



Detection and identification of significant events in historical aircraft trajectory data

Xavier Olive^{*}, Luis Basora

ONERA DTIS, Université de Toulouse, 2 avenue Édouard Belin, 31055 Toulouse cedex 4, France



ARTICLE INFO

Keywords:

ADS-B
Trajectory data analytics
Air traffic flows
Anomaly detection
Air traffic management
Machine learning
Autoencoders

ABSTRACT

A large amount of data is produced every day by stakeholders of the Air Traffic Management (ATM) system, in particular airline operators, airports, and air navigation service providers (ANSP). Most data is kept private for many reasons, including commercial and security concerns. More than data, shared information is precious, as it leverages intelligent decision-making support tools designed to smoothen daily operations.

We present a framework to detect, identify and characterise anomalies in past aircraft trajectory data. It is based on an open source of ADS-B based aircraft trajectories, and extracted information can benefit a wide range of stakeholders: Air Traffic Control (ATC) training centres could play more realistic simulations; ANSP may improve capacity indicators; academics improve safety models and risk estimations; and commercial stakeholders, like airlines and airports, may use such information to improve short-term predictions and optimise their operations.

The technique is based on autoencoding artificial neural networks applied on flows of trajectories, which provide a useful reading grid associating cluster analysis with quantified level of abnormality. In particular, we find that the highest anomaly scores correspond to poor weather conditions, whereas anomalies with a lower score relate to ATC tactical actions.

1. Introduction

The identification of significant events in historical aircraft trajectory data falls within the scope of knowledge discovery and information extraction. It focuses on getting value out of the huge amount of data produced every day by stakeholders of the air traffic management (ATM) system. It is an arduous task as trajectories flown every day by commercial, military, private and other aircraft are subject to a large amount of impacting factors commonly modelled as *uncertainty* and causing deviation from the plan in terms of predicted times at given milestones along trajectories from gate to gate.

Sources of uncertainty in air traffic management are manifold. Weather-related uncertainties cause deviation from the plan due to a strong tailwind, deviations due to thunderstorm activities (also referred to as CB, for *cumulonimbus*, in the following) or disruptions when airport operations are interrupted. Air Traffic Control (ATC) staffing, late maintenance issues, airport operations including late boarding causing late startup time also introduce uncertainties in the whole aviation system.

Some of these sources of uncertainties can be predicted, quantified or logged but as the information produced by each stakeholder is not necessarily made available to all stakeholders, a systematic analysis of correlations between events and of their impact in the air traffic network is a considerable challenge.

* Corresponding author.

E-mail address: xavier.olive@onera.fr (X. Olive).

This contribution presents a successful Machine Learning (ML) approach for anomaly detection and identification of significant events in past trajectory data. The method builds on top of open sources of aircraft trajectory data, as received by a network of receivers of Automatic Dependent Surveillance–Broadcast (ADS-B), stored on and requested from The OpenSky Network ([Schäfer et al., 2014](#)). Our approach is based on trajectory data only, although external sources of not necessarily open data were used to help assessing and interpreting detected situations.

The proposed methodology relies on autoencoder neural networks, a widely used reconstruction-based anomaly detection technique. The main contribution of the paper is a framework to relate the trajectory anomaly scores computed by autoencoders to significant operational events such as re-routings or deconfliction measures. Detected significant events on a wide range of study cases include strong deviations (weather disruptions, regulations) and ATC actions, such as deconfliction and sequencing actions. The proposed method complements well with clustering techniques to identify air traffic flows and highlights trajectories within flows as presenting a degree of dissimilarity important enough to justify further analysis.

Detecting and identifying significant events is a valuable asset to address many applications, including safety analyses, preparation of ATC training simulations with realistic situations and feeding collaborative decision making (CDM) tools with operational information coming from the observation of past data in order to improve their acceptability. In particular, identifying traffic situations which could have led to a separation loss, analysing context and proximate events are valuable inputs for collision risk models (CRM) ([Garcia Gonzalez, 2013](#)), resulting in estimations of mid-air collision risks. Locating hot spots based on identified situations could also help assess the complexity of traffic inside a given sector when it is in operation.

The evaluation of our methodology is performed on several months of data at various scales: first with a focus on routes between city pairs, i.e. trajectories flying from city A to city B; then looking from a sector point of view, coupled with periods when the sector was operationally deployed according to the Sector Configuration Plans (SCP). Finally, we have a closer look at Terminal Manoeuvring Areas (TMA) and have a look at internal layers of autoencoder neural networks to understand what kind of representation is learned before being able to detect significant events.

The paper is organised as follows. Section 2 reviews the state of the art with respect to trajectory clustering, anomaly detection and their application to safety analyses. Section 3 presents the formalism, based on an extensive use of a special kind of artificial neural networks called autoencoders. Section 4 presents various use cases and analyses detected and labelled situations. Section 5 provides insight on the representation that is produced by our neural networks through the learning process.

2. Background and literature review

Detection and identification of significant events in large amounts of historical aircraft trajectory data is an active area of research. What a significant event means depends ultimately on the specific use case addressed. In general, it encompasses operational situations or pattern behaviours in air traffic which are worth identifying for online tracking or post-operational analysis because of their potential impact on the safety or efficiency of air traffic operations.

The methodology described in this paper builds on two previous research efforts ([Olive et al., 2018](#); [Olive and Basora, 2019a](#)) based on trajectory clustering and anomaly detection techniques. This paper focuses primarily on the application of autoencoder-based anomaly detection to identify significant events. However, trajectory clustering may be needed to compute the air traffic flows prior to the detection of events. Thus, the most relevant research on these two areas will be introduced here in the corresponding subsections.

Trajectory clustering is useful to identify the air traffic flows within an airspace. The major and recurrent flows correspond to the standard modes of operation in an airspace, such as Instrument Landing Approaches (ILS) procedures in TMA or the main ATS routes linking the defined entering and exiting points in an en-route sector. Once the standard modes of operation are determined, it is possible to detect in a more precise way (by using other techniques) specific anomalous traffic patterns ([Olive and Morio, 2019](#)). At the trajectory level, concrete examples of these significant events will be given as part of the literature review, but we are generally referring to flight trajectories or flight interactions within or among the main flows with a certain degree of unexpected behaviour. At the flow level, although it is not the focus of our approach, non-nominal flow configurations in a certain airspace at certain time could also be considered as examples of significant events.

2.1. Trajectory clustering and flow identification

Clustering is an unsupervised data analysis technique widely used to group similar entities into clusters according to a similarity, or a distance function. Multiple clustering algorithms exist in the literature to cluster point-based data such as k-means ([Lloyd, 1982](#)), DBSCAN ([Ester et al., 1996](#)) or HDBSCAN ([Campello et al., 2013](#)). When clustering is applied to trajectories, it requires a proper distance function to be defined between trajectory pairs, which is challenging because of the functional nature of trajectories. The most commonly used approach is to simply sample the trajectory so as to obtain a n -dimensional vector of points for the use of point-based clustering algorithms and distances such as the Euclidean one.

A significant number of clustering methods exist in the literature for flow identification, where the goal is to determine the set of clusters that best fit the operational air traffic flows within an airspace. These methods associate each cluster to an air traffic flow which is a traffic pattern with both temporal and spatial characteristics. The exact definition of what a flow is ultimately depends on the specific application context. Also, it is worth noticing that a majority of the research have focused primarily on studying the spatial dimension of flows.

Some flow identification methods are developed for analysis of air traffic operations in the terminal area. For instance, [Eckstein](#)

(2009) combines Principal Component Analysis (PCA) with k-means to evaluate the performance of individual flights in TMA procedures. Gariel et al. (2011) propose a framework based on PCA, DBSCAN and k-means for airspace monitoring and complexity assessment also in the TMA. Enriquez (2013) apply hierarchical and spectral clustering techniques to identify air traffic flow patterns from and to an airport. Murça et al. (2016) present a framework based on DBSCAN and other algorithms to identify and characterise air traffic flow patterns in New York transition/terminal airspace for daily assessment of the tactical air traffic operations.

Other research focus on flow identification within en-route airspace. Sabhnani et al. (2010) present a specific clustering algorithm to identify flow patterns and critical points for en-route airspace redesign. Basora et al. (2017) propose a framework for the analysis of air traffic flows based on trajectory clustering with the HDBSCAN algorithm (Basora et al., 2017).

The flow identification method applied in this paper for the use case concerning the en-route airspace (Basora et al., 2018; Olive and Basora, 2019a) (Section 4.4) is based on the DBSCAN algorithm and a progressive clustering technique originally developed by Andrienko et al. (2018), Rinzivillo et al. (2008). This method was especially designed for the identification of air traffic flows within an en-route sector and was validated with real operational data and operational expertise from Bordeaux Area Control Center (ACC) (Basora et al., 2018).

2.2. Anomaly detection

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behaviour (Chandola et al., 2009; Pimentel et al., 2014). Anomaly detection techniques can be used to detect significant events in flight data, as these usually correspond to unusual operational situations and so presenting a certain degree of anomaly. Go-around operations, runway excursions, conflict resolution manoeuvres and traffic rerouting are just a few examples of significant events that could be identified by applying anomaly detection methods.

A recent survey (Basora et al., 2019) reviews some of the main anomaly detection techniques and their application to aviation data. The review covers an exhaustive number of use cases and methods for the identification of significant events which is too large to be reproduced here again. Instead, we will focus only on the most relevant and recent research work in the field.

MKAD (Das et al., 2010) can be considered as one of the first methods designed to effectively detect operationally significant anomalies with heterogeneous sequences of both discrete and continuous variables. Based on kernel functions and OC-SVM, MKAD can identify operational situations in flight data such as anomalous flight approaches.

Approaches based on clustering algorithms have been widely used for anomaly detection. For instance, Li et al. (2011) propose a cluster-based anomaly detection method based on DBSCAN (ClusterAD) to detect anomalies in a airline dataset for 365 B777 take-off and approach operations. Following up this research, Li et al. (2015) present a method based also on DBSCAN called ClusterAD – Flight to identify abnormal flights during take-off or approach as a whole.

More recently, Li et al. develop ClusterAD – DataSample (Li et al., 2016) which is a Gaussian Mixture Model (GMM) based method capable of instantaneously detecting abnormal data samples during a flight rather than abnormal flights as a whole during a specific flight phase. Puranik and Mavris (2017) present a generic methodology based on DBSCAN and SVM for identifying anomalies from general-aviation in the approach and landing phase.

Functional principal component analysis (FPCA) is another technique that has been recently applied to identify atypical approaches and landings (Olive and Bieber, 2018; Jarry et al., 2018), sometimes combined with clustering techniques such as DBSCAN or HDBSCAN with GLOSS (Campello et al., 2015; Jarry et al., 2018).

Deshmukh and Hwang (2019) propose a temporal logic based anomaly detection algorithm (TempAD) applicable to trajectories in the terminal airspace. The algorithm, based on a temporal-logic learning approach (Kong et al., 2016), can learn human-readable mathematical expressions from data which facilitates the feedback and interaction with operational experts. The method uses DBSCAN as a preprocessing step to identify the clusters with similar trajectories on which the detection of anomalies with TempAD becomes more effective.

In spite of these remarkable contributions, the same recent survey (Basora et al., 2019) draws the attention towards a fact: the limited application of the advances in the area of deep learning and neural networks to the detection of operational events in air traffic operations. This is unfortunate as these approaches should be more adapted than traditional machine learning methods when processing large-scale high-dimensional flight data.

For instance, Janakiraman and Nielsen (2016) propose an anomaly detection approach based on a type of feed-forward neural networks (extreme learning machines) as an alternative to MKAD for the identification of safety risks in large aviation datasets.

As another effort in this direction, we study the use of a different neural network called autoencoder (Kramer, 1991), which is widely applied in the field of anomaly detection (Sakurada and Yairi, 2014; Zhou and Paffenroth, 2017) and whose application to air traffic management deserves to be further investigated. Although several variants of autoencoders exist, our methodology is based on the standard version introduced in Section 3.1.

As a reconstruction-based method, an autoencoder is trained to learn the most salient features in the training data where normal data is by definition predominant. Thus, it is assumed that the reconstruction error computed by an autoencoder will be lower for normal data than for the anomalous one, i.e. a reconstruction error can be considered as an anomaly score. Another interesting property of autoencoders is their capacity to project the input data to a lower-dimensional latent space.

We think these characteristics make of the autoencoders an excellent candidate technique for anomaly detection. As long as we know, our contribution is the first systematic attempt to exploit both the reconstruction loss and the latent space for the detection of a wide variety of significant events in air traffic management.

2.3. Significant events and safety analysis

In the air traffic operations context, the main value of an anomaly detection method is the capacity to automatically highlight, amongst the potentially large number of trajectories, a manageable amount of significant events. Then further risk assessment can be carried out by a safety expert focusing on these subset of events. On the other hand, the frequent occurrences of certain significant events like deconfliction measures within an area of an airspace can create hotspots leading to high complexity and potentially higher safety risk. Therefore, not only the identification of individual events but also their frequency and geographical density are important from a safety point of view.

An efficient method for identification of significant events from flight data can be beneficial for a collision risk model (CRM) to improve safety level estimation. For instance, [Garcia Gonzalez \(2013\)](#) develops a CRM for Eurocontrol to assess European en-route airspace safety risk. The detection of what they call proximate events (situations where the aircraft involved may evolve towards a collision due to a separation minima infringement) is performed based on either a track segmentation technique or the simulation of the traffic evolution via a time step and look-ahead time parameters. Our methodology could be useful in such context by providing an alternative mean of detecting changes in trajectories associated with aircraft manoeuvres or flight interactions corresponding to potential conflicts.

However, the major focus of the past research has been put on identifying significant events for an individual flight during a specific flight phase, especially during the approach, landing and take-off phases because of their inherent higher safety risks. For instance, MKAD ([Das et al., 2010](#)) can identify significant events during the approach phase such as go-around operations, unusually high airspeed flights at low altitude, flights impacted by gusty winds and abnormal approaches. ClusterAD ([Li et al., 2011](#)) can identify operational situations including high/low energy approaches, unusual pitch excursions, abnormal flap settings and high wind conditions. ClusterAD – DataSample ([Li et al., 2016](#)) is able to detect several cases of unstable approaches, strong tailwind and late localiser intercept, just to mention a few examples. Unfortunately, these approaches work with on-board recorded data whose access is restricted and needs authorisation from the airlines.

In comparison, our approach is more generally applicable to a wider range of operational contexts (TMA, en-route) and potentially at a more macroscopic scale involving traffic interactions in several air traffic flows. For instance, we intend to detect significant events associated with tactical ATC orders for conflict resolution in en-route or sequencing instructions in the TMA, as well as re-routings in en-route or deviations in the approach phase caused by bad weather. Finally, our approach relies on ADS-B data which is more easily available than airline data or radar tracks necessary to many of the methods in the literature.

3. Methodology

The purpose of our methodology is to provide a framework for the detection and identification of significant operational events in ADS-B trajectory data. This framework relies on the use of autoencoders (Section 3.1) as an anomaly detection technique to compute the reconstruction loss (or anomaly score) associated to each trajectory. By analysing the resulting distribution of reconstruction errors, it is possible to identify several kind of significant events. To be more precise, we show how different degrees of anomalies can be linked to specific operational events such as re-routings or deconfliction measures.

This methodology is particularly well adapted to the analysis of anomalies in the context of traffic flows, which can be defined for instance as a set of flights between a city-pair route or landing at a particular runway. If the flows in an airspace need to be computed from the trajectories, we propose to use one of the clustering techniques available in the literature (Section 2.1). We illustrate the use of a clustering method for flow identification ([Basora et al., 2018; Olive and Basora, 2019a](#)) in the airspace scenario (Section 4.4).

The flows provide the different modes of traffic operation from which deviations can be more easily detected by the autoencoder and analysed. However, we realised that an autoencoder only based on a standard loss function such as a mean square error (MSE) provided sub-optimal results for some of the use cases. This motivated the introduction of a specific regularisation term which facilitates the convergence of the autoencoder training process and should prevent overfitting as well (Section 3.3).

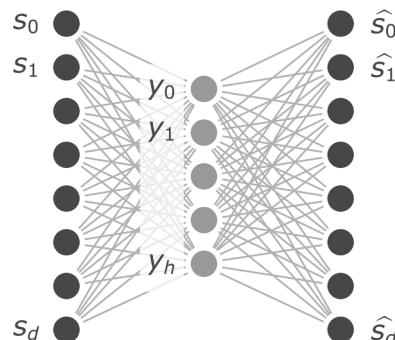


Fig. 1. Autoencoder neural network architecture with one layer.

3.1. Autoencoders

Autoencoders are artificial neural networks consisting of two stages: encoding and decoding. A single-layer autoencoder (Fig. 1) is a kind of neural network consisting of only one hidden layer. Autoencoders aim at finding a common feature basis from the input data. They reduce dimensionality by setting the number of extracted features to be less than the number of inputs. Autoencoder models are usually trained by backpropagation in an unsupervised manner. The underlying optimisation problem aims to minimise the distance between the reconstructed results and the original inputs.

The encoding function of an autoencoder (such as the one depicted in Fig. 1) maps the input data $s \in \mathbb{R}^d$ to a hidden representation $y \in \mathbb{R}^h = e(s) = g(w \cdot s + b)$ where $w \in \mathbb{R}^{d \times h}$ and $b \in \mathbb{R}^d$ are respectively the weight matrix and the bias vector and $g(\cdot)$ is a non linear activation function such as the sigmoid or hyperbolic tangent functions. The decoding function maps the hidden representation back to the original input space according to $\hat{s} = d(y) = g(w' \cdot y + b')$, $g(\cdot)$ being most of the time the same activation function.

The objective of the autoencoder model is to minimise the error of the reconstructed result:

$$(w, b, w', b') = \operatorname{argmin} \ell(s, d(e(s))) \quad (1)$$

where $\ell(u, v)$ is a loss function determined according to the input range, typically the mean squared error (MSE) loss:

$$\ell(u, v) = \frac{1}{n} \sum ||u_i - v_i||^2 \quad (2)$$

where u is the vector of observation, v the reconstructed vector and n the number of available samples indexed by variable i .

Depending on the volume and complexity of the considered data set, autoencoders' layers may be stacked (see Fig. 2) in an attempt to better grasp the structure of the underlying traffic. Other loss functions, such as cross-entropy loss, or custom loss functions with regularisation terms (see Section 3.3) may be designed for a better robustness and stability of the results across executions of the method.

3.2. Distribution of reconstruction errors and evidence of convergence

Autoencoders are trained to project, or compress, data onto a latent space of a smaller dimension, then to regenerate the original data in the original space based on a representation in a smaller dimension. The rationale behind the use of autoencoders for anomaly detection is that these artificial neural networks specialise on finding a representation of the data with few parameters, just enough to reconstruct the original data in most cases. They are trained to minimise a loss function, and learn to reconstruct the original data, at the risk of not being able to reconstruct anomalous samples.

Fig. 3 plots the evolution of the MSE loss together with the distribution of reconstruction errors across all the samples contained in a dataset. Details about the specificity of the dataset are not relevant at this point: we will introduce them in depth in Section 4.

After one iteration, i.e. with random weights on the edges of the neural network, reconstruction errors present an initial distribution which starts moving towards lower reconstruction errors after 10 or 100 iterations. After 1000 iterations, although the convergence is obviously not complete, we see that most samples distribute close to 0 and few samples reconstruct with higher residual errors.

Within the unsupervised learning paradigm, after the loss function converges, we have no way to check the accuracy of our network, the authors recommend to plot this distribution as a rule of thumb to verify the convergence of the training process and the relevance of the resulting autoencoder.

3.3. A new term of regularisation

In some of the datasets introduced in Section 4, we found that several runs of the training phase of our autoencoders would distribute along two variation modes. Fig. 4 displays such an example of distribution when variation modes of unbalanced weights

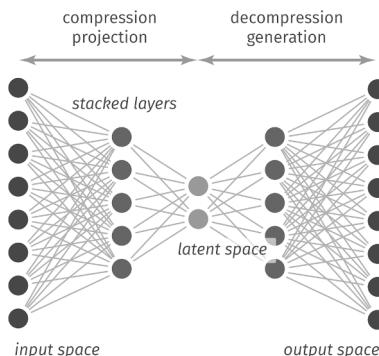


Fig. 2. Autoencoder neural network architecture with stacked layers.

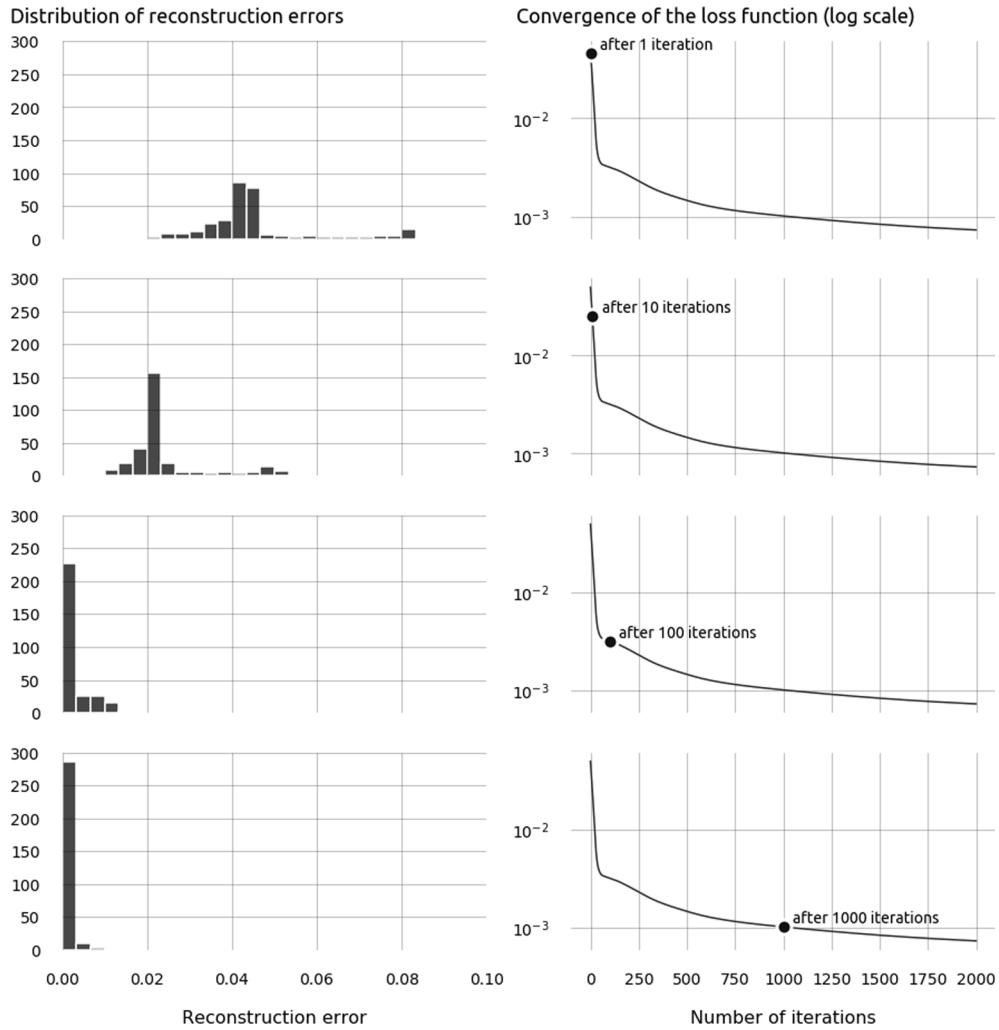


Fig. 3. Distribution of reconstruction errors during the training of the autoencoder. The sample dataset used to generate this figure is presented in Section 4.

exist in the original dataset: these modes may subsist in the distribution of reconstruction errors in the form of a distribution with two "hills".

In order to limit a premature optimisation of the autoencoder which would learn to favour one mode over the other (the one with the more samples) in some runs of our algorithm, we improved the training of our network with a regularisation term added to the loss $\ell(u, v)$.

A regularisation term is a penalty term added to the loss that is commonly used to prevent overfitting. Among neural networks, L1- (resp. L2-) regularisations penalise the loss after each iteration with the sum of the absolute (resp. squared) weights of the neural network. In our case, since we expect a distribution of reconstruction errors that would fit an exponential law, we propose a regularisation term based on a measure of distance between distributions.

After each iteration, we fit an exponential law to our distribution of reconstruction errors. The best fit to an exponential distribution can be written based on the mean of all $\rho_i = \|u_i - v_i\|^2$ samples, which have already been computed in form of the MSE loss $\ell(u, v)$. Therefore, the best fit for the probability density function becomes:

$$f : x \mapsto \frac{1}{\ell(u, v)} \cdot e^{-\frac{x}{\ell(u, v)}} \quad (3)$$

Then, we compute the distance between the distribution of reconstruction errors and the fitted exponential probability density function. For each $t_j \in [0, \max(\rho_i)]$ equally sampled with $j \in [1, m]$, we evaluate the difference:

$$\delta_j = \left(\frac{1}{n} \sum_i K_{\rho_i}(t_j) \right) - f(t_j) \quad (4)$$

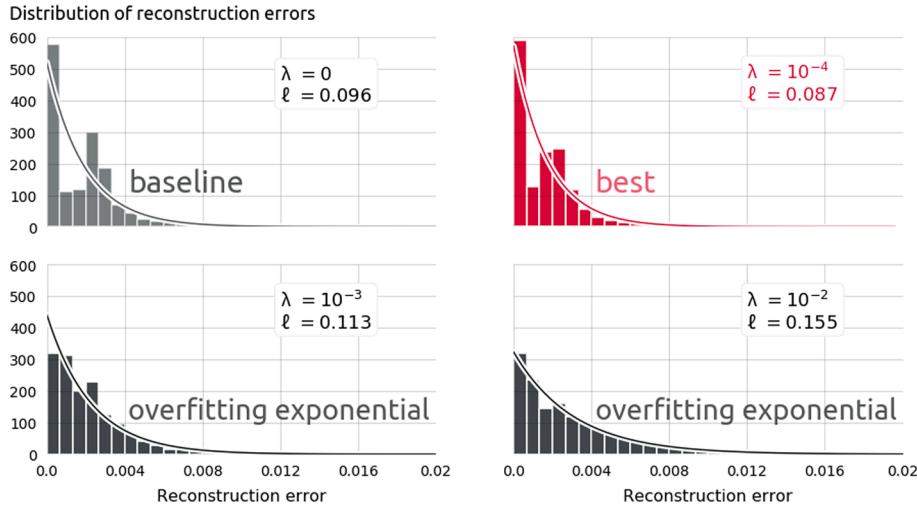


Fig. 4. Distribution of ρ_i , suggesting two modes of variations, and their fitted exponential law: the regularisation term aims at fitting the distribution under the curve while the MSE loss aims at pushing the bins toward zero. When the regularisation is prevalent (overfitting on the exponential distribution), all the bins fit under the curve but samples reconstruct poorly with no added value for anomaly detection.

with K_{ρ_i} a Gaussian Kernel function:

$$K_{\rho_i}(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2}\left(\frac{x-\rho_i}{\sigma}\right)^2} \quad (5)$$

Finally, we sum all the δ_j as a regularisation term to the original square loss. For the specific example of Fig. 4, with a bandwidth term $\sigma = 0.02$, we found $\lambda = 10^{-4}$ to be particularly efficient as it helped converging to a better MSE loss than the one obtained without regularisation.

$$\ell^*(u, v) = \ell(u, v) + \lambda \sum_{j=1}^m \delta_j \quad (6)$$

We performed several training runs with the same dataset and parameters. We observed that the regularisation term help balance bias and variance resulting in more stable and predictable results which hopefully should prevent model overfitting and help staying away from multimodal distributions.

The differentiation of the regularised loss $\ell^*(u, v)$, necessary to implement the gradient descent and backpropagation during the training period has been delegated to the autograd module of PyTorch (Paszke et al., 2019). All terms presented in this subsection can be written with `torch` functions which provide all that is needed for backpropagation.

4. Application to the identification and detection of significant events

4.1. Description of the use cases

We considered in this paper a series of particular use cases. The first proof of concept of this method has been made on a set of city pairs (Olive et al., 2018), before being extended to flows in an airspace (Olive and Basora, 2019a). In this paper, we present a novel use case in the Terminal Manoeuvring Area (TMA) around Zurich airport (LSZH) which we will explore further in Section 5.

Automatic Dependent Surveillance–Broadcast (ADS-B) is a cooperative surveillance technology which provides situational awareness in the air traffic management system. Aircraft determine their position via satellite, inertial and radio navigation and periodically emit it (roughly one sample per second) with other relevant parameters to ground stations and other equipped aircraft. Signals are broadcast at 1090 MHz: a decent ADS-B receiver antenna can receive messages from cruising aircraft located up to 400 km far away, while the range is much lower for aircraft flying in low altitude or on ground.

The data used for this study is collected by the OpenSky Network (Schäfer et al., 2014), a network of ADS-B receivers, which offer querying capabilities on their database for academics. Recorded data contains timestamps (added on the receiver side, with many receivers equipped with a GPS nanosecond precision clock), transponder unique 24-bit identifiers (icao24), space-filled 8 character callsigns, latitude, longitude (in degrees, 5 digit precision), (barometric) altitude (in ft, w.r.t. standard atmosphere), GPS altitude (in ft), ground speeds (in kts), true track angle (in degrees), vertical speed (in kts). Based on sensors located on the landing gear, different messages with similar information are sent when the aircraft is on the ground, resulting in a boolean flag in the OpenSky records.

Aircraft compute their position within uncertainty margins they are able to estimate and broadcast. These information are not

provided decoded in the OpenSky Network database but could be processed (Sun et al., 2019) from the raw messages on an as-needed basis. However, we kept the uncertainty analysis out of the scope of this paper and chose to manually filter irrelevant, corrupted, or incomplete data as part of the preprocessing step.

The *city-pair* data set consists of one full year of 3536 trajectories flying from Paris–Orly (LFPO) to Toulouse–Blagnac (LFBO) airports between January and December 2017. This data set has been requested based on a set of 28 callsigns commonly attributed to trajectories serving this route.

The *airspace* data set consists of seven months of trajectories flying through the LFBBPT airspace of the French Bordeaux Area Control Centre (ACC) between January 1st and August 6th 2017. The data set is limited to 14,461 trajectories crossing the airspace during the time intervals when the sector was operationally deployed according to the Sector Configurations Plans (SCP), also known as *opening schemes*. The goal of using the SCP is for the traffic under analysis to be representative of operational situations with a level of workload deemed acceptable by controllers.

The *landing* data set consists of 19,480 trajectories landing at Zurich airport (LSZH) between October 1st and November 30th 2019. We relied on The OpenSky Network (Schäfer et al., 2014) database to properly label trajectories landing at LSZH.

All datasets have been requested and preprocessed with the help of the Python `traffic` library (Olive, 2019) which downloads OpenSky data, converts the data to structures wrapping `pandas` data frame and provides a specialised semantics for aircraft trajectories (e.g., intersection, resampling, filtering, and more). In particular, it iterates over trajectories based on contiguous timestamps of data reports from a given `icao24` identifier: all trajectories are then assigned a unique identifier and resampled to one sample per second.

Fig. 5 shows a preview of the data contained in the `pandas` data frame underlying the trajectory structures. The data for the three use cases presented in this paper is now provided as generic imports in the `traffic` library, triggering a download from a corresponding figshare repository (Olive and Basora, 2019b) if the data is not in the cache directory of the user:

```
from traffic.data.datasets import (
    # -- The city pair use case --
    # -> 3536 trajectories between Paris--Orly to Toulouse--Blagnac (2017)
    paris_toulouse_2017,
    # -- The airspace use case --
    # -> 14,461 trajectories crossing the LFBBPT sector (1st semester 2017)
    airspace_bordeaux_2017,
    # -- The landing use case --
    # -> 19,480 trajectories landing at Zurich airport (Oct./Nov. 2019)
    landing_zurich_2019
)
```

4.2. City pairs

The provided dataset includes full trajectories from Paris–Orly to Toulouse–Blagnac airports. Trajectories from the OpenSky Network are subject to errors happening in different steps of the acquisition chain: errors before the emission of data (imprecision in the localisation, quantification artefacts) and errors in the receiving and decoding of the data by a pool of heterogeneous feeders. A first set of basic filtering and resampling methods provided in the `traffic` library has been applied to (partly) sanitise all data sets before publication.

Then, the second step consists in defining a bounding box containing the set of trajectories. We chose in this section to consider roughly drawn bounding boxes including most trajectories before they enter the TMA of LFBO airport (here the bottom-left and top-right coordinates of the learning box are resp. (44°17' N, 0°57' E) and (45°32' N, 2°45' E)). We also chose to include in the bounding box the first navigational beacon (i.e., a set of geographical coordinates) of the standard arrival (STAR) procedures.

The next step in the data preparation consists in resampling the trajectories in the selected bounding box. Since autoencoders have a fixed number of inputs, we resample each subset of trajectory cropped to our learning box so as to get only d equally distributed samples. **Fig. 6** plots a resampling with $d = 15$ for illustrations' sake but we chose a larger number of samples $d = 150$ for our experiments. This choice is arbitrary and we found that other values of d in the same order of magnitude have no significant impact on the results.

Features are chosen among all data provided in the ADS-B specifications: latitude, longitude, GPS and barometric altitude, track angle, ground speed, vertical speed. Different airspeeds (CAS, IAS, TAS, etc.) are sent by aircraft upon request of Secondary Surveillance Radars on DF 20 and 21 but as consistent request is not an universal process, we chose to keep these features out of our

	timestamp	altitude	callsign	geolatitude	groundspeed	icao24	latitude	longitude	squawk	track	vertical_rate	flight_id
0	2017-03-14 20:34:57+00:00	24725.0	AF526KB	25550.0	429.786785	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
1	2017-03-14 20:34:58+00:00	24750.0	AF526KB	25550.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
2	2017-03-14 20:34:59+00:00	24750.0	AF526KB	25550.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
3	2017-03-14 20:35:00+00:00	24800.0	AF526KB	25600.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117
4	2017-03-14 20:35:01+00:00	24825.0	AF526KB	25650.0	430.784957	392ae6	48.115631	2.063919	1000	183.460183	1408.0	AF526KB_4117

Fig. 5. Preview on the trajectory data associated to one trajectory.

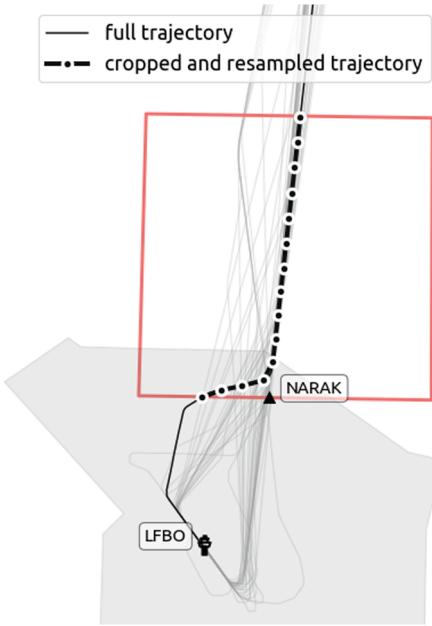


Fig. 6. Data preparation: each trajectory is cropped and resampled with 150 samples.

dataset for future work beyond the scope of this paper. Controller's actions are most often expressed in terms of altitude ("climb to flight level 310"), track angle ("turn left heading 210", "direct to NARAK"), and speed ("reduce speed to 160 kts"); since speeds are expressed in IAS and not in ground speed, we focused on track angles and altitude profiles.

For the sake of clarity, we focus here on an analysis on normalised track angles. The input dimension, i.e. the number of neurons on both input and output layers of our autoencoder, has been set to $d = 150$. The embedding dimension, i.e. the number of neurons on the hidden layer has been set to a lower value of 64. All neurons are defined with a sigmoid activation function. The loss function used is the mean squared error as defined in (2), which compares vectors of scaled track angles (through min–max normalisation) with their *reconstructions*, i.e. their images through the autoencoder.

As a result of our training process, we get a *reconstruction error*, i.e. a measure of the difference between a given trajectory and its autoencoded representation. Fig. 7 plots the distribution of these reconstruction errors after 2000 iterations. The model is trained to minimise the sum of all reconstruction errors, so the distribution is centred around zero.

The same distribution is plotted on a logarithmic y-axis to emphasise the few specific trajectories with higher reconstruction errors:

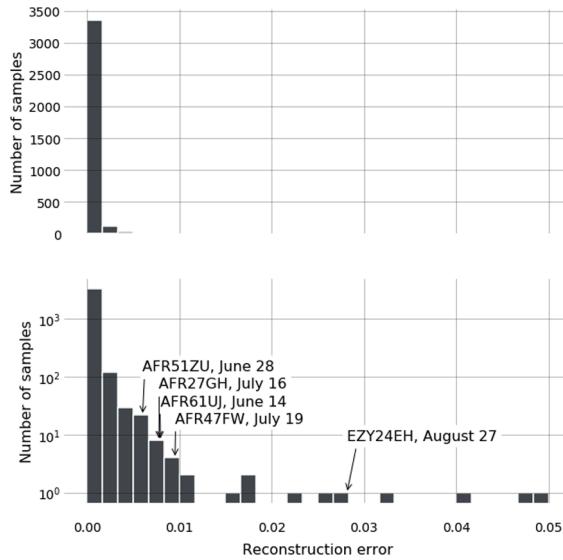


Fig. 7. Distribution of reconstruction errors on the Toulouse scenario. The bottom distribution replicates the top distribution on a semi-log y-axis.

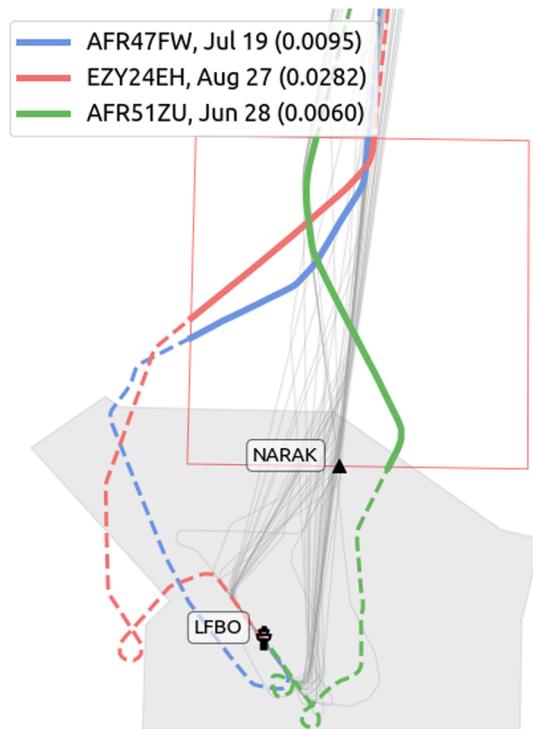


Fig. 8. Situations with high reconstruction errors on the city-pair scenario.

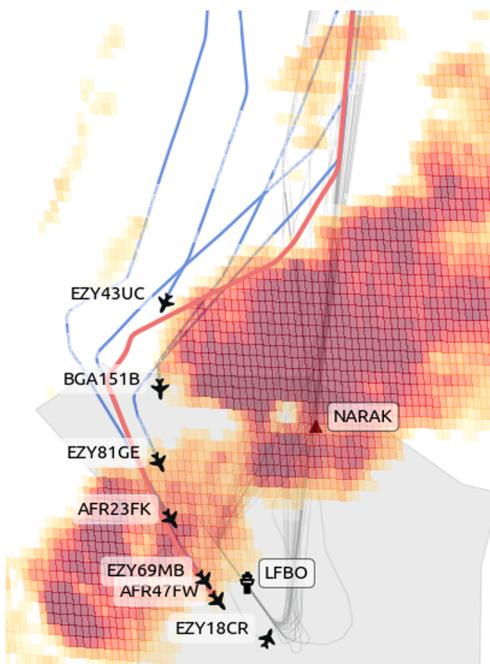


Fig. 9. Weather impact on STAR procedures on the city-pair scenario.

we study in the following section the contextual situations associated to such specific trajectories pointed on the distribution. We focused first on trajectories with the highest reconstruction errors, which should be representative of the most unusual trajectories; then on a few situations with lower reconstruction errors, closer to the tail of the bell-shape distribution.

The analysis of the context was made after analysing traffic around anomalous detected trajectories. ADS-B may be an incomplete

source of data for a thorough analysis of the situation since some aircraft are still not properly equipped. In that case, flight plans and trajectories inferred by multilateration give a hint but situations analysed in this paper have been selected so as to be representative of the wide panel of detected situations and to be explainable with ADS-B tracks openly accessible. A perfect analysis would have also involved confirmation of our hypotheses with radio recordings; unfortunately, none were available for the chosen scenarios.

In general, we found that very high reconstruction errors are associated to less common situations; explanations are more to be found in the METAR history or regulation history. Conversely, reconstruction errors closer to the tail of the bell-shaped distribution, are more prone to yielding nominal situations which may be explained by ATC orders issued for a deconfliction or sequencing purpose.

4.2.1. Highest values of reconstruction errors relate to less common situations

Flight AFR51ZU on June 28th ([Fig. 8](#)) yields a high (although not the topmost) reconstruction score and its peculiar route on final approach called the authors' attention. This day was marked with a lot of delays because of weather. METAR on that day is particularly explicit: thunderstorm (TS), presence of cumulonimbus (CB) and a gusting wind forecast (20G35KT). Local news reported that traffic was interrupted on that day in Toulouse–Blagnac when AFR51ZU was in approach. The aircraft was not rerouted, but the interruption may explain the unusual path and the two loops during final approach.

```
LFBO 281600Z AUTO 20006KT 180V240 9999 TS FEW033/// SCT047/// BKN060/////CB 20/14 Q1004 TEMPO  
28020G35KT 2000 TSRA=
```

Traffic around Toulouse airport on July 19th was also impacted by poor weather and cumulonimbus located on the path of the STAR procedure starting from NARAK. All aircraft (including AFR47FW, see [Fig. 9](#)) coming from the North-East were deviated to the West prior to entering the TMA, so as to be sequenced on the usual STAR procedures applicable to aircraft coming from the North-West.

```
LFBO 192100Z AUTO 02004KT 350V050 9999 TS BKN024/// OVC031/////CB 23/19 Q1014 TEMPO 1500 TSRAGR BKN010  
BKN040CB BECMG 14010KT=
```

QFU refers to the magnetic heading of the runway in use. ATC may change the QFU at any time depending on the weather conditions. On August 27th, EZY24EH took a very peculiar route ([Fig. 8](#)) as other aircraft were landing from the south on QFU32. After the last flight has landed (16:51), a first aircraft landed at 17:08 on QFU14 and EZY24EH was on hold before landing 3 min later.

As we validated our approach on different city pairs, including flights from Paris–Charles-de-Gaulle (LFPG) to Frankfurt (EDFF), we found that regulations set by ATC may also result in very high reconstruction errors. Historical records contain many regulations in place between Paris and Frankfurt (Reims and Langen ACC) in early evening on June 2nd, 2017, because of cumulonimbus present in the area. In particular, DLH2F was impacted by a 15 min delay before departure because of a regulation filed by Frankfurt arrivals. [Fig. 10](#) plots the last filed flight plan (Filed Tactical Flight Model, FTFM) and the Current Tactical Flight Model (CTFM), refined version of the FTFM based on live positions for DLH2F. Such a pattern suggests that DLH2F adapted its route to avoid further regulations in the area.

4.3. High values of reconstruction errors relate to tactical ATC actions

In contrast, we found more classical and conventional ATC situations in the belly of the reconstruction error distributions.

On [Fig. 11](#), AFR27GH has a higher reconstruction error, probably because its trajectory is pushed quite to the East of the NARAK beacon, a less usual pattern for trajectories on the LFPO to LFBO route. As we investigate closer into the situation, it appears that AFR27GH was flying behind EZY81GE before being instructed to turn left. As EZY743L arrived from Lyon (to the North-East), AFR27GH was sequenced behind with an appropriate ATC order. RYR3YM arrives next and is sequenced behind AFR27GH in a similar manner. In this situation, only AFR27GH, plotted in red, is part of our dataset. Other aircraft have been collected and plotted to give an idea of the contextual situation.

On [Fig. 12](#), DLH4J flies into Frankfurt area. The density of traffic converging on this IAF at this time of the day would probably not explain this shift in trajectory. The explanation could come from TAP571 which took off a bit earlier from runway 18. Without a turn left order to DLH4J, her climb path could have crossed DLH4J's trajectory, probably causing a potential loss of separation.

4.4. Airspaces

The provided dataset includes seven months of trajectories from the LFBBPT sector operated by Bordeaux Area Control Center (ACC) (see [Fig. 13](#)) for periods when the sector was operationally deployed according to the Sector Configuration Plans (SCP), also known as opening schemes.

The challenge of this section is to apply our anomaly detection method to identified flows of trajectories entering and leaving a sector rather than from city-pair trajectories. Identified clusters may result from the aggregation of sparsely distributed trajectories.

The flow identification method applied in this paper for the use case concerning the en-route airspace ([Basora et al., 2018; Olive and Basora, 2019a](#)) is based on a progressive clustering technique originally developed by [Andrienko et al. \(2018\)](#), [Rinzivillo et al. \(2008\)](#). This method is based on the DBSCAN algorithm, which requires two main parameters determining the size n and the density ϵ of clusters. For the first application of DBSCAN, we have set ϵ to 0.4 and n to 1% of the total number of trajectories. For the refinement of the clusters, DBSCAN has been executed with ϵ set to 0.5 and n set to 1% of the total number of trajectories in the cluster where it is applied. The minimum number of trajectories for a cluster to be formed has been established to 2% of the traffic in the sector.

The resulting cluster centroids representing the flows are displayed in [Fig. 14](#). A total number of nine flows have been identified with a percentage of outliers reaching 26.4% of the traffic. We have checked how well the generated clustering centroids match the ATS Route Network (ARN) also published on eAIP. Some clusters fit well to sections of the published air routes, e.g. UN869 (cluster 1),

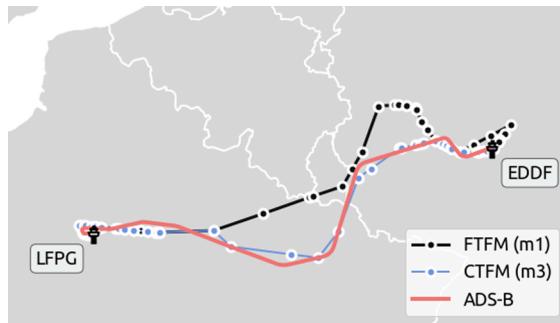


Fig. 10. Last filed flight plan and trajectory of DLH2F on June 2nd, 2017, detected as outlier on one year of traffic between LFPG and EDDF.

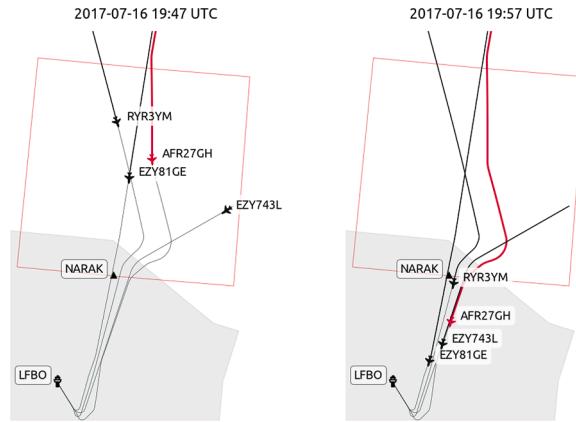


Fig. 11. Aircraft are sequenced for landing before entering Toulouse TMA; AFR27GH is vectored behind EZY81GE and EZY743L.

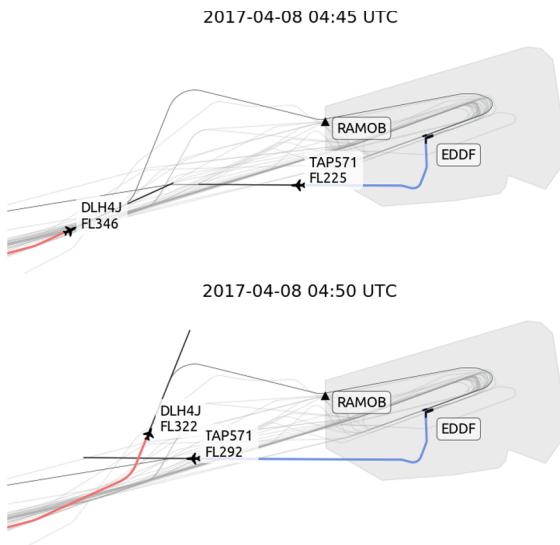


Fig. 12. Deconfliction between DLH4J scheduled for landing and TAP571 taking off from Frankfurt airport, runway 18.

UM728 (cluster 5) or UN460 (cluster 7) but a similar match is less evident with clusters 0, 3 and 4. Cluster 0 is an evolving flow with flights taking off from Paris area, so it seems reasonable to have it separated from clusters 3 and 4 which are both stable flows with a centroid at FL360. The reason for separating clusters 3 and 4 is less obvious, but certainly due to the fact that the exit points of these two clusters are separated by a less dense area, which can be observed in Fig. 13.

All trajectories within a cluster are then considered independently. We applied to each cluster the anomaly detection technique

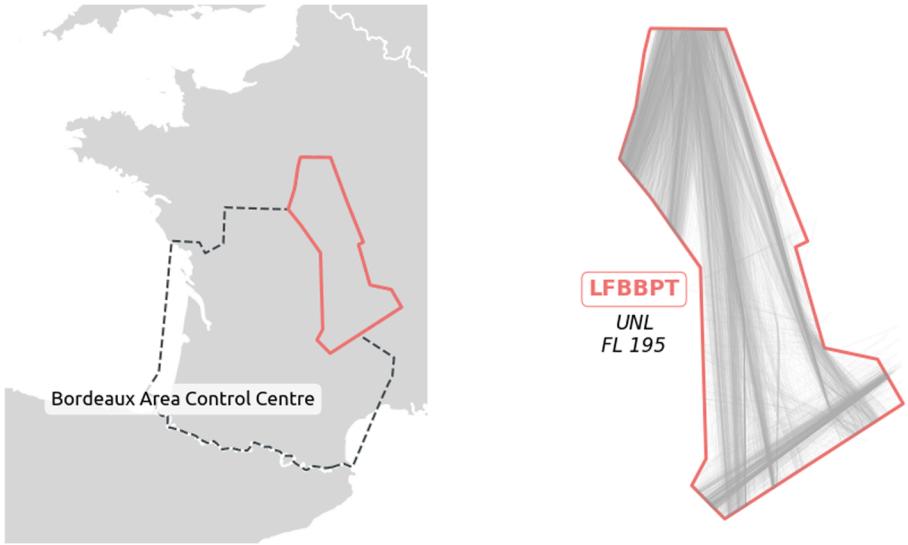


Fig. 13. Description of the airspace dataset over the LFBBPT sector of the Bordeaux Area Control Centre.

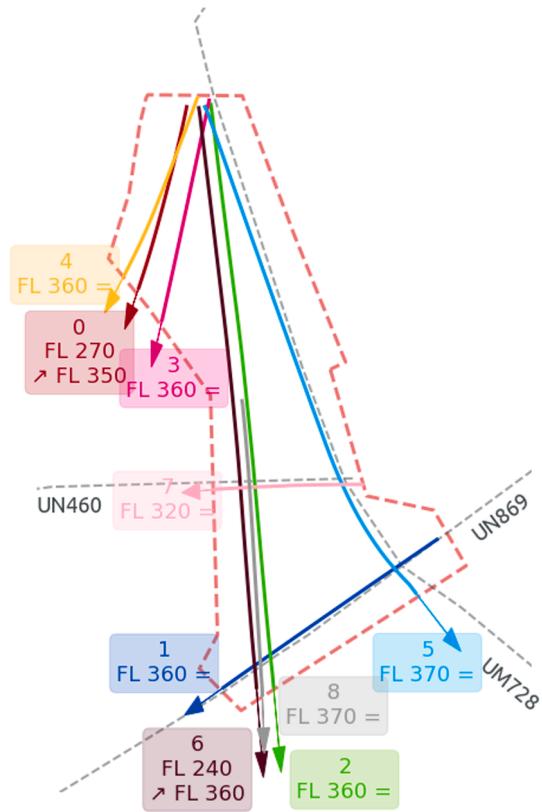


Fig. 14. The clustering method applied to our dataset trimmed to LFBBPT sector resulted in the nine following clusters. The altitude below the identifier of the cluster reflects the altitude profile of the centroid.

presented in Section 3.1 with a different autoencoder network architecture, adapted to the resampling of our trajectories at 50 points per trajectories, working only with normalised track angles so as to focus on lateral resolutions of potential conflicts.

The MSE loss converged properly after 3000 iterations for each cluster, although the distribution of reconstruction errors of clusters with more sparsely distributed trajectories (e.g. cluster 4) lead to distribution profiles suggesting two modes, as reflected in Fig. 4. The regularisation term presented in Section 3.3 lead to better results and validated the idea behind regularisation which consists in

penalising our criterion (the MSE loss) hoping we can avoid overfitting and converge toward more robust solutions.

4.4.1. Weather related events

We found in Section 4.2 with the city-pair data set that once the autoencoder has converged, the few samples with the highest reconstruction errors were associated with exceptional events (mostly weather related) whereas reconstruction errors located in the “belly” of the distribution were matching tactical ATC operations. In order to validate this assumption with the airspace data set, we selected the top 10 trajectories with the highest reconstruction errors for each cluster, for a total of 90 trajectories. Table 1 shows that two days were particularly represented in that subset of trajectories. The last column (rank) reflects the position of the sample in the distribution: 1 stands for the highest reconstruction error in the cluster, 2 for the second highest, etc.

A first look at the METAR history in airfields located around the LFBBPT sector (LFLX, Châteauroux, to the North-West of LFBBPT and LFLC, Clermont-Ferrand in the Southern part of the sector) reflects locations of cumulonimbus (CB) and tower cumulus (TCU) consistent with the location of anomalous trajectories (Fig. 15): CB impacted the whole sector (hence clusters 2, 4, 5, 6, 7 and 8) on June 20th but only the Northern part of the sector (mostly clusters 0) on July 8th.

Fig. 15 plots positions, trajectories for the past 15 min of aircraft flagged as anomalous by our method on June 20th 2017. The location of cumulonimbus is estimated from Thermal IR data from the Spinning Enhanced Visible and InfraRed Imager (SEVIRI), collected by the Meteosat Second Generation series of satellite.

The sector was implemented during the last hours of daylight, which may suggest that pilots avoid thunderstorms areas based on their visual perception in the early hours of operation, and relied on their on board weather radar in the later hours. Another explanation for aircraft seeming to fly through CBs may rely in the anvil shape of such cloud developments: satellite images only see the top of the clouds so it may be safe to fly below the cap and still avoid the vertical development of the cloud. Based on this kind of heatmap, future safety studies could assess how pilots manage to avoid thunderstorm areas based on the information of on board weather radars.

4.4.2. Detection of tactical ATC actions

Isolating deconfliction ATC orders in regular traffic is a difficult task because most flights are executed without much deviation from their original intention. Reconstruction errors help isolate flights calling for further analysis. We considered hereafter only a subset of our trajectories, namely trajectories with a reconstruction error higher than a given threshold. We defined the threshold based on the fitted exponential distribution f defined in (3) and illustrated in Fig. 16 with the set of samples $\{x_i\}$ s.t.

$$f(x_i) \leq \frac{1}{5} \cdot f(0) \text{ i.e. } x_i \geq \log(5) \cdot \ell_{\text{cluster}}(u, v) \quad (7)$$

In an attempt to automatise the process, we computed the closest point of approach (CPA) for all pairs of trajectories which fly at the same moment in LFBBPT and which belong to our subset of trajectories defined in (7). For the CPA computation, we used the distance between two trajectories based on the cylindrical norm defined in Munoz and Narkawicz (2010):

$$d_{\text{CPA}} = \min_t \left(\max \left(\frac{d_{\text{lat}}(t)}{5 \text{ nm}}, \frac{d_{\text{vert}}(t)}{1,000 \text{ ft}} \right) \right) \quad (8)$$

where d_{lat} is the distance between the two WGS84 coordinates and d_{vert} the difference of altitudes. 5 nm and 1000 ft are respectively the lateral and vertical separation minima required between aircraft flying within Reduced Vertical Separation Minima (RVSM) airspace (ICAO, 2008). Since we observe traffic which has allegedly been deconflicted by ATC, all pairs of trajectories should be separated by a distance $d_{\text{CPA}} \geq 1$. However, we assume that an action of deconfliction is likely to involve pairs of trajectories with a d_{CPA} relatively small.

In the following we focus on pairs of trajectories respecting the following conditions:

Table 1
Most significant trajectories/days in LFBBPT grouped by date.

Callsign	Date and time of entry in LFBBPT	Cluster	Rank
TRA47R	2017-06-20 18:21:00Z	2	6
RYR9TG	2017-06-20 19:09:28Z	4	1
TOM84T	2017-06-20 19:14:51Z	4	3
RYR79EY	2017-06-20 18:35:46Z	4	7
SAA235	2017-06-20 19:07:15Z	5	9
DAH1007	2017-06-20 18:33:34Z	6	8
DLH68F	2017-06-20 18:25:17Z	7	8
VLG83TJ	2017-06-20 18:21:00Z	8	2
AFR88DM	2017-07-08 19:55:01Z	0	1
VLG8248	2017-07-08 20:09:53Z	0	6
AEA1008	2017-07-08 19:39:50Z	0	9
AAF221	2017-07-08 20:08:52Z	0	10
FIN611	2017-07-08 19:42:22Z	2	8

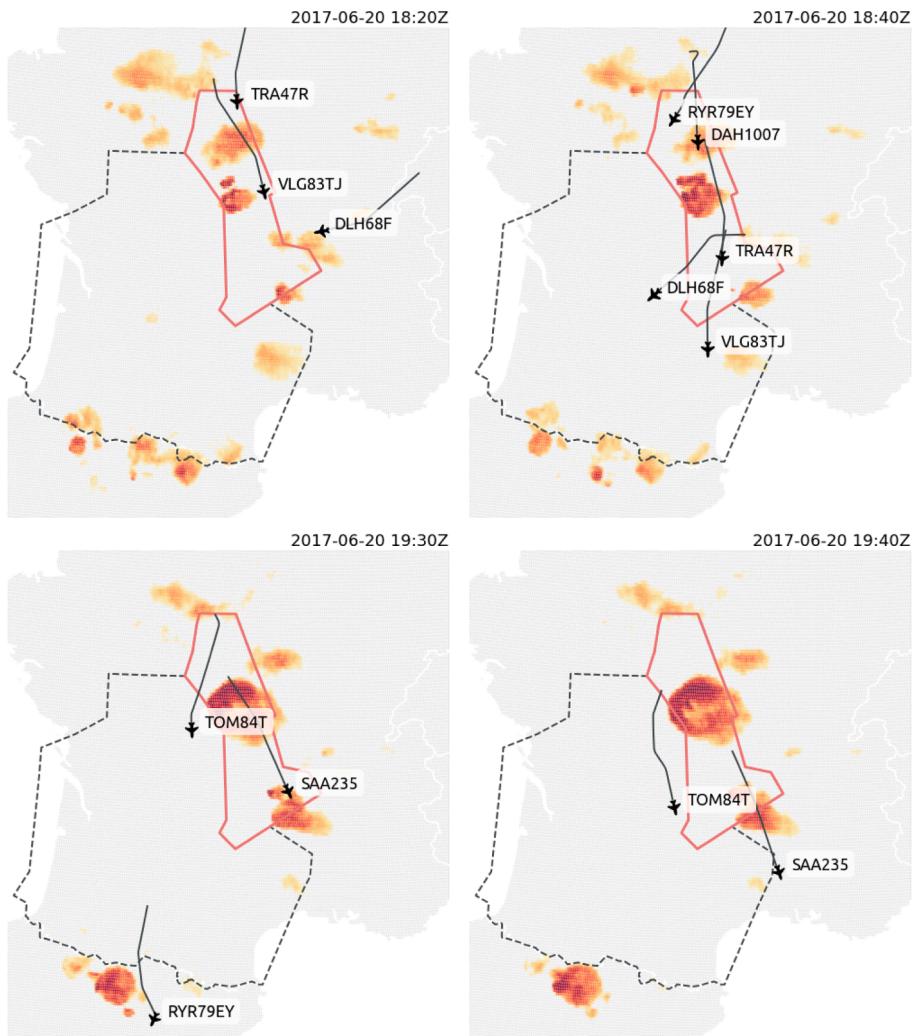


Fig. 15. Trajectories of aircraft flying to avoid thunderstorm activities.

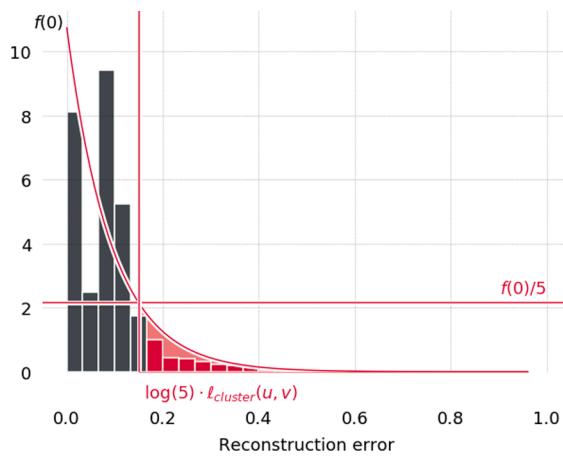


Fig. 16. Selection of the sample trajectories with the higher reconstruction scores (example of cluster 4). For the purpose of the study, we chose a threshold of $1/5 = 20\%$ although this could be reconsidered in the future.

1. each trajectory has a reconstruction error higher than $\log(5) \cdot \ell_{\text{cluster}}(u, v)$ (see Fig. 16);
2. their $d_{\text{CPA}} \leq 2$, i.e. the lateral and vertical distance at the CPA should be smaller than 10 nm and 2000 ft respectively. In addition, we impose a constraint on the vertical distance to be smaller than 1500 ft in order to focus only on aircraft flying at adjacent flight levels.

Each trajectory being associated with a cluster, we build the density matrix as shown in Fig. 17 which reads as follows: the darker the colour at position (i, j) with $i \geq j$, the more trajectories from cluster i and j are possibly subject to a deconfliction order from the ATC. A first consistency cross-check with the map on Fig. 14 seems convincing: cluster 0 interacts with clusters 3 and 4 (mostly cluster 3); cluster 1 interacts with clusters 2, 6 and 8, albeit less with cluster 5. Cluster 7 flies at a relatively lower altitude and only interacts with cluster 6 (trajectories climbing) but not with clusters 2 and 8 (constant altitudes).

We focus in the following on specific situations in converging flows, then on pairs of trajectories in the same flow which may be impacted by the same factors.

Fig. 18 reflects two situations involving cluster 1 and one of clusters 2, 6 and 8, containing mainly deconfliction situations for aircraft flying at the same level. The first situation of Fig. 18 involves TVF021Z (cluster 6) and TAP933A (cluster 1). The dashed line projects the situation as if no order had been given five minutes before the closest point of approach with a possible conflict situation at the intersection of both routes. On the right hand side, ATC orders given to IBK7VY and IBK6113 seem to have anticipated the situation earlier with a probable deconfliction order given ten minutes before the closest point of approach. For such situations of converging routes at the same flight level, a future direction for improvement could be to automatically detect the level of anticipation of the deconfliction by looking backward from the CPA.

The density matrix on Fig. 17 reflects a high number of potential deconfliction situations between pairs of trajectories from cluster 0 and from cluster 1. Fig. 19 looks into those situations. On the left hand side, AEA1038 takes off from Paris–Charles-de-Gaulle and RAM781S from Paris–Orly. They both belong to cluster 0 and will probably be vectored on the same route. Having similar climb profiles (both aircraft are B738), special attention is paid to their lateral separation, probably leading to these peculiar trajectories. On the right hand side, TAP817 flies Milan–Porto while CES709 flies Shanghai–Madrid. When both aircraft join route UN869, a special attention is paid to their separation (see the level-off at FL360) before they are probably separated by being placed on lateral offsets from UN869.

4.5. Terminal manoeuvring areas

The provided data set includes 19,480 trajectories landing at Zurich airport between October 1st and November 30th 2019. Here, we applied a stacked autoencoder on 14,399 trajectories landing on runway 14 which is the runway the most often in operation. All trajectories are selected from the moment they enter the area within 40 nautical radius around the airport until the last point before the runway threshold (see Fig. 20).

The challenge here is to be able to detect anomalies in a sample of converging flows, in contrast with Sections 4.2 and 4.4. The autoencoder architecture we selected reduced 100 samples of normalised track angles to 48, 24 then 6 neurons per layer (encoding part) before reconstructing the samples through layers of 24, 48 and 100 neurons each. Fig. 21 plots a distribution of reconstruction errors after 10,000 iterations which looks reasonable in spite of a low density in the bin containing the best reconstructed samples.

Fig. 22 plots abnormal trajectories and gives hints to interpret why they are so poorly reconstructed. One one hand, Flight EWG7ME had three landing attempts (the first one aborted at an altitude of 2000 ft), and Flight OTF6410 stacked six holding patterns before being allowed to proceed further and align runway 14. However, the autoencoder did not only fail at recovering holding patterns in trajectories: Fig. 23 explains why Flight AZA572's trajectory is atypical and non representative of aircraft landing from the South-East incoming flow. The right part of the figure shows how landing operations where disrupted during that time frame and could provide hints why the aircraft was instructed that way.

5. Information extraction on the latent space

This reconstructing approach may not be the most relevant to detect ATC tactical instructions in a Terminal Manoeuvring Area when pilots are in constant contact with ATC. Yet it remains a powerful tool to detect unusual patterns.

As explained in details in Section 3.1 and Fig. 2, the first part of the autoencoder is a projection operator. Samples are projected onto a smaller dimension space before a generation operator attempts at reconstructing the original samples. In spite of a resulting poorer reconstruction ability, we added a layer to the network used in Section 4.5 in order to project our samples further onto a two-dimensional space in order to plot the distribution of projected samples on Fig. 24. Colours are assigned to all trajectories based on the bearing (interval) from the point they enter the 40 nautical mile radius with respect to the airport. Each colour can be associated to an incoming flow into the TMA of Zurich airport. Fig. 24 shows how the network learned to organise all trajectories on the two dimensional space based on similar features.

Fig. 25 plots the same distribution for the subset of self-intersecting trajectories, to be related to holding patterns. Three pairs of trajectories that are close to each other in their 2D representation are represented: the top right map displays two trajectories stacking two holding pattern before getting the clearance to land on runway 14. They are both located on the left part of the “cluster” matching the flow of self-intersecting trajectories landing from the West-North-West of Zurich. The two other pairs of trajectories on the bottom right of the same cluster also display similar features when plotted on a map.

Looking at the latent space of autoencoders trained to reconstruct trajectories is particularly enlightening with respect to their

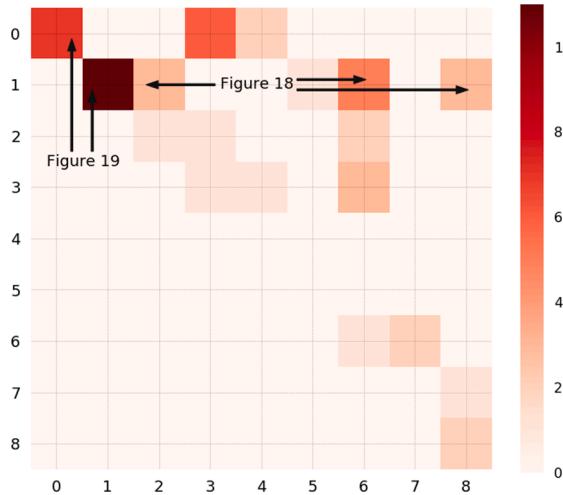


Fig. 17. Density matrix (upper triangle only): the darker the colour in (i,j) , the more trajectories from clusters i and j are possibly subject to a deconfliction order from the ATC.

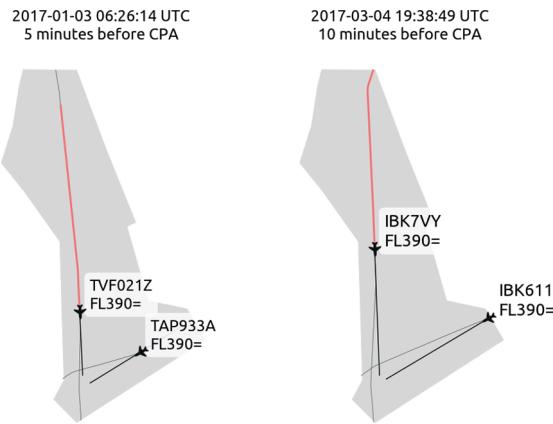


Fig. 18. Deconfliction actions between cluster 1 and cluster 6 (left, on January 3rd), resp. cluster 1 and cluster 2 (right, on March 4th).

ability to extract information from large amounts of data which comes as a side effect of their first intended use.

6. Discussion

We presented in this paper a full framework able to detect and identify significant events in air traffic data. Given a large number of trajectories, the method makes use of autoencoders and considers distributions of reconstruction errors in order to select a subset of situations which are easier to put in context and analyse. The position of a sample in the distribution seems related to the nature of the selected situation.

Autoencoders are trained to fit a projection operator into a lower dimensional space and a generation operator to reconstruct the original data, with a tendency to better reconstruct statistically common samples. Once an autoencoder has been trained on a given dataset, new trajectories matching the same data preprocessing (cleaning, clipping, resampling, etc.) can be projected and reconstructed using the same operators in order to compute a new reconstruction error. However, the resulting value only makes sense when considered with respect to the original distribution.

Commonly, anomaly detection techniques assume that most samples in a given dataset are normal and that few samples do not reflect a usual behaviour. When the objective is to search for ATC deconfliction orders, an anomaly detection technique makes full sense because these situations are not so common in an en route context. Conversely, common situations distributing around one or several reference trajectories will tend to cluster in the latent space and to be more easily reconstructed.

Autoencoders have this intrinsic clustering tendency in their latent space, which may serve as an explanation regarding how they

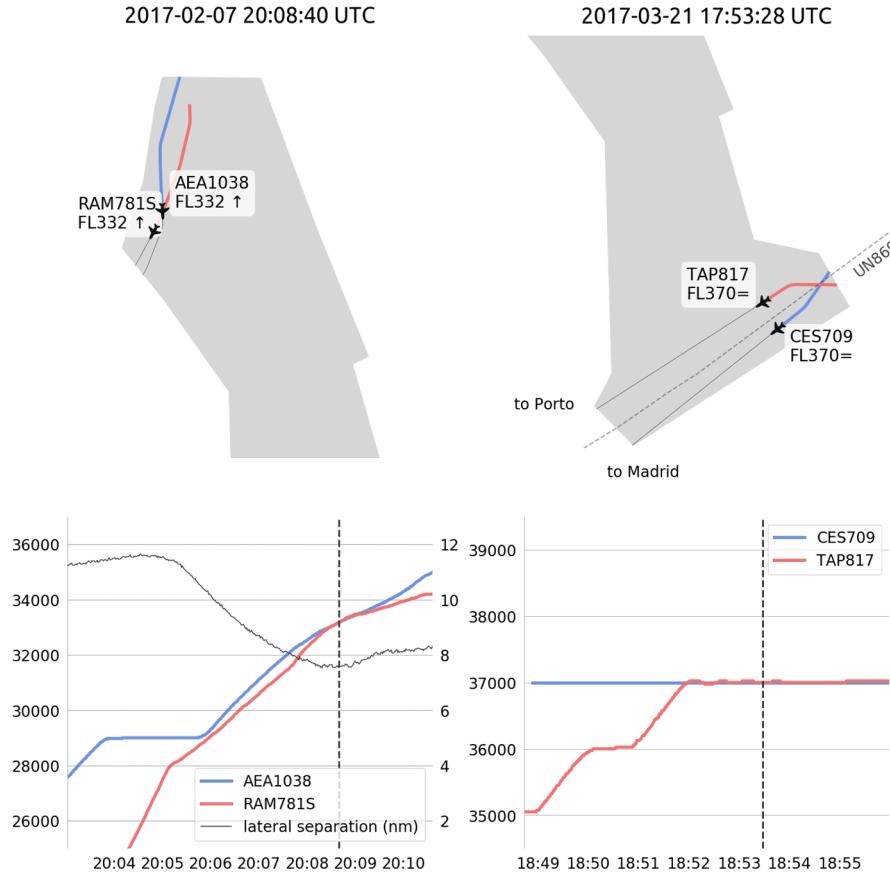


Fig. 19. Deconfliction situations between trajectories in the same clusters.

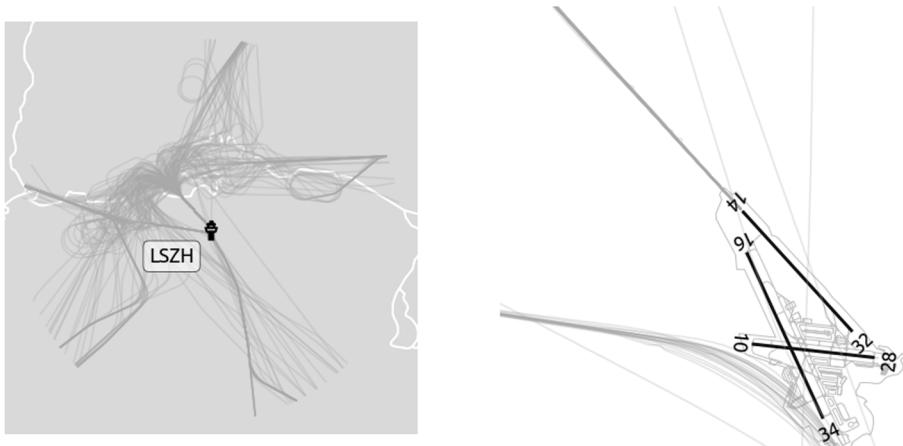


Fig. 20. All trajectories landing at Zurich airport are cropped between a 40 nautical miles distance to the airport and the runway threshold.

process data.¹ However, a neural network trained on a simple single flow data set (the *city-pair scenario*) will have a better reconstruction ability than a network trained on a more complex dataset (the *landing scenario*). Then, we presented a regularisation term in order to cope with the poorer reconstructive power of autoencoders on datasets with several modes on the *airspace scenario*.

The objective of this contribution is to detect and identify significant events. As the clustering insight produced on the latent space

¹ A possible labelling of clusters based on the latent representation is less direct.

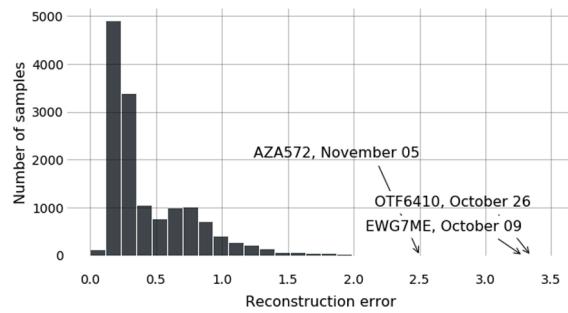


Fig. 21. Distribution of the reconstruction errors on the landing dataset. Highlighted samples are plotted on [Fig. 22](#).

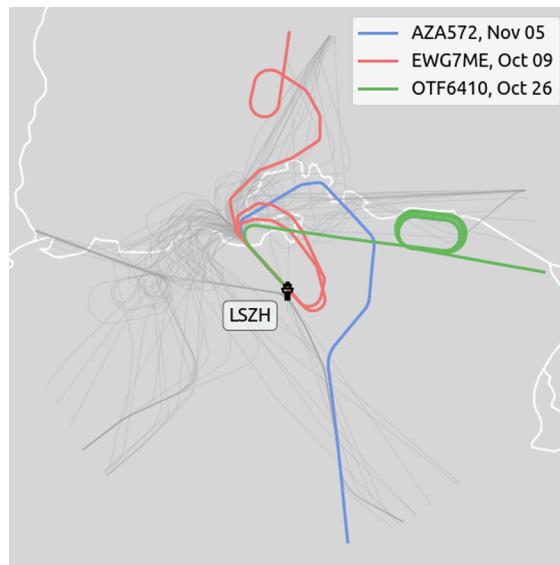


Fig. 22. These three trajectories have very high reconstruction errors: Flight EWG7ME had three landing attempts, Flight OTF6410 stacked six holding patterns; the particular case of Flight AZA572 is further detailed on [Fig. 23](#).

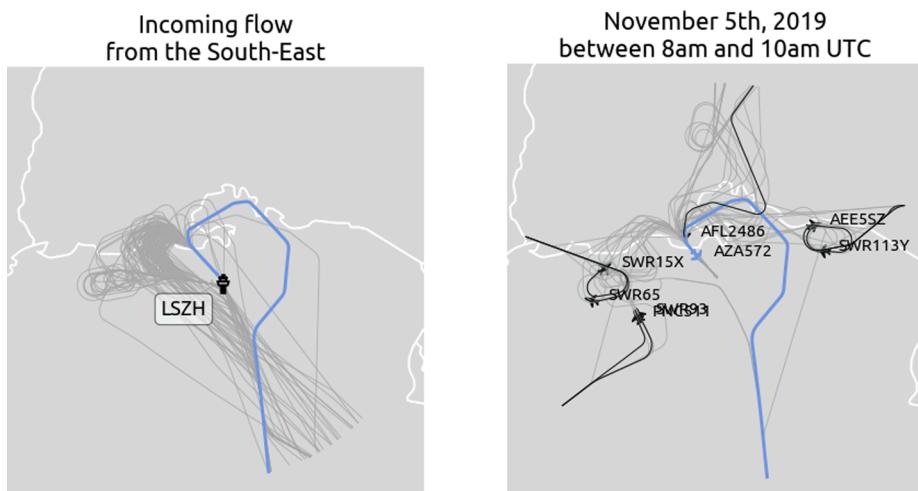


Fig. 23. AZA572 is atypical with respect to the usual approach from the South-East (left). Indeed, traffic was particularly disrupted during that time frame (right).

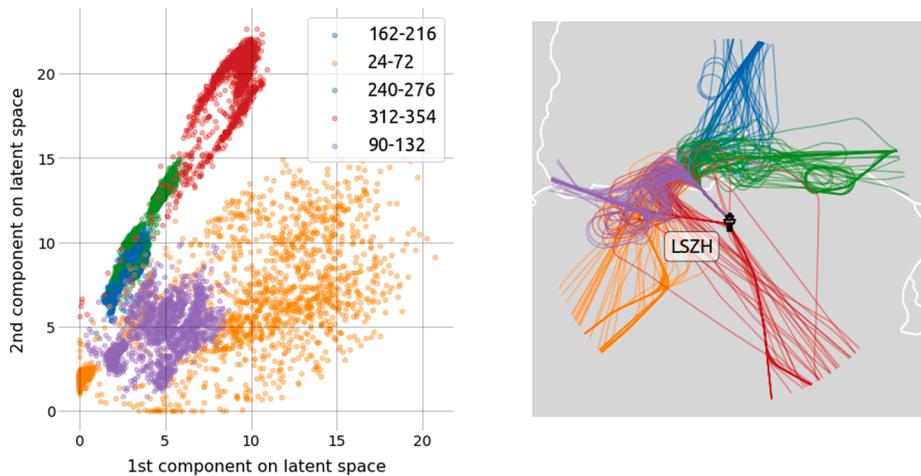


Fig. 24. As autoencoders learn to reconstruct samples, they project the data on a small dimension space. Even when forced onto a latent space with two dimensions, they tend to organise the data by similarity. Here colours are associated to an incoming flow.

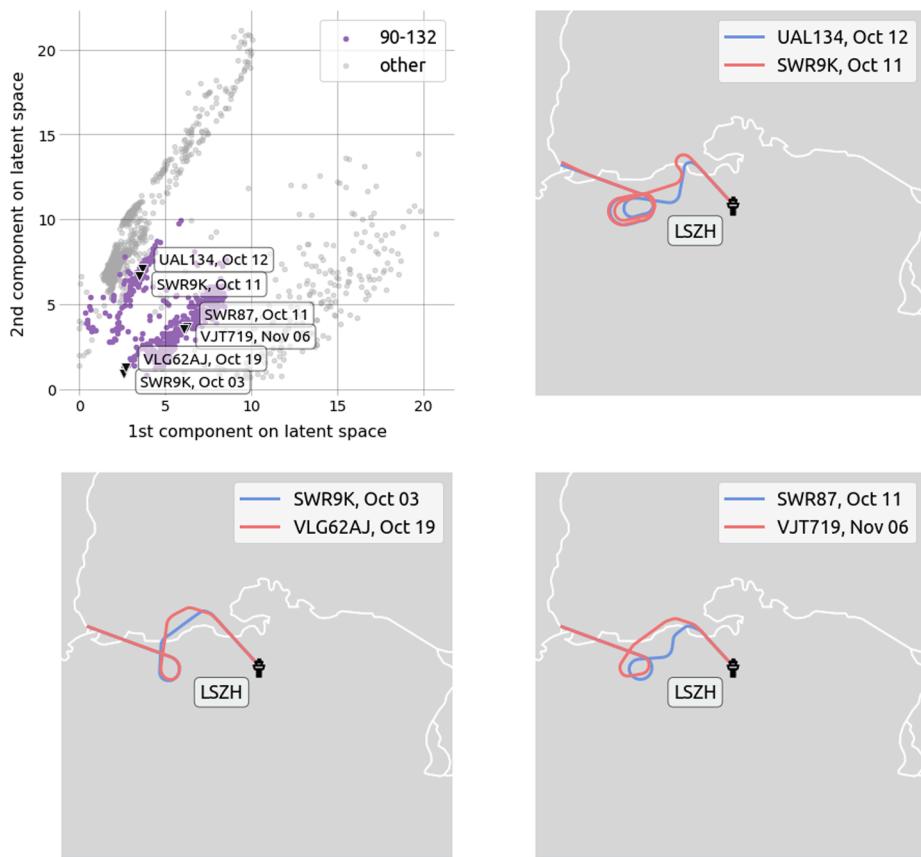


Fig. 25. The scatter plot displays the latent space on trajectories containing one or several loops. Samples that are close to each other reveal similar patterns on the map. In particular, trajectories stacking two holding patterns are plotted on the top right and correspond to the left part of the purple cluster in the scatter plot.

is not the main focus of interest, we recommend to first proceed with a basic clustering (as simple as one based on entry/exit pairs) before training the autoencoder on each cluster. The resulting latent space will be different, less cluster-oriented and will probably better learn how to reconstruct samples. Resulting distributions become more relevant for the cherry-picking of situations of interest.

7. Conclusions

We presented in this paper a simple and successful framework to detect, identify and characterise anomalies in past aircraft trajectory data. The method is based on unsupervised learning methods, namely autoencoders and clustering algorithms. We challenged this technique on three data sets very different in nature albeit covering a wide range of situations commonly encountered by air traffic controllers.

We introduced autoencoding neural network architectures and explained how a resulting reconstruction error can give hints about the nature of detected anomalies. Most anomalies are the result of ATC tactical actions, and the higher the reconstruction error, the more disruptive the situation. We found critical situations resulting from the impact of poor weather conditions, regulation measures and capacity issues.

This method has a wide range of applications in air transportation studies. Future works should leverage the information extracted from such massive amounts of data in order to produce predictive models in terms of ATC tactical actions (from an ANSP point of view) or estimated take-off and landing times (from an airline or airport point of view). Statistical studies on large amount of detected unusual situations would also improve the quality of safety analyses and collision risk models.

Lastly, unsupervised machine learning being about manipulating unlabelled data sets and extracting information, the question remains as to how to measure the quality of detection models such as the one introduced in this paper. Finding anomalies in past trajectory data is akin to looking for needles in a haystack but the variety of situations presented in this contribution attests the knowledge discovery power unleashed by this approach. In addition to advocating open data sets which are provided for reproducibility and comparison concerns, the authors will pursue their research efforts in determining a convincing set of metrics measuring the relevance and accuracy of such frameworks.

CRediT authorship contribution statement

Xavier Olive: Conceptualization, Methodology, Software, Validation, Formal analysis, Data curation, Visualization, Writing - original draft, Writing - review & editing. **Luis Basora:** Methodology, Software, Validation, Formal analysis, Data curation, Writing - original draft, Writing - review & editing.

Acknowledgments

In no particular order, the authors would like to thank Julie Saint-Lot from ENAC for her invaluable help in providing an ATC feedback from the first days of this research. Richard Alliger from ENAC found a bug in the code we published with our first implementation for the regularisation term in Section 3.3 and immediately offered to fix it. Finally, Simon Proud from Oxford University provided the weather data which helped understanding the most disrupted situations in our datasets.

References

- Andrienko, G., Andrienko, N., Fuchs, G., Garcia, J.M.C., 2018. Clustering trajectories by relevant parts for air traffic analysis. *IEEE Trans. Visualizat. Comput. Graphics* 24.
- Basora, L., Morio, J., Mailhot, C., 2017. A trajectory clustering framework to analyse air traffic flows. In: Proceedings of the 7th SESAR Innovation Days.
- Basora, L., Courchelle, V., Bedouet, J., Dubot, T., 2018. Occupancy peak estimation from sector geometry and traffic flow data. In: Proceedings of the 8th SESAR Innovation Days.
- Basora, L., Olive, X., Dubot, T., 2019. Recent advances in anomaly detection methods applied to aviation. *Aerospace* 6.
- Campello, R.J., Moulavi, D., Sander, J., 2013. Density-based clustering based on hierarchical density estimates. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, pp. 160–172.
- Campello, R.J., Zimek, A., Sander, J., 2015. Hierarchical density estimates for data clustering, visualization, and outlier detection. *ACM Trans. Knowledge Discovery from Data (TKDD)* 10, 5.
- Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: A survey. *ACM Comput. Surv.* 41, 1–58.
- Das, S., Matthews, B.L., Srivastava, A.N., Oza, N.C., 2010. Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 47–56.
- Deshmukh, R., Hwang, I., 2019. Anomaly detection using temporal logic based learning for terminal airspace operations. In: AIAA Scitech 2019 Forum, p. 0682.
- Eckstein, A., 2009. Automated flight track taxonomy for measuring benefits from performance based navigation. In: 2009 Integrated Communications, Navigation and Surveillance Conference. IEEE, pp. 1–12.
- Enriquez, M., 2013. Identifying temporally persistent flows in the terminal airspace via spectral clustering. In: Proceedings of the 10th USA/Europe Air Traffic Management Research and Development Seminar.
- Ester, M., Kriegel, H.-P., Sander, J., Xu, X., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. AAAI Press, pp. 226–231.
- Garcia Gonzalez, E.J., 2013. Development of a 3-dimensional mathematical collision risk model based on recorded aircraft trajectories to estimate the safety level in high density en-route airspaces, Ph.D. thesis, Aeronauticos.
- Gariel, M., Srivastava, A.N., Feron, E., 2011. Trajectory clustering and an application to airspace monitoring. *IEEE Trans. Intell. Transp. Syst.* 12, 1511–1524.
- ICAO, 2008. A unified framework for collision risk modelling in support of the manual on airspace planning methodology with further applications, circ 319-an/181 ed, International Civil Aviation Organization, Montreal, Canada.
- Janakiraman, V.M., Nielsen, D., 2016. Anomaly detection in aviation data using extreme learning machines. In: 2016 International Joint Conference on Neural Networks (IJCNN). IEEE, Vancouver, BC, Canada, pp. 1993–2000.
- Jarry, G., Delahaye, D., Nicol, F., Féron, E., 2018. Aircraft atypical approach detection using functional principal component analysis. In: Proceedings of the 8th SESAR Innovations Days 2018.
- Kong, Z., Jones, A., Belta, C., 2016. Temporal logics for learning and detection of anomalous behavior. *IEEE Trans. Autom. Control* 62, 1210–1222.
- Kramer, M.A., 1991. Nonlinear principal component analysis using autoassociative neural networks. *AIChE J.* 37, 233–243.

- Li, L., Gariel, M., Hansman, R.J., Palacios, R., 2011. Anomaly detection in onboard-recorded flight data using cluster analysis. In: Proceedings of the 30th Digital Avionics Systems Conference, IEEE, pp. 4A4–1.
- Li, L., Das, S., John Hansman, R., Palacios, R., Srivastava, A.N., 2015. Analysis of flight data using clustering techniques for detecting abnormal operations. *J. Aerospace Informat. Syst.* 12, 587–598.
- Li, L., Hansman, R.J., Palacios, R., Welsch, R., 2016. Anomaly detection via a Gaussian Mixture Model for flight operation and safety monitoring. *Transport. Res. Part C: Emerg. Technol.* 64, 45–57.
- Lloyd, S., 1982. Least squares quantization in PCM. *IEEE Trans. Informat. Theory* 28, 129–137.
- Munoz, C.A., Narkawicz, A.J., 2010. Time of closest approach in three-dimensional airspace, Technical Report, NASA.
- Murça, M.C.R., DeLaura, R., Hansman, R., Jordan, R., Reynolds, T., Balakrishnan, H., 2016. Trajectory clustering and classification for characterization of air traffic flows. *AIAA Aviation*.
- Olive, X., 2019. traffic, a toolbox for processing and analysing air traffic data. *J. Open Source Softw.* 4, 1518.
- Olive, X., Basora, L., 2019. Identifying Anomalies in past en-route Trajectories with Clustering and Anomaly Detection Methods. In: Proceedings of the 13th USA/Europe Air Traffic Management Research and Development Seminar.
- Olive, X., Basora, L., 2019. Reference data sets for detection and identification of significant events in historical aircraft trajectory data.
- Olive, X., Bieber, P., 2018. Quantitative assessments of runway excursion precursors using mode S data. In: Proceedings of the 8th International Conference for Research in Air Transportation.
- Olive, X., Morio, J., 2019. Trajectory clustering of air traffic flows around airports. *Aerosp. Sci. Technol.* 84, 776–781.
- Olive, X., Grignard, J., Dubot, T., Saint-Lot, J., 2018. Detecting Controllers' Actions in Past Mode S Data by Autoencoder-Based Anomaly Detection. In: Proceedings of the 8th SESAR Innovation Days.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library, arXiv: 1912.01703 [cs, stat]. ArXiv: 1912.01703.
- Pimentel, M.A., Clifton, D.A., Clifton, L., Tarassenko, L., 2014. A review of novelty detection. *Signal Process.* 99, 215–249.
- Puranik, T.G., Mavris, D.N., 2017. Anomaly detection in general-aviation operations using energy metrics and flight-data records. *J. Aerospace Informat. Syst.* 22–36.
- Rinzivillo, S., Pedreschi, D., Nanni, M., Giannotti, F., Andrienko, N., Andrienko, G., 2008. Visually driven analysis of movement data by progressive clustering. *Informat. Visualizat.* 7.
- Sabhnani, G., Yousefi, A., Kostitsyna, I., Mitchell, J., Polishchuk, V., Kierstead, D., 2010. Algorithmic traffic abstraction and its application to nextgen generic airspace. In: Proceedings of the 10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, p. 9335.
- Sakurada, M., Yairi, T., 2014. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In: Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis. Association for Computing Machinery, New York, NY, USA, pp. 4–11.
- Schäfer, M., Strohmeier, M., Lenders, V., Martinovic, I., Wilhelm, M., 2014. Bringing up OpenSky: A large-scale ADS-B sensor network for research. In: Proceedings of the 13th International Symposium on Information Processing in Sensor Networks, pp. 83–94.
- Sun, J., Vu, H., Ellerbroek, J., Hoekstra, J.M., 2019. pyModeS: Decoding Mode-S surveillance data for open air transportation research. *IEEE Trans. Intell. Transp. Syst.*
- Zhou, C., Paffenroth, R.C., 2017. Anomaly Detection with Robust Deep Autoencoders. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '17. ACM Press, Halifax, NS, Canada, pp. 665–674.