A Framework to Use Machine Learning and Provide Assurance for Safety-Critical Air Traffic Management Applications

Methodology and Examples of Applications

Luca De Petris
Euranova
Mont-Saint-Guibert, Belgium
luca.depetris@euranova.eu

Ivan De Visscher
Wake Prediction Technologies (WaPT)
Louvain-la-Neuve, Belgium
ivan.devisscher@wapt.be

Guillaume Stempfel
Euranova
Marseille, France
guillaume.stempfel@euranova.eu

Frédéric Rooseleer
EUROCONTROL
Brussels, Belgium
frederic.rooseleer@eurocontrol.int

***Abstract*—The continuous expansion of the use of Artificial Intelligence (AI) in new domains brings along new challenges. When dealing with safety-critical applications, where any system failure could lead to catastrophic events it is of vital importance to be able to safely use AI and provide overall operation efficiency benefits. This work proposes a framework to use AI in safety-critical applications. It focuses on the use of AI and more specifically Machine Learning (ML) in the Air Traffic Management (ATM) domain. Three applications of the proposed framework implemented and tested for three major European airports are then described.**

Keywords-air traffic management; machine learning; safety-critical;

I. INTRODUCTION

For several years, Artificial Intelligence (AI) has spread across different industries to try to model and solve problems, or to optimise processes and operational predictions with the use of data. Data can be seen as a gold mine from where it could be possible to learn from the past to then predict the future. The spread of AI and specifically Machine Learning (ML), so the algorithms and statistical models that a machine uses to learn patterns and behaviour from data, has also reached Air Traffic Management (ATM).

For some ATM applications, as in other domains, a high level of safety needs to be guaranteed, which constitutes a significant challenge in adopting the usage of ML. This introduces the broad concept of safety-critical applications, as opposed to decision-support solutions, where any system failure could lead to catastrophic events such as fatality or serious injury to humans, severe damage to equipment, or environmental harm, as for example in avionics or automotive [1]. In the context of this paper, safety-critical applications will be defined as applications that need to respect a predefined homogeneous error rate. More specifically, an error rate will be defined as an acceptable (maximum) percentage of times the system is allowed to make an error in its prediction and homogeneity will refer to the requirement that the error rates should be constant across different feature combinations of the system input space. In such safety-critical applications, the use of ML needs to be reliable (i.e., safe), yet it should also provide operational efficiency benefits.

The advantages of the use of AI over traditional analytical solutions spread in a wide spectrum: possibility to have more accurate prediction, possibility to achieve homogeneous error rates across different situations and possibility to have an automated model training, validation, testing and update pipeline.

As always, all that glitters is not gold, so also in this context, AI (and in particular ML) brings with it some challenges. There is indeed a need to prove that the ML models are still safe for occurrences of rare or even previously unseen events so as to be able to use the solution in the real world and covering the operational variability, where the past can explain, characterise and be used to predict the future up to a specific point of evolution.

The application of AI in safety-critical applications has paved the way for the need to define how to assess, provide assurance and certify AI models. In that framework, the European Union Aviation Safety Agency (EASA) has released a first guidance and approval basis for Level 1 machine learning applications [2][3]. An important point of this guidance is the clear identification of the Operational Design Domain (ODD), defined to capture specific operational limitations and assumptions.

To address the need for robust safety assurance, a framework to use ML models in safety-critical applications is proposed in this paper. The framework is composed of five major blocks:

- Predictive strategy
- Coverage functions
- Conservative strategy
- Strategy selector
- Validation & Verification component

The predictive strategy itself is composed of two sub-components: the predictive and the buffer models. The predictive models have the objective to predict the expected value of the indicator(s). The buffer models are introduced as a way to capture the variance of the target(s) in order to achieve homogeneous error rates across the feature space while saving performance as much as possible. The coverage functions are responsible for defining in which conditions the predictive models are safe to be used. As fall back, a conservative strategy shall also be defined based on more traditional data analysis or based on operational expertise. Decision trees are then used, as strategy selector, based on the previously defined coverage functions in order to decide when the predictive models should be used and when instead the conservative strategy will be required. Finally, the Validation & Verification component is used to assess the overall safety of the implementation.
The method is applied to three examples of applications described in Section IV.

II. USING AI FOR ATM

AI has been an integral part of the European Commission's Digital Single Market Strategy since 2017, supported by €1.5 billion co-funding under the Horizon 2020 programme from 2018 to 2020. This was followed by a roadmap and establishment of the European AI Alliance in 2018 to put Europe firmly on the path to becoming a leader in the AI revolution.
EUROCONTROL held an inaugural Forum on Aviation and AI in early 2019 which brought together key players and served as the launch point for a European AI Aviation Network [4].
ATM stands to benefit significantly from AI by virtue of its reliance on repetitive activity – which lends itself to analysis and machine learning. In addition, much of the complexity is embedded in the driving factors that deliver safe air traffic control: for example, flight planning, flow management, safety assessments and conflict prediction. It is no surprise the industry is adopting the technology to enhance both planning and operational activities, and early trials by EUROCONTROL reveal gains of between 20 to 30% in terms of predictability and efficiency [4].
AI is currently being tested and used in many applications in the ATM domain, for Network, En-route and Airports area. For example, in the Airport and Approach-Departure operations, it is currently being used to help predict aircraft load factors as well as individual passenger counts per aircraft [5]. Many works are also related to the application of AI for the prediction of the Runway Occupancy Time (ROT) (see, e.g., [6]-[8]), the prediction of flight departure and arrival routes [9]-[19] and the optimization of landings [20][21] and take-offs [22].
Applications of AI can also be found in wake vortex detection [23], wake behaviour prediction [24] and the prediction of severe weather events developing at small spatial and temporal scales impacting airports and flights[25].
Several researches can be found around the automation of conflict detection and resolution [26]-[28], flight plan prediction [29], prediction of go-arounds [30] and delay predictions [31]. Furthermore, the use of AI models can also be found in Air Traffic Controllers (ATC) speech recognition tasks [32][33].
Note that a large part of these developments was also performed in the framework of SESAR projects.
Some of the above-listed applications are considered as being-safety critical and needs for certification before any operational deployment. Certification processes are well-known for classical deterministic systems, but as soon as data and stochasticity are involved, these processes (usually based on formal methods) are not suitable.

III. SOLUTION FRAMEWORK

The solution proposed in this paper takes its roots in the need to understand when it is possible to efficiently and safely use the ML models (i.e., providing operational benefits while guaranteeing the required safety level) and when instead it is necessary to restore to a more conservative solution.
The work presented in this paper aims at defining a framework to be able to use AI and provide assurance for safety-critical ATM applications with a predefined level of safety. This section describes the proposed framework considering both safety and the benefits brought by the use of AI.
Note that the proposed framework is limited to the training and validation of models and the definition on how to use them in an operational environment. The actual implementation and deployment of the solution in an operational tool (e.g., an Air Traffic Control support tool) is not covered. The framework does hence not contain components related to, e.g., implementation verification, live monitoring or on how online learning could be used.

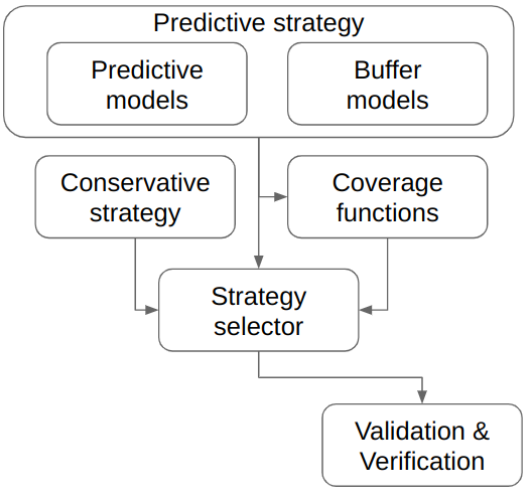

Figure 1. Overview of the components that form the proposed solution.

Figure 1 gives an overview of the components that form the proposed solution: Predictive strategy, Coverage functions, Conservative strategy, Strategy selector and finally, the Validation & Verification component. Each component is further presented below in terms of why it is needed and how it can be defined.

In the following, we focus on regression tasks. Let define $X \in \mathbb{R}^n$ as the set of descriptors (or features). Let define $Y \in \mathbb{R}$ as the set of targets. Let D be a distribution over $X \times Y$.
Typically, we aim at learning a function f: X → ℝ, where $X \in \mathbb{R}^n$ is the set of features and the image the target indicator.
Note that for the sake of conciseness, we consider that all features are numerical. This is made without loss of generality, since categorical features can easily be encoded into numerical features.
We define the error rate as $E_{(X,Y)\sim D} \sim$ D $1_{(f(X)<y)}$, where $1_z = 1$ when the Boolean expression z is true.
The error rate is the probability that f underestimates the target for a random sample drawn from D. Note that depending on the problem, the error rate could also be defined as an overestimation rate.

A. Predictive Strategy

This first component of the framework has the objective of predicting the quantities at interest with the double objective of ensuring the required levels of safety and optimizing the operations efficiency.
For explainability purpose, we propose to split the responsibility of this prediction between two components:

- Predictive models
- Buffer models

1) Predictive Models

The predictive model is an estimator of the target given the input features. For example, should the application deliver the predicted Runway Occupancy Time (ROT) for a flight, then in this component of the framework, a model will be learned from a set of features with information about the flight in order to be able to predict an expected ROT.
More formally, a predictive model is a function $f: X \rightarrow \mathbb{R} \in F$. F is the space of functions in which the optimization of the predictive model is done. It can be the set of linear regressor, the set of random forests… Ideally it should capture the expected behaviour of the target, and typically aims at minimising the Least Squared Error (LSE) on the distribution D: $f = \min_{f \in F, (x,y) \sim D} E\left((f(x) - y)^2\right)$.

2) Buffer Models

Because designing buffers with a 0% error rate level would lead to over-conservative (and unrealistic) design and because other safety mitigations exist for ATM applications (e.g., the margin compared to the reference spacing minima applied by air traffic controllers in their separation delivery), one important parameter to be defined when using the proposed framework is the error rate considered acceptable for the considered application. This error rate is defined as the percentage of times the quantity of interest can be underestimated (or overestimated, depending on the task).

Those design (or target) error rates need to be defined by the Service Provider (also as part of the local implementation safety assessment and subject to agreement with the Regulation Authority), and practically can be derived from operational experience to remain at least as safe as today (e.g., what is the level of under-spacing observed today in a specific airport?).
In order to take into account these target error rates, what is here referred to as buffer models is introduced. The buffer models predict an offset that must be added to the output of the predictor such that the error rates are likely respected. The buffer models have the objective of learning a minimum quantity to be added to the predictions made by the predictive models by targeting the predefined error rate. Differently than the predictors that target an average behaviour, the buffer models capture the variance of the predictor error. The buffers shall:

- Be large enough to respect the target error rates
- Be small enough to limit the overestimation of the target which would make the system inefficient
- Have good properties of homogeneity (i.e., not exhibiting excessive error rates in some edge cases). In other words, the error distribution shall be as homogeneous as possible across the feature space.

Typically, for a given predictor $f: X \rightarrow Y$, the buffer model is defined as a function $b_f : X \rightarrow \mathbb{R}$, and for a sample (x, y) the prediction including the buffer writes as $f(x) + b_{f(x)}$.
The buffer models target a quantile of the difference between the predicted value of the quantity of interest, obtained with the use of the predictive models, and the value of the same quantity but computed from the ground-truth data. The quantile to target is defined based on the a-priori determined target error rate.
For example, if a 2% error rate is considered acceptable then the buffer model would be set to target the 0.98 quantile of the target distribution, meaning that theoretically in 98% of the cases the buffer predicted will be equal or bigger than the target.

Note that the proposed two sub-components for the predictive strategy, predictive models and buffer models, could be merged together and result in a unique model that directly learns the desired targets while respecting the error rates. Further reasonings about why we recommend keeping the two sub-components separated can be found in Section V.

B. Coverage Functions

Once the predictive models and buffer models are learned, there is the need to assess when they can be used with confidence respecting the targeted error rates.
This is done by defining coverage functions. Coverage functions assess the predictions made considering a specific input feature or a combination of several input features.
The coverage functions are a set of functions CV = $\{cv_{f,b_f,X_i'}: X_i \rightarrow \{0,1\}\}$ where all $X_i' \subset X$ that output 0 if the confidence in the predictor and the buffer is too low, and 1 if the confidence that the target rate α is respected is high. Basically, a coverage function $cv_{f,b_f,X_i'}$ is designed such that:

$$cv_{f,b_f,X_i'} = 1_{E_{(x,y)\sim D|x\in X_i'}[((f(x)+b_f(x)<y)]<\alpha} \quad (1)$$

This could be done either theoretically, by deriving theoretical bounds from f and b_f, or empirically using an independent dataset to evaluate error rates. This dataset shall not have been previously used for learning the predictor and the buffer models to preserve independence and avoid introducing any bias in error rates evaluation. The choice of features to be used to define the coverage functions can be led by the feature importance or also by the valuable operational expertise. In the case of the ROT predictions, the set of coverage functions can include whether or not the predictions for a given runway respect the targeted error rates or whether or not the predictions for a given aircraft type respect the targeted error rates. Note that these coverage functions are in fact defining the predictive strategy inference ODD identification on a data-driven basis.

C. Conservative Strategy

To be able to still provide predictions for cases for which the predictive strategy (Predictive + Buffer models) is not covered, there is the need to define a backup strategy. This leads to the introduction of what is here referred to as conservative strategy. These models are defined with the sole objective of being safe when used by relaxing the performance requirement[1]. A conservative strategy could be based either on human expertise, basic physics-based rules or even data-driven statistics. They shall be designed in order to complete the satisfaction of the safety criteria and support the approval by regulation authorities.

The conservative strategy could also be envisioned as a backup strategy where the human is completely responsible in making the decisions instead of relying in any kind of statistical model.

D. Strategy Selector

Now that all previous components have been defined, there is a need to determine a way to bring them all together. It is here that what is referred to as strategy selector comes in the picture. The objective of the selector is to be able to decide from the coverage functions decisions, whether predictive models or conservative strategy should be used for a given scenario.

It is important to note that the predictive models and the conservative strategy are completely independent from each other, both in terms of definition and in usage since at inference stage only one or the other will be used for a given input case.

E. Validation & Verification

Finally, the last component of the proposed framework is responsible for verifying and validating (V&V) the overall pipeline (all components listed above) based on a completely independent dataset. This process also aims at assessing the coverage of the system ODD and the robustness of the system.

This V&V is proposed to be performed at two different levels. The first level is a "classical" test of the pipeline on an independent data set (i.e., not used for the model training and validation nor for the coverage determination). In this "local" test phase, the model error rate shall be compared to the target ones globally but also for specific combinations of conditions in order to verify the error distribution homogeneity, ensuring lack of biases and unintended side effects. The second proposed test phase is denoted as "generic" as it aims at testing the whole chain on emulated input data. Those input data are built from a combination of operational experience and statistical analysis of the local data but ensuring to generate very rare events. In this phase, the model results shall be explained based on operational expertise or using an independent baseline model and also considering the probability of occurrence of the input scenarios (i.e., the model chain is expected to be more conservative for rarer events).

IV. Examples of Applications

Three examples of safety-critical operational applications of the presented framework are presented here, developed and tested on SESAR 2020 Wave 2 PJ02 solutions. The development of all three solutions has been led by EUROCONTROL, each relying on traffic and meteorological data from three different major European airports.

A. TBS-ORD

Time-Based Separation (TBS) in the final approach is an operational solution, which uses time instead of distance to separate aircraft on their final approach to a runway. The TBS solution mitigates the negative impact of headwind on runway capacity. Indeed, headwind conditions on final approach cause a reduction of the aircraft ground speed which for distance-based separation results in increased time separation for each aircraft pair, a reduction of the landing rate, and a lack of stability of the runway throughput during arrival operations.

The TBS solution allows stable arrival runway throughput in all headwind conditions on final approach. However, in order to apply TBS, approach and tower air traffic controllers need to be supported by a separation delivery tool which provides a distance indicator (final target distance – FTD), enabling to visualise on the surveillance display the distance corresponding to the applicable TBS minima and taking in account the prevailing wind conditions and integrating all applicable separation minima and spacing needs.

This separation delivery tool, providing separation indicators between arrival pairs on final approach, also enables an increase in separation performance when providing a second indicator (Initial Target Distance – ITD): a spacing indicator to optimise the compression buffers and ensure optimum runway delivery (ORD).

The calculation of the TBS-ORD tool indicators (FTD and ITD) requests to properly model/predict aircraft speed and behaviour in short final and the associated uncertainty. A too conservative definition of buffer can lead to a reduction of efficiency whereas making use of advanced Machine Learning techniques for aircraft behaviour prediction allows improvements of separation delivery compared to today while maintaining or even reducing the associated ATCO workload[20]. For being operationally deployed, such a separation delivery tool allowing the TBS application, directly used by Approach

[1] Note a completely ineffective conservative approach will penalize the whole system efficiency.

Control to space and separate traffic, while improving ATCO performance and management of complex business rules (separation/spacing), has to be demonstrated as fully reliable and meet the determined safety criteria.

For that purpose, EUROCONTROL has developed the enhanced ORD (eORD) solution [21] [34], and 'COAST' (Calibration of Optimised Arrival Spacing Tool) ML training pipeline prototype, that support Air Navigation Service Providers and their Approach ATM system providers, to train Machine Learning Models on historical traffic and meteorological data, for a given airport case. Those models, once integrated into a separation delivery tool, allow the computation of optimised separation to apply and the associated buffer ensuring a level of safety in line with the criteria set in the TBS Safety Case [35].

With the use of a separation delivery tool and COAST, runway throughput can be increased at major airports during peak hours, up to 5% under low wind conditions and even more in stronger wind conditions thanks to the optimisation of the spacing and separation delivery performance.

The eORD / COAST ML-based solution design for TBS-ORD tool calibration, follows the framework presented in this paper.

1) Predictive strategy

a) Predictive models

In order to model the required separations between consecutive landing aircraft, the modelling of the aircraft during final approach is required. Since, in this phase of the flight, aircraft are aligned to the runway and follow a predefined latitude-longitude trajectory, their final approach is modelled as the time needed for the aircraft to reach the runway threshold. This model is hence referred to as the Time-to-fly model. Specifically, one ElasticNet [36] model for each predefined distance from the runway threshold is defined. Time-to-fly models are estimated on a training dataset. For example, one model is responsible for predicting how much time the aircraft needs to fly a ground distance of 2 km to the runway threshold. With the use of the Time-to-fly model, for each aircraft pair, the separation indicators (FTD and ITD) corresponding to the different spacing constraints can be computed.

b) Buffer models

The observed time and distance separations if applying the indicator at minima are then also computed from the recorded data and compared to the applicable spacing constraints (e.g., time-based wake turbulence separation minima). The difference between these two quantities defines the targets for the buffer models. Buffer models are defined as a Gradient Boosting Regressors [37].

2) Coverage functions

Coverage functions are defined independently for a pre-defined set of features: aircraft type, airline, runway, wake turbulence categories, wind band… A feature is considered as covered if the statistical bounds of the empirical error rate are close enough to the design criteria.

3) Conservative strategy

When a flight or a pair is not covered, a fallback is needed. To do so, conservative models are defined for both time-to-fly and buffers.

Time-to-fly conservative models are here built from a combination of the average true air speed behaviour defined per aircraft type and surface headwind band on the training set and the expected headwind profile. In the cases where a type is too rare to compute an average, the worst-case behaviour of the wake turbulence category is used.

Conservative buffers are also computed as high quantiles of the buffer distributions observed on the training set. All conservative buffers are defined per leader aircraft type, follower wake turbulence category and surface headwind band. These conservative models can also be used for previously unseen cases. This conservative approach guarantees that safety is preserved with a limited operational cost in terms of over-spacing. The uncovered pairs are by definition rare enough for not impacting dramatically the overall performance.

4) Strategy selector

The strategy selector, as described in the general framework, is defined from the use of the coverage functions in a specific order to decide if for a given input flight pair, the predictive or conservative models should be used. A flight is covered if its airline and aircraft type are covered. A pair is covered if both flights, runway, and the combination between leader categories, follower categories and wind conditions are covered. Note that here we are doing a strong assumption on features independence that is discussed in Section V.

5) Validation & Verification

For COAST usage validation and verification, two methods were developed in order to assess the developed model chain. Both are assessing the accuracy of the obtained separation and spacing indicators (FTD and ITD) when using the ML models and not directly of the different sub-models themselves.

The first method assesses the model accuracy against local operational data corresponding to the environment on which the models were trained (but obviously using data independent from the training dataset). The validation consists in checking that the FTD and ITD design criteria are met when computing the FTD and ITD for that independent test dataset. The assessment is performed globally but also making the distinction between various pair types and wind conditions. It then also allows assessment of the model error homogeneity. The objective of this assessment is to ensure to avoid observing global error rates in line with the FTD and ITD design criteria yet with some rare events systematically failing but compensated by other more frequent events.

The second method aims at testing and explaining the ML model behaviour for generic inputs representative of what could be encountered operationally. For that purpose, for various input conditions (various aircraft types, various runways, various wind conditions, etc.), the ITD, FTD and compression values computed using ML models and methodology are compared to those obtained when using a generic analytical model that is explainable and with parameters calibrated on another independent dataset. Because the ML approach uses much more features compared to the analytical model that only

uses aircraft type and wind as input, the ML results are expected to be more accurate (hence with lower buffers) for cases that are frequent. On the other hand, the COAST methodology increases the buffer in the indicator computation for rarer cases through the use of conservative models in the FTD and ITD computation decision trees. This approach then also allows model assessment in extreme cases not necessarily found in measurement database.

This generic test then also supports the explainability of the ML models. The ML approach is indeed expected to be globally in line with the results from a simple analytical approach based on knowledge on flight behaviour. It shall however allow further refinement and hence optimisation for frequent events for which the model was able to "learn" whereas it should be conservative for rarer events.

It also performs checks on the system stability so as to ensure that the system predictions do not change much when the training data is slightly modified. Furthermore, it checks specifically the stability for edge cases so samples with one feature on the border of the ODD, for feasible corner cases so samples with most (or all) features on the border of the ODD and for cases that present novelties so samples with some feature values that were never seen before in the training data.

The presented application of the framework is so far the way being proposed for providing assurance and ensuring compliance with TBS safety criteria, as well as addressing objectives of the EASA guidance [2], in view of the implementation of the eORD solution and supporting local safety case.

B. Optimised Spacing Delivery for Departure

The Optimised Spacing Delivery (OSD) tool [38] aims at displaying to the Air Traffic Controller an automated digital countdown timer which provides an optimised clearance time ensuring that all separation and spacing constraints will be satisfied between two consecutive departing aircraft. The advantage of such a tool is that it also supports (and hence enables) the application of complex separation schemes, such as pairwise separation schemes or weather dependent separation, whether time or distance-based. For each aircraft pair, the OSD tool takes into consideration all applicable separation and spacing minima in order to calculate an optimized clearance time displayed through the countdown timer.

As for TBS-ORD, ML techniques can be leveraged in order to predict aircraft departure behaviour more accurately and hence further optimized spacing delivery [22]. This enhanced OSD (eOSD) solution with ML is being developed by EUROCONTROL in SESAR 2020 Wave 2 [39]. Yet, and as for TBS-ORD, the provided clearance time must be proven to be safe.

1) Predicitve strategy

a) Predictive models

Assuming that aircraft during the first part of their flight are instructed to strictly follow a predefined path, known as the Standard Instrument Departure (SID) path, four ML models are then required to describe the departing trajectory of an aircraft: 1) a Rolling Time ML model, 2) a Rolling Distance ML model, 3) an Altitude to Time ML and, 4) a Time to True Air Speed (TAS) ML model.

The Rolling Time and Rolling Distance models are needed to describe the aircraft when it is moving along the runway before the rotation point. The Altitude to Time model describes the time needed by the aircraft to reach a specific altitude while flying along the SID path. The time to TAS model provides the TAS of the aircraft as a function of time while the aircraft is flying along the SID route. The trajectory and speed profile were modelled this way as it is assumed that the altitude to time and time to TAS models, do not depend on the wind; meaning that wind data was not required for the training of these ML models.

b) Buffer models

In addition to the four ML models required for predicting the trajectory and the speed profile of the aircraft, an additional ML model, called the buffer model, was also defined. The buffer ML model is required to cover any uncertainties caused from the previous four models and their combination as well as from the wind variability on the initial departure path.

2) Coverage functions

The coverage functions are required to determine on which cases the predictive models can be used with sufficient confidence, or not. In order to assess the accuracy of the predictive models, an independent dataset from the one used to train the models is used. For all aircraft pairs in this dataset, all clearance times corresponding to all spacing constraints are computed using only the predictive strategy (trajectory models and buffer models).

The error rates regarding the target constraints are then computed on several subsets of this dataset. A subset is defined by the value of one or several features. If the target error rates are respected with enough confidence for all constraints on a subset, then this subset is considered as covered with regards to the feature/set of features of interest. A coverage function is then computed for each feature of interest.

3) Conservative strategy

When a flight or a pair is not covered, a fallback is needed. To do so, conservative models are defined for both predictive models and buffers.

For each of the ML trajectory and speed profile models (Rolling Time, Rolling Distance, Time to TAS and Altitude to Time) a corresponding conservative model is defined as the mean by aircraft type and surface runway headwind band. In the cases where a type is too rare to compute an average, we resort to the worst-case behaviour of the category.

Conservative buffers are computed as high quantiles of the buffer distributions observed on the training set. All conservative buffers are defined per leader type, follower category and surface headwind band.

4) Strategy selector

The strategy selector, as described in the general framework, is defined from the use of the coverage functions in a specific order to decide if for a given input pair of flights the predictive or conservative models should be used.

5) Validation & Verification

At this stage of the project development, the model was only tested based on a local independent dataset. However, similar approach as was done for TBS-ORD could be developed.

C. D-PWS-A

Dynamic Pairwise Wake Separation for Arrivals (D-PWS-A) [24] [40] aims at safely reduce, when possible, wake turbulence separation minima between consecutive arrivals on the final approach based on wake risk monitoring. Wake turbulence separations indeed ensure safety under all conditions, but have been shown to be over-conservative in some meteorological conditions which directly penalises runway performance.
The D-PWS-A solution uses ML algorithms to determine, from the aircraft wake behaviour from previous flights monitored through LiDAR technology and from meteorological information, the wake separation minimum reduction that can be safely applied between subsequent arriving aircraft.
The solution architecture of this final application of the use of ML in a safety-critical application was inspired by the framework presented in this paper. The various components of the framework will be described and the major differences between them will be highlighted.

1) Predictive strategy

In this application, the predictive and the buffer models are merged together in a single component. Two predictive models are defined: transport and decay. The transport model targets the time it takes for both the wakes to be at least n meters away from the runway centerline. The decay model targets what is the time that it takes for both the wakes to be below a certain intensity.
For each flight, a prediction is done using both the transport and the decay models and the smallest between the two predictions is taken as the final predicted wake separation minima.

2) Coverage functions

To assess if it is possible to use the predictive strategy, the ML support model is introduced. The support model will tell if the predictive model has been trained with enough data in similar wind conditions and therefore it is confident that the results are correct.

3) Conservative strategy

For cases where the predictive models cannot be used no reduction of separation is proposed, resorting to Distance-Based Separation (DBS) wake turbulence minima.

4) Strategy selector

Based on the support model decision the predictive strategy or the conservative strategy is used at prediction.

5) Validation & Verification

At this stage of the project development, the model was only tested based on a local independent dataset. However, similar approach as was done for TBS-ORD could be developed.

V. Discussion and Conclusion

As previously mentioned, the framework proposed in this paper, has the objective to be able to use AI in safety-critical applications with predefined levels of safety. It was tested and used for three different use cases using ML regression techniques but could be extended to other AI techniques.

However, this framework relies on some assumptions that are further discussed in this section. Thoughts for potential next steps are also suggested.

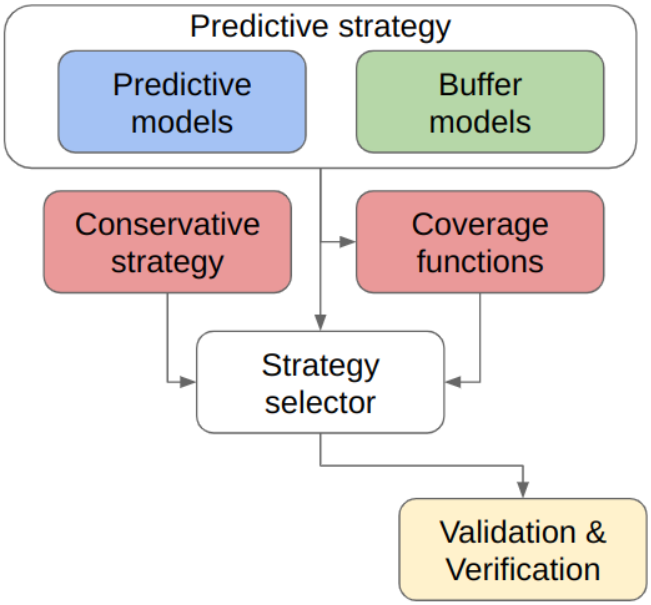

Figure 2. Overview of the proposed solution with in blue, green, red and yellow the components that require independent data.

The various components of the framework require data and some of them require independence between the used datasets. More specifically from Figure 2 it can be seen that a minimum of four independent data splits are required. One to be used by the predictive models training, a second one used by the buffer models training, a third one used to define the conservative strategy and the coverage functions, finally a fourth split used for the Validation & Verification. In practical situations, where not enough data are available, it could be possible to use a three-data split scenario, where the predictive models and the buffer models use the same split, as currently done in TBS-ORD and OSD. The advantage of a four-data split scenario is that should there be any overfitting of the predictive models this would be taken into account by learning the buffer models on a different data split. Otherwise, in the three-data split scenario, this last issue would be taken into account by the coverage functions, but with the disadvantage that any overfitting of the predictive models would be somehow passed on to the buffer models and then detected by the coverage functions resulting in an overall lower coverage.
Still focusing on the predictive and buffer models, it is recommended to keep these two components separated also for explainability reasons. Practically they could be merged in one component resulting in only the computation of the required buffers. The advantage of having two separate components gives a better view on what is calculated as an expected (average) behaviour (predictive models) and what is responsible for the variance in the process (buffer models). Predictive models are in charge of fitting the average behaviour of the indicator, while the buffers are somehow a variance

estimator. This setup then allows an intermediate point of control.

Moving forward to the coverage functions and the strategy selector components, it is important to notice how the design and decision on what features should be used currently requires the need of expert advice.

It is also worth noting that in the strategy selector component, currently there is a strong assumption of independence between features that is usually not realistic. As a simplified example, a dataset with a single airline that uses two different aircraft types (A and B) is here considered, with one aircraft type (A) much more frequent than the other (B). Two coverage functions are assumed to be defined, one for the airline and one for the aircraft type. Assuming a very low error rate for the aircraft type A but a very high error rate for the aircraft type B, this could potentially lead to the non-coverage of the airline overall, whilst aircraft A from the airline is in fact covered. This situation could be avoided by having only one coverage function that takes into account both the airline and the aircraft type features. However, the use of combined features requires more data for the coverage function computation since they will be partitioned more, potentially reducing the overall coverage due to lack of a sufficient number of samples per partition.

It is also important to note that the proposed framework in this paper does not intend to cover the implementation and deployment of the solution for the use by the final user (e.g., in an Air Traffic Control support tool). With this in mind, no components are dedicated to live monitoring of the solution in operations or for example on how online learning could be used.

In the future, use of AI will continue to expand even further in ATM and more generally in safety-critical applications. This makes the authors strongly believe that frameworks, as the one presented in this paper, will be a must have in order to safely deploy and authorize AI in operations.

The operational solutions cited in this paper have received funding from the SESAR Joint Undertaking under the European Union's Horizon 2020 research and innovation program under grant agreement No 874477.

For more information about the SESAR references provided in this paper please contact Catherine Chalon Morgan at catherine.chalon-morgan@eurocontrol.int .

REFERENCES

[1] CBC, "Uber self-driving car involved in 2018 fatal crash had software flaws, U.S. agency says," 6 November 2019 https://www.cbc.ca/news/business/uber-self-driving-car-2018-fatal-crash-software-flaws-1.5349581.

[2] EASA, "EASA concept paper: first usable guidance for level 1 machine learning applications," December 2021. https://www.easa.europa.eu/en/downloads/134357/en.

[3] EASA, "Machine learning application approval," 2022. https://www.easa.europa.eu/en/research-projects/machine-learning-application-approval.

[4] EUROCONTROL, "Why artificial intelligence is highly relevant to air traffic control," 29 November 2019. https://www.eurocontrol.int/article/why-artificial-intelligence-highly-relevant-air-traffic-control.

[5] EUROCONTROL, "Passenger demand support service - Beta testing," 2022. https://www.eurocontrol.int/dashboard/passenger-demand-support- service.

[6] D. Martinez, S. Belkoura, S. Cristobal, Wachter, F. Herrema and P. Wachter, "A boosted tree framework for runway occupancy and exit prediction," in 8th SESAR Innovation Days, Salzburg, Austria, 2018.

[7] Z. J. Lim, S. K. Goh, I. Dhief and S. Alam, "Causal effects of landing parameters on runway occupancy time using causal machine learning models," in Symposium Series on Computational Intelligence (SSCI), Canberra, ACT, Australia, 2020.

[8] Stempfel G.; Brossard V.; Bonnefoy A.; Ellejmi M.; Treve V. & De Visscher I.; Applying Machine Learning Modeling to Enhance Runway Throughput at A Big European Airport, in Proc. 10th EASN Virtual International Conference on Innovation in Aviation & Space to the Satisfaction of the European Citizens, IOP Conf. Ser.: Mater. Sci. Eng.1024 012106, 2021

[9] A. Heffar, R. Dalmau and E. Allard, "Prediction of flight departure and arrival routes with gradient boosted decision trees," in 11th SESAR Innovation Days, 2021.

[10] M. Enriquez, "Identifying temporally persistent flows in the terminal airspace via spectral clustering," in 10th USA/Europe Air Traffic Management Research and Development Seminar, Chicago, IL, 2013.

[11] M. C. RochaMurça, R. J. Hansman, LishuaiLi and P. Ren, "Flight trajectory data analytics for characterization of air traffic flows: A comparative analysis of terminal area operations between New York, Hong Kong and Sao Paulo," Transportation Research Part C: Emerging Technologies, vol. 97, pp. 324-347, December 2018.

[12] X. Olive and J. Morio, "Trajectory clustering of air traffic flows around airports," Aerospace Science and Technology, vol. 84, pp. 776-781, January 2019.

[13] S. J. Corrado, T. G. Puranik, O. J. Pinon and D. N. Mavris, "Trajectory clustering within the terminal airspace utilizing a weighted distance function," in 8th OpenSky Symposium, 2020.

[14] X. Olive, L. Basora, B. Viry and R. Alligier, "Deep trajectory clustering with autoencoders," in 9th International Conference for Research in Air Transportation (ICRAT), Tampa, FL, 2020.

[15] R. Marcos, O. Garcia-Cantu and R. Herranz, "A machine learning approach to air traffic route choice modelling," in 8th SESAR Innovation Days, Salzburg, Austria, 2018.

[16] Q. Duong, T. Tran, D.-T. Pham and A. Mai, "A simplified framework for air route clustering based on ads-b data," in International Conference on Computing and Communication Technologies, 2019.

[17] H. Naessens, T. Philip, M. Piatek, K. Schippers and R. Parys, Predicting flight routes with a deep neural network in the operational air traffic flow and capacity management system, E. M. U. A. C. Centre, Ed., Maastricht Airport, 2017.

[18] Y. Liu and M. Hansen, Predicting aircraft trajectories: A deep generative convolutional recurrent neural networks approach, 2018.

[19] M. C. R. Murca and M. d. Oliveira, "A data-driven probabilistic trajectory model for predicting and simulating terminal airspace operations," in 39th Digital Avionics Systems Conference (DASC), 2020.

[20] I. De Visscher, G. Stempfel and F. &. T. V. Rooseleer, "Data mining and machine learning techniques supporting Time-Based Separation concept deployment," in 37th Digital Avionics Systems Conference (DASC), London, UK, 2018.

[21] EUROCONTROL, "EUROCONTROL calibration of optimised approach spacing tool (COAST) with use of machine learning models," April 2021. https://www.eurocontrol.int/publication/eurocontrol-coast-calibration-optimised-approach-spacing-tool-use-machine-learning.

[22] L. De Petris, I. De Visscher, G. Stempfel, A. Jacques, M. Saidi and C. Chalon Morgan, "Machine learning supporting enhanced optimized spacing delivery between consecutive departing aircraft," in 12th EASN International Conference on "Innovation in Aviation & Space for opening New Horizons", 2022.

[23] N. Baranov and B. Resnick, "Wake vortex detection by convolutional neural networks," in 11th SESAR Innovation Days, 2021.

[24] L. Frigerio, I. De Visscher, G. Stempfel, R. Barragan. Montes and C. Chalon Morgan, "Dynamic pairwise wake vortex seperations for arrivals using predicitive machine learning models," in 33rd Congress of the International Council of the Aeronautical Sciences, Stockholm, Sweden, 2022.

[25] A. Parodi, V. Mazzarella, M. Milelli, M. Lagasio, E. Realini, S. Federico and R. C. Torcasio, "A nowcasting model for severe weather events at airport spatial scale: the case study of Milano Malpensa," in 11th SESAR Innovation Days, 2021.

[26] C. V. Gallego, C. Xia, M. G. Martínez, D. P. Álvarez and F. J. P. Heras, "Increasing the detection performance of genuine separation minima infringements with XGBOOST," in 11th SESAR Innovation Days, 2021.

[27] L. Caranti, M. Ribeiro, J. Ellerbroek and J. Hoekstra, "Safety optimization of a layered airspace structure with supervised learning," in 11th SESAR Innovation Days, 2021.

[28] I. Buselli, L. Oneto, C. Dambra, C. V. Gallego and M. G. Martinez, "Natural language processing and data-driven methods for aviation safety and resilience: from extant knowledge to potential precursors," in 11th SESAR Innovation Days, 2021.

[29] M. Mateos, I. Martín, R. Alcolea, R. Herranz, O. G. Cantú-Ros and X. Prats, "Unveiling airline preferences for pre-tactical route forecast through machine learning," 2021.

[30] I. Dhief, S. Alam, C. C. Mean and N. Lilith, "A tree-based machine learning model for go-around detection and prediction," in 11th SESAR Innovation Days, 2021.

[31] I. Dhief, Z. J. Lim, S. K. Goh, D.-T. Pham, S. Alam and M. Schultz, "Speed control strategies for e-aman using holding detection-delay prediction model," in 10th SESAR Innovation Days, 2020.

[32] H. Helmke, S. Shetty, M. Kleinert, O. Ohneiser, H. Ehr, A. Prasad, P. Motlicek, A. Cerna and C. Windisch, "Measuring speech recognition and understanding performance in air traffic control domain beyond word error rates," in 11th SESAR Innovation Days, 2021.

[33] M. Kleinert, N. Venkatarathinam, H. Helmke, O. Ohneiser, M. Strake and T. Fingscheidt, "Easy adaptation of speech recognition to different air traffic control environments using the deepspeech engine," in 11th SESAR Innovation Days, 2021.

[34] D4.6.002: SESAR 2020 PJ02-W2-14.6a & PJ02-W2-14.6b SPR-INTERP/OSED. D for V2. Part I. Ed. 00.01.00

[35] EUROCONTROL, Time-Based Separation (TBS) Principles as Alternative to Static Distance-Based Separation for Final Approach, Safety Case report, Ed 1.1, 2020

[36] H. Zouand and T. Hastie, “Regularization and variable selection via the Elastic Net” in Journal of the Royal Statistical Society. Series B (Statistical Methodology)”, vol.67, no. 2, pp.301-320, 2005

[37] J.H. Friedman, “Greedy function approximation: a gradient boosting machine” in Annals of statistics, pp. 1189-1232, 2001

[38] V. Cappellazzo, V. Treve and J. &. D. V. I. Toussaint, A Dynamic Departure Indicator Tool Allowing Optimised Spacing Delivery, Rome, Italy: XXV International Congress of the Italian Association of Aeronautics and Astronautics (AIDAA 2019), 2019

[39] SESAR OSED: D4.8.002. SESAR 2020 Wave 2 PJ02-W2-14.8 SPR-INTEROP/OSED for V2. Part I. Ed. 00.02.00

[40] SESAR: D4.7.002. PJ02-W2-14.7 SPR-INTEROP/OSED for V2 Part I. Ed. 00.02.00