128 GB RAM. With this powerful hardware, it is possible to handle some of the computation-intensive applications that run within the Aircraft edge-cloud infrastructure. Referring to Fig. 2, the srsLTE Stand Alone (SA) gNB [1] is deployed, alongside the corresponding software-defined radio (i.e. Ettus X310 USRP) [2], to configure the 5G RAN onboard the aircraft. Moreover, the Aircraft edge-cloud test-rack leverages on a Kubernetes VIM to manage the MEC host that consists, but is not limited to, initiation and termination of the RDU failure prediction and recommendation system MEC Apps.

B. IFEC screens failure prediction and ML experiments

The preliminary ML experimentation activity was conducted on an existing platform to run the initial tests for IFEC screens (i.e., RDUs) failure prediction before transferring the more advanced experimentation to the Aero Edge-Cloud. To this extent, we created a criterion list in which four different platforms were evaluated, including Azure [3], H2O [4], TPOT [5], and NNI [6]. The criteria involved Graphical User Interface (GUI) support, AutoML support (the process of automating the tasks of applying machine learning), Multi-core support (splitting the work across multiple CPU cores), contain ensemble models (combination of more than one model), among the most relevant. Based on these criteria, H2O was identified as the best suited as it provides thorough functions around autoML, multi-core support, and and built-in ensemble algorithms [18]. Moreover, it gives either a layman-friendly GUI interface in its web-based version or freedom of personalized coding in its coded-based.

[1]https://docs.srsran.com/en/latest/
[2]https://files.ettus.com/manual/page_usrp_x3x0.html
[3]https://azure.microsoft.com/en-us/products/machine-learning
[4]https://docs.h2o.ai/h2o/latest-stable/h2o-docs/automl.html
[5]https://github.com/EpistasisLab/tpot
[6]https://github.com/microsoft/nni

A pre-processed data set (remove duplicates, null values, etc.) was divided into a part comprising 20% of the meaningful data for testing and an 80% part for training. Referring to the problem described in Sect. III-A, the IFEC screens failure predictions stands as a binary classification problem. Regarding this, H2O AutoML provides several in-built algorithms that we aim to compare: XGBoost Gradient Boosting Machines, H2O Gradient Boosting Machines, Distributed Random Forests (DRF), Generalized Linear Models, and two Stacked Ensemble models (generated by combining other base models). During each training iteration, each model was evaluated using the Area Under The Curve (AUC) of the corresponding Receiver Operating Characteristic (ROC) as a performance indicator. The $0 \leq AUC \leq 1$ allows ranking the algorithms and their capability of distinguishing between target classes. To address instead the data set imbalance, the built-in auto-balancing method based on SMOTE (Synthetic Minority Over-sampling Technique) [19] was used. In each training iteration, the top-performing model (i.e., showing the highest AUC) was selected and afterward the F1 score was calculated during the testing phase (i.e., on the 20% of the whole data set). AUC and F1 scores are commonly used metrics for evaluating classification performance when the data set is imbalanced. The data set that was initially used for training and testing the algorithms mentioned above included all features shown in Fig. 5 with their importance weight. The experiments reveal that very low-weight features have marginal contribution to the F1 score, but can largely affect the computation time during training. We clarify that the training time is an H2O platform indicator that includes all classification algorithms. Hence, the training and testing experiment was repeated with the top 12 most important features and then with the top 6 features as discussed below. Training and testing experiments were conducted on a Dell PowerEdge server with AMD EPYC 7402P processor, 24 CPU cores, 64 GB of RAM with Linux OS Kernel version 4.19 and Debian 10 distribution. We further specify that training iterations were repeated 10 times for the two data sets with the top 12 and 6 features, as mentioned already. The training time and F1 score results are shown in Fig. 6 and in Fig. 7, respectively. Training time was computed in the H2O Web-based version (indicated as Flow in the figure) and H2O Code-based version (indicated with R in the figure). Fig. 6 allows us to conclude that H2O Code-based version provides lower training time. In addition, the AUC rank that was computed at each training iteration allows us to select the DRF algorithm as the one with the highest AUC value (i.e., 0.9865 against 0.9861 for the Stacked Ensemble). Referring to Fig. 7, the F1 score of the DRF algorithm was computed in H2O Web-based and Code-based comparing again the top 12-feature and 6-feature data sets. The 12-feature data set in the H2O Code-based version provides the highest F1 score. We conclude that the IFEC Failure Prediction App can rely on the H2O Code-based platform with the DRF algorithm.

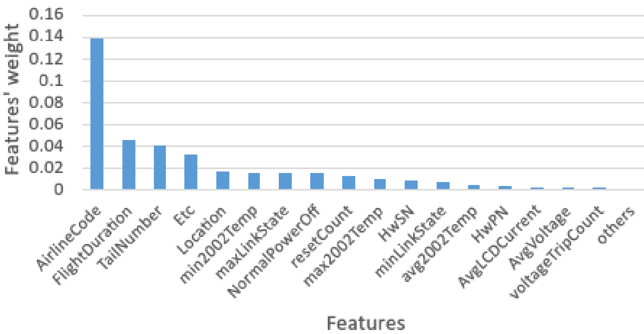

Fig. 5: Overall features importance ranking

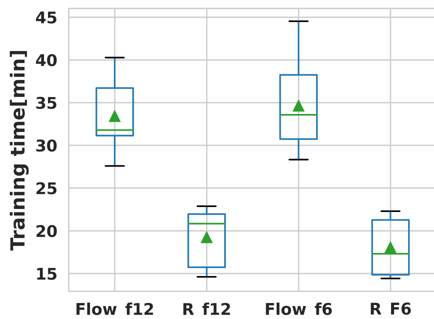

Fig. 6: Training time for all algorithms in H2O web-based (denoted as Flow) and Code-based (denoted with R) using top 12 and top 6 features data sets.

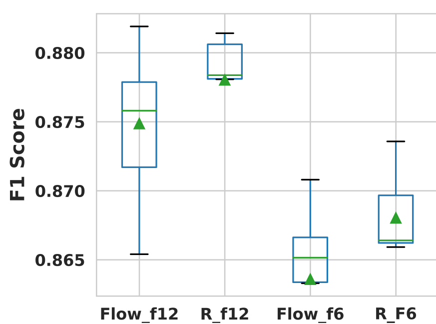

Fig. 7: F1 score of DRF in H2O Web-based and Code-based version for top 12 and 6 top features data sets.

V. Conclusion and Future Works

This article presented a 5G and beyond test-bed implementation for an aeronautical environment with the addition of LEO satellite backhaul. As mentioned, the work was developed as part of the H2020 AI@EDGE project. The system we have presented is divided into three distinct parts: the on-board Aero Edge-Cloud network, which leverages the paradigms of the ETSI MEC; the Apps used to deliver ML-based predictive maintenance for IFEC screens, as well as popularity-based content recommendation for the airlines; the increasing opportunity provided by rising LEO constellations such as Starlink. Our test bed results already allowed us to test the ML-based IFEC screens failure prediction App, which exhibits an efficiency of 87% using standard model validation techniques. The second developed MEC App, the recommendation system, was mostly discussed and takes into account new parameters such as the passenger' watching ratio, the number of viewers in previous months, and the release year, which were derived from historical data sets obtained from various airlines. This innovative approach provides a new dimension to creating different recommendations tailored specifically to inflight passengers uniquely taking into account the airline and the airplane route. In future work, we plan to simulate the effect of mobility on the system to fully demonstrate the capabilities of this innovative cutting-edge aeronautical network design.

VI. Acknowledgment

This work has been performed within the EU's Horizon 2020 project AI@EDGE (grant agreement No. 101015922).

References

[1] E. Temprado *et al.*, "In-Flight Entertainment and Connectivity in the 5G Era: The 5G ESSENCE Experimental Platform," in *Proc. of EuCNC*. IEEE, 2019.

[2] S. Kekki *et al.*, "MEC in 5G Networks," *ETSI white paper*, vol. 28, no. 28, 2018.

[3] "Multi-access Edge Computing (MEC) framework and reference architecture," ETSI, GS MEC 003 v2.2.1, Tech. Rep., 2020.

[4] S. Deng *et al.*, "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, 2020.

[5] A. Ghosh *et al.*, "5G Evolution: A View on 5G Cellular Technology Beyond 3GPP Release 15," *IEEE access*, vol. 7, pp. 127 639–127 651, 2019.

[6] "TR 38.821, "Solutions for NR to Support Non-Terrestrial Networks (NTN)," V16.0.0," Tech. Rep., January 2020.

[7] "TR 22.822, "Study on Using Satellite Access in 5G," V16.0.0," Tech. Rep., August 2018.

[8] X. Lin *et al.*, "5G From Space: An Overview of 3GPP Non-Terrestrial Networks," *IEEE Communications Standards Magazine*, 2021.

[9] R. Riggio *et al.*, "AI@ EDGE: A Secure and Reusable Artificial Intelligence Platform for Edge Computing," in *Proc. of EuCNC/6G Summit*. IEEE, 2021.

[10] "TS 123.501 , "5G; System Architecture for the 5G System" V15.3.0," Tech. Rep., September 2018.

[11] D4.1 Design and Initial Prototype of the AI@EDGE Connect-Compute Platform. Accessed: 27.01.2023. [Online]. Available: https://aiatedge.eu/wp-content/uploads/2022/03/AI@EDGE_D4.1_Design-and-initial-prototype-of-the-AI@EDGE-connect-compute-platform_v1.pdf

[12] O. B. Osoro and E. J. Oughton, "A Techno-Economic Framework for Satellite Networks Applied to Low Earth Orbit Constellations: Assessing Starlink, Oneweb and Kuiper," *IEEE Access*, vol. 9, pp. 141 611–141 625, 2021.

[13] "TR 23.737, "Study on Architecture Aspects for Using Satellite Access in 5G," V17.2.0," Tech. Rep., March 2021.

[14] M. David Dangut *et al.*, "A rare failure detection model for aircraft predictive maintenance using a deep hybrid learning approach," *Neural Computing and Applications*, pp. 1433–3058, 2022.

[15] V. Bahel *et al.*, "A Comparative Study on Various Binary Classification Algorithms and Their Improved Variant for Optimal Performance," in *Proc. of TENSYMP*. IEEE, 2020.

[16] A. Al Jobaer *et al.*, "An Advanced Recommendation System by Combining Popularity-Based and User-Based Collaborative Filtering Using Machine Learning," in *Proc. of ICSCT*. IEEE, 2021.

[17] S. Chaudhary and C. Anupama, "Recommendation System for Big Data Software using Popularity Model and Collaborative Filtering," pp. 551–559, 2020.

[18] E. LeDell and S. Poirier, "H2O AutoML: Scalable Automatic Machine Learning," in *Proceedings of the AutoML Workshop at ICML*, 2020.

[19] K. W. Bowyer *et al.*, "SMOTE: synthetic minority over-sampling technique," *CoRR*, vol. abs/1106.1813, 2011. [Online]. Available: http://arxiv.org/abs/1106.1813