[EN-A-053] Design of a software environment to support machine learning analysis of aircraft trajectories

(EIWAC 2017)

[+]C. Morales*, J. Sanz**, S. Moral***

* NATO Eurofighter and Tornado Management Agency (Spanish Air Force secondee), Munich, Germany
cmorram@correo.ugr.es

** A400M Mission Planning & Restitution System Development Team, Airbus Defence & Space, Getafe, Spain
jaime.sanz.external@airbus.com

*** Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain
smc@decsai.ugr.es

Abstract: This paper describes a software environment designed to import and/or simulate aircraft trajectory data, in combination with aeronautical information from multiple sources. Several interfaces and functionalities are being implemented to enable interoperability with machine learning and data mining tools. Potential applications of the system include research on ATM and airspace optimization, test and validation of algorithms related with aviation engineering, operational criteria or other related fields where the analysis of aircraft trajectories may benefit from data science. The software environment aims to promote the use of SWIM standards, exploiting their advantages in terms of information availability, data robustness and synergies with advanced computing software tools. This is achieved through the use of trajectory, weather, airspaces and NOTAM data from several providers within the tool, expecting to prove how SWIM is shaping the future of aeronautical information management. The paper also describes the continuous effort dedicated to performing an analysis of the best applicable alternatives in different fields like simulation, flight planning, machine learning applicability, cloud computing and big data, in order to enhance the interoperability and the availability of information.

Keywords: SWIM, aircraft trajectories, flight simulation, machine learning, cloud computing

1. INTRODUCTION

1.1 Background

This paper is part of a research about the measurement of pilot situational awareness (SA) using machine learning (ML) tools to study the relationship between individual human factors and aeronautical parameters, including aircraft trajectories, pilot control actions and especially the management of information in the cockpit during the flight, particularly through the use of electronic flight bags (EFB).

As presented in previous papers [1], we have implemented a simulation environment where flight and aeronautical information compliant with System Wide Information Management (SWIM) standards is used to perform calculations and generate variables that will be used to train the ML algorithms. In previous research stages a special effort was dedicated to the discretization and linear regression of the variables learnt by dynamic Bayesian networks.

1.2 Motivation

The software tool was initially designed for a very specific purpose related with human factors, but it was soon noticed that the nature of the data collected in the datasets and the developed ML algorithms had a potential to be applied to applications designed for different purposes within the aerospace context. The role of SWIM as an enabler of interoperability with external computation software tools can be applied to other researches and even to commercial applications.

We find the results related with the discretization of flight parameters particularly useful for applications where the expert assessment or human validation is difficult to include, due to the high dependency on the characteristics of each flight. On the other hand, the increasing utilization of cloud computing, either for small datasets or big data, and the progress in data visualization and web-based dashboard applications, led us to develop the current toolbox, expecting it to prove useful for other aviation professionals.

1.3 Contents of this paper

Section 2 contains an overview of the software environment that will be presented during the EIWAC 2017 workshop, including a brief description of the database structure and the user functionality. Special attention is dedicated to the interfaces with the flight simulator, storage in cloud-based repositories and integration with computing tools, both running locally and in the cloud. We have also created the website *www.trajectorymaster.aero* to provide a sample of the environment's functionality.

Assuming that the EIWAC audience is not necessarily familiar with ML, section 3 provides a very short introduction to the general concepts of this discipline and some specific descriptions of how the aeronautical data variables are created, modelled and manipulated in the environment in order to apply ML processes.

Section 4 describes the role of aeronautical data and interoperability with data providers in the software environment, with special attention to SWIM, as a key enabler to obtain relevant and accurate data. Section 5 focuses on the simulation functionality and the technology used by the environment to perform simulations that produce realistic trajectory data, based on route, performance and operational criteria in accordance to real planning standards.

Finally section 7 describes the implementation and the ongoing activities related to one of the novelties of the tool: the incorporation of cloud computing and big data.

2. OVERVIEW OF THE ENVIRONMENT

2.1 User interface and target users

The software environment described in this paper is designed to be used by aviation professionals with an engineering or data scientist profile. An effort has been made to provide a friendly and intuitive appearance to the application, combining maps, graphs and visualization of raw data either using tables or directly in xml code.

The user should be familiar with the database structure (which will be documented) in order to be able to use the data to perform queries, calculations and to store the resulting variables. Operations like the definition of points of interest, calculation of vertical and horizontal distances or time estimations, etc. are included. Computer programming skills are not required to use the tool, but since the it contains an interface to enable the use of external software packages, they will be very useful for instance, to create *Java* classes for data manipulation, or to implement additional dashboards and data visualization assets using tools like the *R Shiny* package, for which specific support is provided.

2.2 Databases and interface with external tools & data

Some operations and calculations can be performed directly using the application and an integrated navigation database with World global coverage. However, an important asset of this environment is based on the possibility to perform calculations using external data, including SWIM providers, as described in section 4.

In relation with database management, the software environment is designed to allow data exchange with external applications, including allowing them to read and modify the databases. Therefore, any tool having access to the local or cloud databases can perform calculations as desired. This is the case of software tools like *R*, *Matlab*, *WEKA* and *Python*, being *R* our currently preferred option due to the existence of a very wide support in the ML community and the existence of multiple specialized packages, as described in [2].

An overview of the software environment's components is shown in figure 1. Details about cloud-based database and cloud computing are included in sections 6 and 7.

2.3 Simulator interface

Another important interface present in the software environment is the connection with a flight simulator. In the previous application described in [1], the selected simulation software was *Microsoft FSX*. Although the current environment maintains compatibility with the former simulator connection, the current version of the environment has only been performed with *Lockheed Martin Prepar3D v3*. It should be noted that the simulator interface is not available in the website mentioned in section 1.

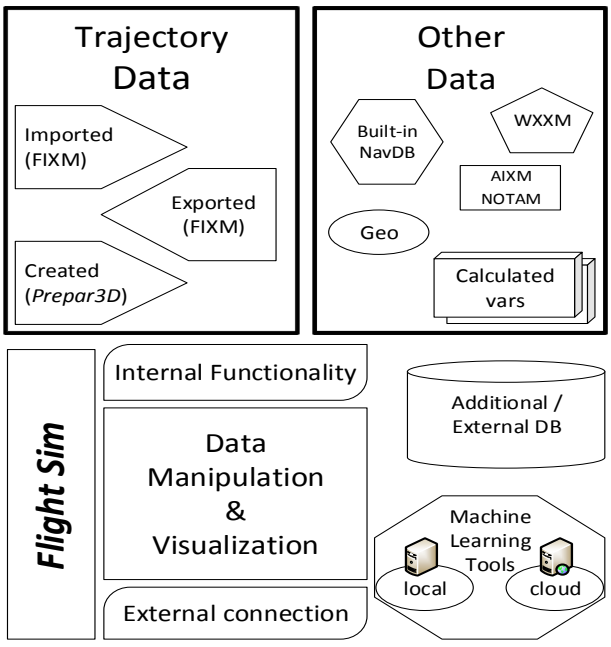

Figure 1 Components of the software environment

3. MACHINE LEARNING

This sections contains a very compressed introduction to the processes related with ML, in order to provide a better context to the readers not familiar with this discipline. The processes described are based in the generic model shown in figure 2. Each one of the following subsections includes a short definition of the process from the point of view of our approach in previous stages of our research, when the goal was to calculate the pilot SA.

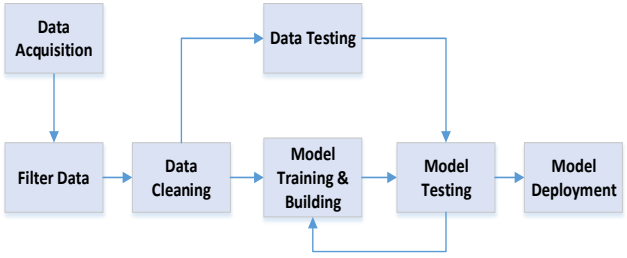

Figure 2 Generic process of Machine Learning

3.1 Data acquisition

Apart from the data included in the integrated navigation database and geographical elements, the most important data are acquired when the trajectories are loaded into the system. Data acquisition also includes the load of additional aeronautical data (NOTAM, METAR, etc.).

3.2 Filter data

This step consists in using the original data to define new variables that might have a high informative impact for the posterior inferences in our model. This is done with the help of experts which a knowledge on the relevant data used for decision making. A very simple example can be to define the distance from the current position to the expected trajectory.

3.3 Data cleaning

Data often contain noise or incorrect measures, and this step consists in detecting outliers or non-usual data.

It also consists in missing data imputation (for training models assuming a full observation of data) and variable discretization (for models only defined for discrete variables). There are ML-related processes which can be used for this task, as explained in [3].

3.4 Create Test Datasets

This is an important step to validate and refine the created models. In our case it will include normal flights of the same type used on the learning task as well as unusual flights in which the pilot is not responding appropriately to the current situation.

3.5 Model Training and Building

In previous stages of our research we used temporal Bayesian networks, a model which is especially appropriate to represent complex relationships between a set of variables which are measured in different timestamps [1]. In the case of continuous variables it is based on the use of linear regression to predict any variable using the variables from previous time, but allowing that the residuals are also estimated by using a discretization procedure and estimating them using other variables also from previous step.

3.6 Model Testing

In our model, there is a variable representing the pilot SA. The model allows to compute on-line a probability of having a good SA based on the pilot actions and state of flight observations. We have to test the model by checking that the probabilities of good SA are high in normal flights and low when the pilot is not responding appropriately.

The current implementation has been designed to allow a flexible implementation of this process using the external interfaces.

3.7 Model Deployment

Once the model is created, it will be integrated in the tool. The associated calculations are polynomial and can be obtained in computing devices of intermediate power, but it will also be possible to collect the data and send them to a server so that computations can be done in the cloud. The idea is that this is done in an automatic way without pilot intervention. The result will be a numerical value which can be communicate to the pilot in the form of a colored button which is *more red* when SA is bad and *greener* when it is good.

In general this process falls out of the scope of the paper's scope.

4. SWIM AND AERONAUTICAL DATA

4.1 Relevance of SWIM

The concept of SWIM covers a complete change in how ATM information is managed, improving digital data distribution and accessibility in terms of quality of the data exchanged. SWIM contributes to achieving the following benefits:

- Improved decision making to all stakeholders during all phases of flight (pre-flight, in-flight and post-flight) through:
  - Improved shared situational awareness; and
  - Improved availability of quality data and information from authorized sources.
- Increased system performance.
- More flexible and cost-effective communications by the application of common standards for information exchange.

- Loose coupling which minimizes the impact of changes between information producers and consumers.
- Support of ATM Service Delivery Management.

4.2 Use of SWIM platforms and data management

One of the main operational goals of our software environment is the possibility to request and visualize information in the following SWIM standards:

- Flight trajectory data in FIXM standard.
- Weather data in WXXM standard.
- Aeronautical data in AIXM standard.

The tool includes an interface with the *Laminar Data Platform*, developed by the *Snowflake* Company, which merges multiple ATM data sources into a single SWIM compliant service that includes a well-documented application program interface (API) available in their website [4]. The use of this interface allows users to retrieve and visualize the following ATM information:

- Flight Data: retrieval of real time route trajectories by aerodrome pair, by a specific area of Interest or by GUFI (Aircraft FPLN identifier).
- Weather Data: retrieval of Aerodrome METAR, TAF and enroute SIGMET information in real time.
- NOTAM Data: Retrieval of aerodrome and enroute NOTAM information by FIR or ICAO code.
- Aeronautical Data: Retrieval of airspaces, navaids, waypoints, aerodrome and regulation data.

This interface also provides the Trajectory Master website with a high SWIM compatibility enabling the user to retrieve and load real time aircraft trajectories between airport pairs, and also to query any desired operational information that may affect the flight (Weather, NOTAM, Airspace regulations etc.). The tool also provides limited support for loading FIXM trajectories in xml files from other sources.

It should be noted that at the time of writing this paper, the *Laminar Data Platform* still does not provide a worldwide coverage, since it is limited to the European and North American areas. It would be desirable to be able to access additional SWIM providers, and this is a challenge for future stages of the development. For instance, the possibility to register in the *Eurocontrol Network Manager* service has been assessed, but the applicable elegibility criteria in principle do not cover research applications. This topic could be discussed in forums like the EIWAC workshop, as they also play a relevant role in the future adoption of SWIM as the most relevant global standard and eventually facilitates its implementation in operational systems.

5. TRAJECTORY SIMULATION

5.1 Description of the requirement

The software environment can load aircraft trajectories directly from the sources described in section 4. This functionality has been implemented mainly to import real trajectories from different sources, although trajectory files generated with other methods and compliant with the supported formats can also be loaded as a trajectory (e.g. csv files from the site FlightRadar24.com), allowing users to import additional data into the databases and to represent them in the user interface.

However, the described trajectory formats have important limitations: the trajectories available during the tool's implementation were limited to very few parameters; basically the aircraft position (timestamp and coordinates) with a low sample rate, and optionally speed (from surveillance systems), altitude and heading. From our experience, research applications usually require high sample rates and as many relevant variables as feasible to provide applicable results.

When the use of real aircraft data is mandatory to perform the research, they can be obtained from flight data recorders. This scenario was successfully implemented in an earlier stage of our research, but it has the disadvantages that the implementation is strongly platform-dependant and in many occasions the results cannot be published due to information property and security reasons. It has therefore been decided to opt for the production of flight trajectories based on the commercial flight simulator *Prepar3D*, as mentioned in section 2.3. The main advantage is that there is a very wide scope of simulated parameters that can be tuned to model the scenario, and it is possible to record a very wide scope of variables, depending on the research / application aviation context (crew training, engineering, aircraft performance estimation, etc.). Reference [5] contains a list of these parameters.

5.2 Simulation methods

Aside from using the flight simulator manually to produce the simulation, we have covered the following two methods to simulate a flight:

5.2.1 Based on a trajectory file

Thanks to the use of a data connection between the main application and the flight simulator, we have automated the production of flight simulator control commands. The aircraft can then follow the trajectory of a FIXM or text file, based on the continuous configuration of its autopilot (or flight director) with the commands to fly to the coordinates at the speed, altitude and heading as stored in the file. This method allows a very high level of accuracy of the simulated trajectory with respect to the actual recorded trajectory that is being intended to simulate,

allowing to obtain simulated trajectories that are very similar to the real / original ones, and contain very detailed data, not only of the trajectory, but also from environment, vehicle and control parameters.

5.2.2 Based on a flight plan

Sometimes the goal of a simulation is not to reproduce a real trajectory because the priorities are set on the navigation procedures, databases or other related criteria. To achieve this and in order to be able to perform the simulations in the most automated way as possible using *Prepar3D*, a survey of commercial simulation software was performed, including the testing of virtual aircrafts and their avionics, navigation databases, mission planners and EFBs. With the current implementation of the tool, the user is able to use third party flight performance calculation tools (e.g. *PFPX*) independently from our software environment, plan the flight according to the desired criteria, which may include aircraft performance, operational criteria, training requirements, etc. The planned flights can be converted into a data package, imported into the simulator. The aircraft's flight director can fly the mission almost without human intervention, while our tool stores the selected parameters into the databases.

It is also planned to implement a simulation environment recording & reproduction interface, based on the simulator status parameters available in the simulation control interface (see [5]). This could be particularly interesting in order to simulate alternative trajectories of an aircraft. In the particular case of UAVs, this functionality is particularly useful in order to refine the programming criteria as a function of terrain elevation, obstacles, weather conditions and aerospace restrictions. This is a potential application of our tool.

6. CLOUD COMPUTING & BIG DATA

6.1 Cloud Computing with Amazon Web Services

The use of cloud computing resources like *Amazon Web Services* (*AWS*), *Google Cloud Platform* or *Microsoft Azure* is growing because of multiple reasons that include a reduced cost of ownership and suitability for mobile devices. This paper contains a proof of concept of a virtual machine based on *AWS* that runs machine learning algorithms and provides a fit for purpose user interface and development kit. The tool is interoperable with the rest of the software environment, including the user interface and the databases.

Security concerns apart, cloud computing offers promising possibilities to aviation, enabling the use of tools that require demanding processing power, with the additional advantage of being able to provide the services online, which in some cases may be necessary when using mobile devices, like EFBs. In order to achieve this a virtual machine running *R* was set up using the *Amazon Elastic Compute Cloud* (*EC2*) service. The *EC2* dashboard allows the installation of *Amazon Machine Images* (*AMI*) with customizable hardware profiles. There is a catalogue of *AMI* instances easily installable, apart from additional ones provided by third parties. These *AMI* include installations of the most popular operating systems and programming languages development kits. A *Matlab AMI* was discarded due to the existence of reported issues, but we will remain observant in case the issues are solved and the software proves useful. It was finally decided to install an *Ubuntu Linux* virtual machine that contains *R-Studio* and *Python*.

After the virtual machine is installed, extra packages and configurations can be added and saved, or loaded directly from external applications using *Secure Shell* (SSH) commands. The goal is to allow users to create and/or manipulate control dashboards based on the K package *Shiny* to implement their desired functionality.

The data, including the navigation databases and the trajectories described in previous sections, can be shared using client-server connections (we are currently using PHP). *AWS* also offers the *Amazon Relational Database Service* (*Amazon RDS*) that has been used to create a cloud-based MySQL database accessible from the main tool.

6.2 Big data demonstrator using *Hadoop*

Although the amount of data currently handled by the application does not fall into a category that could be treated as big data, it has been considered beneficial to set up an *Apache Hadoop* cluster to provide a proof of concept demonstrator and to gain experience about the relationship with other implementation options of the general environment and to help preventing future incompatibilities during the development of the current environment.

It is worth to mention that the *Hadoop Universe* includes a big amount of tools and techniques. A subset of this *Universe* are the database managers, data flow platforms, distributed programming, file systems, cluster managers, etc. that are shown in figure 3 and that constitute the first step of the big data implementation within our project, as the basic components of a big data cluster. Additional components of big data implementations, like non-tabular data or noSQL databases will be considered during further stages of the research.

To finalize this section in a more practical manner, we have included a tutorial created by a researcher on the data science field [6] that describes the necessary configuration steps to set up a *Hadoop* node using an *AWS EC2 AMI*. It contains an approximation to the technologies involved in the interface between both environments.

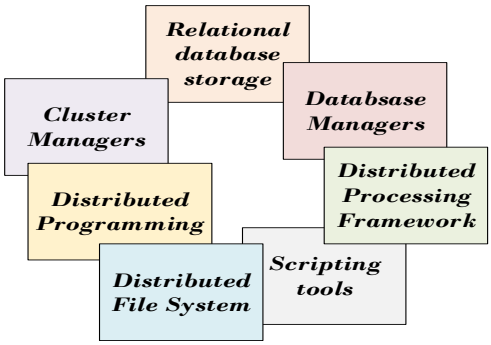

Figure 3 Basic components of a big data cluster

7. CONCLUSIONS AND FUTURE WORK

With this paper we have provided a description of the software environment that is being implemented to perform machine learning with aircraft trajectory data, as well as the requirements and design criteria, from both aeronautical and computing points of view. This is an ongoing initiative that is expected to evolve in the future depending on the feedback received from other collaborators in this research area and also potential forthcoming applications. Improvements may include implementation of modules to support interface with additional programming languages like *Python* or increase the interoperability with *Matlab*, which is currently quite limited. Another important area of improvement is related to big data: the *Hadoop* implementation is still in a preliminary stage, since the volume of data that has been identified does not meet the standard criteria of big data cluster design.

8. ACKNOWLEDGMENTS

The authors would like to thank Xiaodong Lu and Tadashi Koga from ENRI, for their ideas and support, as well as Maj. Roger Bou from the Spanish Air Force's "*Grupo de Escuelas de Matacán*", for his support helping us obtaining feedback from the Spanish UAS/RPAS Military School. This research was supported by the Spanish Ministry of Education and Science under project TIN2013-46638-C3-2-P and the European Regional Development Fund (FEDER).

9. REFERENCES

[1] Morales, C., Moral, S., "Regression Methods Applied to Flight Variables for Situational Awareness Estimation Using Dynamic Bayesian Networks", in Proceedings of Machine Learning Research, vol. 52, September 2016, pp. 356-367.

[2] James, G., Witten, D., Hastie, T., Tibshirani, R. "An Introduction to Statistical Learning with Applications in R". Springer (2013). p.104-126, p.144-167.

[3] García, S., Luengo, J., Herrera, F. (2015) "Data Preprocessing in Data Mining", Springer.

[4] Snowflake Software Ltd. (2017) "Laminar Data API Model". Retrieved from https://developer.laminardata.aero/documentation.

[5] Dowson, P. (2013) "FSUIPC4 Status of IPC Offsets for FSX". Retrieved from http://www.schiratti.com/dowson.html.

[6] Helei, H. (2017) "Hadoop: Setting up Hadoop 2.7.3 (single node) on AWS EC2 Ubuntu AMI". Retrieved from http://www.cs.cityu.edu.hk/~heleicui2/doc/Setup-Hadoop-2.7.3-(single-node)-on-AWS-EC2-Ubuntu-AMI.pdf.

10. COPYRIGHT