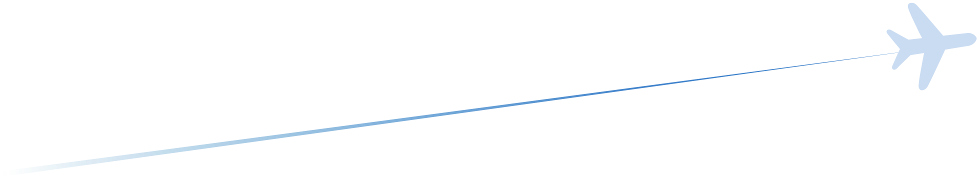
EXPLORATORY RESEARCH



4D trajectory prediction module

Deliverable ID:	D3.1
Dissemination Level:	PU
Project Acronym:	AISA
Grant:	892618
Call:	H2020-SESAR-2019-2
Topic:	SESAR-ER4-01-2019
Consortium Coordinator:	FTTS
Edition date:	31 July 2021
Edition:	00.01.02
Template Edition:	02.00.02

Authoring & Approval

Authors of the document

Name/Beneficiary	Position/Title	Date
Lars L. Schmidt / TU Braunschweig	Research Engineer	10/02/2021

Reviewers internal to the project

Name/Beneficiary	Position/Title	Date
Thomas Feuerle / TU Braunschweig	Team Leader ATM	29/03/2021
Javier A. Pérez Castán / UPM	Assistant Professor	29/03/2021
Tomislav Radišić / FTTS	Assistant Professor (PC)	05/07/2021

Approved for submission to the SJU By – Representatives of beneficiaries involved in the project

Name/Beneficiary	Position/Title	Date
Thomas Feuerle / TU Braunschweig	Team Leader ATM	30/03/2021
Javier A. Pérez Castán / UPM	Assistant Professor	30/03/2021
Tomislav Radišić / FTTS	Assistant Professor (PC)	27/07/2021

Rejected By - Representatives of beneficiaries involved in the project

Name/Beneficiary	Position/Title	Date

Document History

Edition	Date	Status	Author	Justification
00.00.01	10/02/2021	Draft	L.L. Schmidt	New document
00.00.02	17/03/2021	Draft	L.L Schmidt	Initial draft
00.00.03	26/03/2021	Draft	L.L. Schmidt	Final draft
00.01.00	30/03/2021	First Issue	L.L. Schmidt	First issue
00.01.01	10/06/2021	Draft	L.L. Schmidt	Revised draft
00.01.02	20/07/2021	Second Issue	L.L. Schmidt	Revised

Copyright Statement

AISA

AI SITUATIONAL AWARENESS FOUNDATION FOR ADVANCING AUTOMATION

This deliverable is part of a project that has received funding from the SESAR Joint Undertaking under grant agreement No 892618 under European Union’s Horizon 2020 research and innovation programme.

Abstract

Three individual ML modules are developed in the AISA project to study a shared situational awareness of AI and ATCOs. This deliverable describes the design, development and validation of the ML trajectory prediction module. It aims at predicting the true aircraft track and future positions of flights with the initially filed flight plan and the current aircraft state as input. Therefore a two-step process is established. First a neural network is trained to predict the static aircraft track without any prediction in the time domain. Afterwards the current aircraft state is combined with the predicted track to determine a concrete future position prediction. ADS-B data from The OpenSky Network and flight plan data from the DDR2 database from EUROCONTROL is going to be used as database.

Purpose

In WP3 three individual ML modules were developed to be further studied in the field of shared situational awareness in the AISA project: Trajectory prediction module, conflict detection module and air traffic complexity module. This deliverable D3.1 describes the main objectives, development and an initial validation of trajectory prediction module. Besides presenting the results of module development to interested audiences, this report delivers background-information about the trajectory prediction module to upcoming research in the AISA project.

Intended Audience

There are two main groups of the intended audience:

- Experts from the related fields,
- The AISA consortium.

The development of trajectory prediction module via AI SA deliverable (AISA D.3.1) is important for the consortium as:

- In the framework of WP3, it develops one of the ML modules for the AISA project.
- The document will provide direct input to the other technical work packages (WP3, WP4, WP5) and the related deliverables, by providing the trajectory prediction module developed based on ML techniques.

The document is also useful for external stakeholders, especially the following ones:

- Air Traffic Management (ATM) system developers who would like to understand how AI, and particularly ML techniques, can be integrated into ATM,
- ATM experts conducting related research,

General automation and AI experts would like to see the possible use of AI in a new domain.

Associated documentation

The document is linked to several AISA and ATM documents; here, only the most relevant ones are listed:

- AISA D2.1: Concept of Operations for AI Situational Awareness System [1].
- AISA D2.2: Requirements for automation of monitoring tasks via AI SA [2].
- AISA D3.2: Conflict detection module [3].
- AISA D3.3: Air traffic complexity module [4].

Terminology

Following table lists the abbreviations used in this document.

Abbreviation	Description
ADS-B	Automatic Dependent Surveillance Broadcast
AI	Artificial Intelligence
AISA	Artificial Intelligence Situational Awareness
ATCO	Air Traffic Control Officer
ATM	Air Traffic Management
ETA	Estimated Time of Arrival
GA	Grant Agreement
ML	Machine Learning
TFR	Temporary Flight Restriction
TP	Trajectory Prediction
WP	Work Package

Table of Contents

List of Tables

List of Figures

1 Introduction

Three individual machine learning modules will be developed in the AISA project to conduct studies about a shared situational awareness of AI and ATCOs. Besides a conflict detection module [3] and an airspace complexity module [4], this deliverable describes the development, implementation and validation of the trajectory prediction module. When it comes to predicting future aircraft states, various studies have been conducted on diverse parameters: [5] introduces a machine learning approach to predict the aircraft mass for optimal climb prediction and in [6] machine learning algorithms are trained to predict the aircraft speed, also for climb predictions. [7] suggests a combination of clustering and following modelling with a neural network to predict the ETA of flights. An extensive approach of predicting 4D trajectories with meteorological datasets as additional input is described in [8]. The finally applied trajectory prediction method described in this deliverable follows some similarities proposed in [8]. For example does the trajectory prediction used in AISA also deploy flight plans as a basis-input and a neural network is implemented as part of the prediction method. But whereas the previous study applied a comprehensive approach with multiple input datasets and a large geographical region for predicting the trajectories, the approach in this deliverable tries to follow a bottom-up-approach to develop a simple and easy to use trajectory prediction in a small scale region.

When it comes to aircraft trajectory prediction the problem can grow to a comprehensive task, especially when dealing with four dimensional position predictions. This means, when predicting 4D trajectories, in general consecutive sets of four dimensional state-vectors need to be predicted. Due to the limited resources and to minimize the risk of running into complexity problems at the end of the development-period, a bottom-up-approach is followed in this projects task. The starting point is the decision to use flight plans of DDR2 database [9] and ADS-B data from The OpenSky Network ([10] and [11]) as input. Differences in the initially filed flight plans (DDR2) and the finally flown trajectory (ADS-B) shall be predicted with the developed module. These trajectory deviations can have various reasons: for example directs due to low traffic congestion, re-routings required to resolve trajectory conflicts, TFRs, etc.

After introducing the TP (Trajectory Prediction) concept and the general data flow with the applied technologies, the initial gathering of raw ADS-B data and pre-processing is described and analysed. This includes a simple stochastic approach to obtain adequate ADS-B request-queries for the impala shell of the historical database from The OpenSky Network with previously gathered DDR2 flight plans as basis. Following two different approaches to predict track changes between DDR2 and ADS-B data with a neural network in a static manner are implemented and assessed. As this static prediction approach does not take any temporal information into account, a second prediction step is designed with the current aircraft state as input. Thereby a concrete future position prediction is generated by applying an aircraft fixed pattern. The corresponding altitude is afterwards predicted by mapping the planned altitude level of the flight plans to the predicted position and correcting it with the known prediction errors from previous altitude predictions. Finally the results of the established trajectory prediction method are validated regarding the true future position obtained from the ADS-B data, which has also been used to train the neural network and for the current aircraft state as input for the second step of the prediction process.

2 TP concept and data flow

A completely defined aircraft trajectory comprises four dimensions: one dimension for time and three dimensions for position definition. This makes it a complex field to predict en-route segments of aircraft. In contrary arrival predictions only comprise one dimension: time. When predicting the arrival-time of an aircraft, the arrival-airport is generally known and therefore the highly potential final position is known in advance. To scale down the complexity in the task of en-route trajectory prediction, the time-dimension will be excluded in a first step. Furthermore the positions will only be assessed in the horizontal plane and positions will be discretized by waypoint-definitions. Besides adapting the position-definition into discrete waypoint-probability values for feeding and training neural networks, the fixed waypoint discretization also facilitates a simple static information-delivery in the first approach.

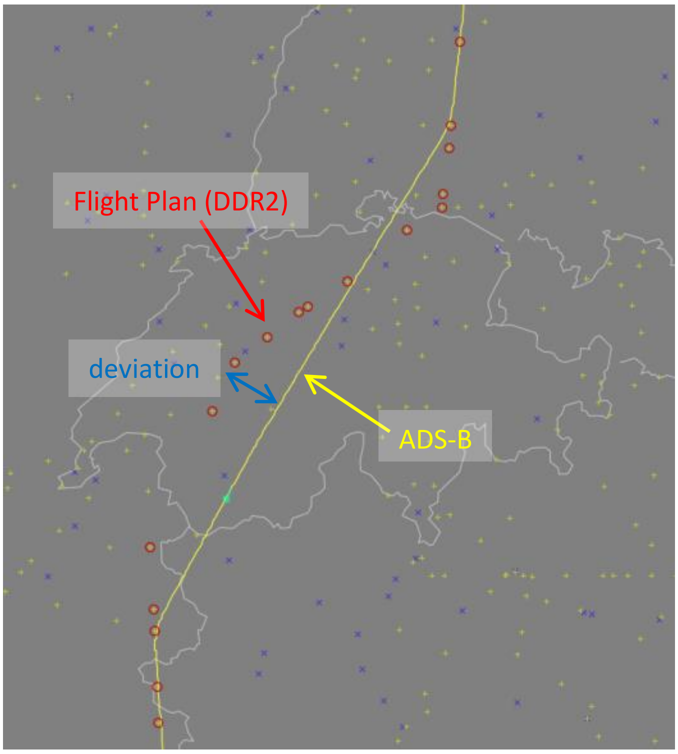


Figure 1: Deviation between flight plan and flown track (ADS-B)[1]

When comparing initially filed flight plans and actually flown trajectories in the horizontal plane, deviations may be observed in some cases (see Figure 1). These deviations can have various reasons: a pilot for example may request a direct to skip certain future waypoints in low congested areas to

[1] Screenshots are taken from the AirTrafficTool, which has been developed at TU Braunschweig and is used in AISA in the framework of Background IPRs.

shorten the flight or an ATCO may request an aircraft to fly an alternate route to solve a conflicting situation with other aircraft. Also the weather-situation may impose necessities to re-route. Other planned re-routings not mapped in the filed flight plan may comprise TFRs (e.g. activated military areas). These deviations between the filed flight plan and the actually flown trajectory is where the concept of the AISA-TP module is deposited. A filed flight plan, known in advance as a scheduled flight, will be the input for a neural network. The neural network shall then model the probability of overflying or skipping certain waypoints in its outputs. For collocation of the actually flown trajectory and to compile the real waypoint-probabilities for training datasets, ADS-B data will be used. Previous studies on ADS-B data of the historical database of The OpenSky Network indicated some gaps in the receiver coverage and various data-faults and misleading trajectory-segments. Therefore a data parser was developed previously to the AISA project at the Institute of Flight Guidance at TU Braunschweig to address these issues in a pre-processing step [12].[2] The overall data-flow of training the neural networks with the ADS-B data parser for pre-processing is depicted in Figure 2. The pre-processed ADS-B data and DDR2 data are fed into data processing routines running in MATLAB with implemented Java algorithms for a better performance. These data processing routines map the flights of DDR2 and ADS-B data, compile discrete waypoint-probability values and generate data-sets for training and testing the neural networks. The neural networks are implemented in Python and TensorFlow. The operational process of the TP module differs from the data-flow and -processing during training, see Figure 3. With the initially filed flight plan the neural network is fed to obtain the waypoint probability grid. Afterwards these probability values are processed in a MATLAB-module with embedded Java-algorithms to calculate a concrete predicted future position of the aircraft. This step needs the actual aircraft state as input (e.g. taken from received ADS-B data).

[2] The ADSbDataParser is used within AISA in the framework of Background IPRs.

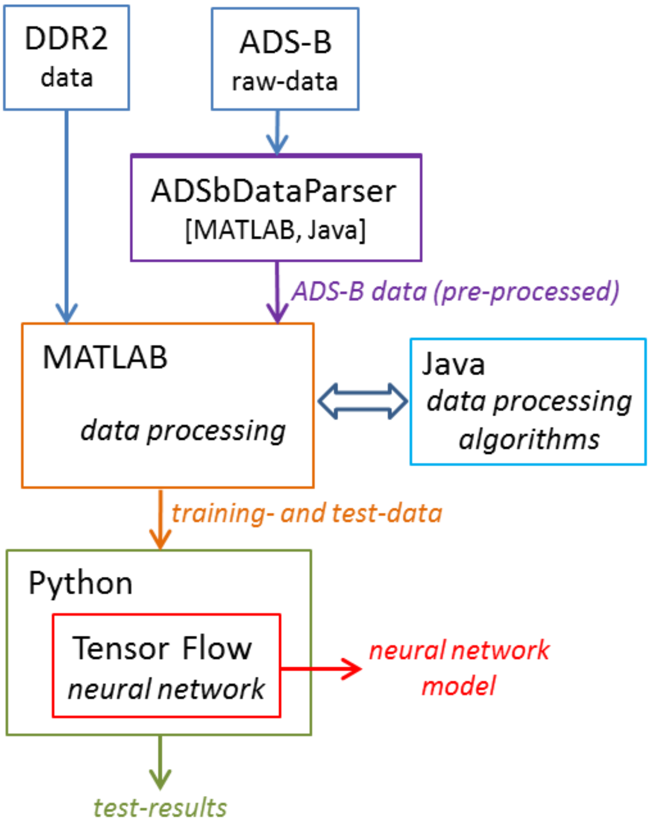


Figure 2: Data flow for neural network training

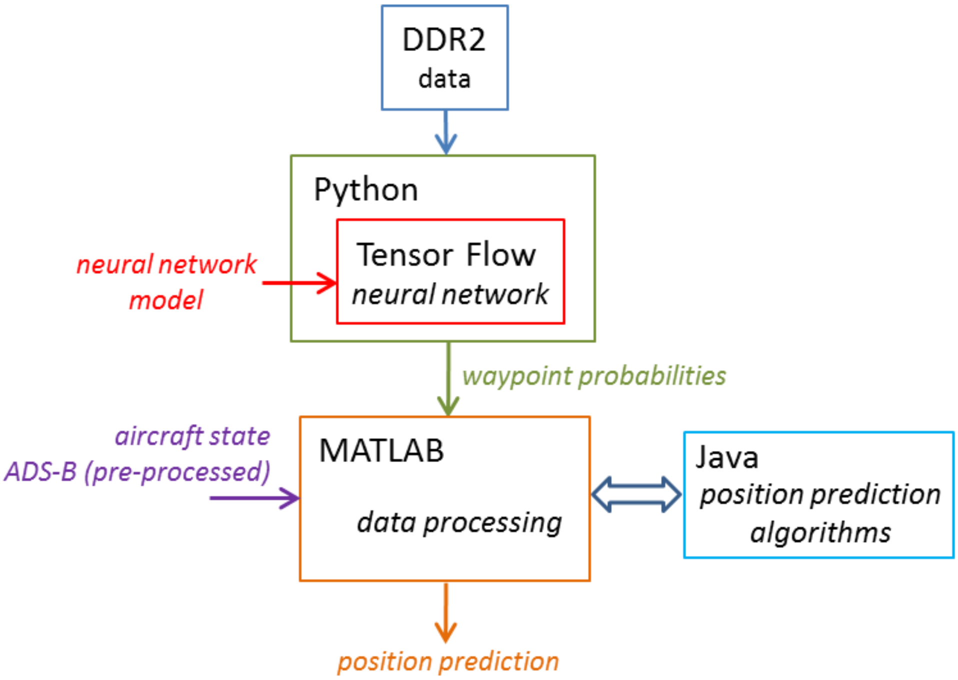


Figure 3: Data flow of operational trajectory prediction with actual aircraft state as input

3 ADS-B data gathering, pre-processing and statistics

The ADS-B data used for the TP module is gathered from the historical database of The OpenSky Network [10]. It is accessed via an Impala Shell SSH connection and various data tables of collected ADS-B messages can be requested. For the development of TP module, decoded ADS-B data of table state_vectors_data4 is going to be used. The structure of this data table is shown in Table 3-1.

Log-files of the SSH connection of this data format are directly processed by the ADS-B data parser. Thereby each aircraft needs to be stored separately within an individual SSH log-file and the data parser achieves best results, when the whole trajectory of a flight is available. This means that for each gathered trajectory a precise request query with adequate start and end timestamps is required. General request-timeframes covering the whole AIRAC cycle would not be precise enough, as callsigns may be re-used for various flights and therefore multiple flights could be concatenated in one request. This concatenation of multiple flights in one log-file currently can't be handled by the data parser without any errors.

The list of callsigns and corresponding timeframes (departure and arrival) is derived from initial DDR2 flight plans (model 1) [9]. Due to departure- and arrival-delays or early power-on of ADS-B transmissions previous to departure, effective timeframes of ADS-B and DDR2 data may vary. Hence a small study about standard deviation was conducted for flights crossing the sector *LSAZM567* used in the AISA project on 20.06.2019.

To gather the trajectories of this test dataset, an offset of four hours previous to departure and four hours after arrival regarding the DDR2 data was used. It was assumed and confirmed later on, that this offset of four hours is sufficient to comply with most of the flights to be within the timeframe. After gathering the test-data, it is processed by the ADS-B data parser and finally the standard deviation of arrival- and departure-time between DDR2 data (model 1) and the parsed ADS-B trajectories is analysed. Regarding departure a stochastic expectancy value of $\mu = 22.54\,min.$ and a standard deviation of $\sigma = 52.42\,min.$ was derived from the initial test-dataset. For arrival the expectancy value is $\mu = 4.87\,min.$ and the standard deviation is $\sigma = 43.03\,min.$ Hence with a probability of $99.73\,\%$ the ADS-B data will be within a timeframe offset of $(22.54 - 3 \cdot 52.42)\,min. = -134.72\,min.$ for departure and $(4.87 + 3 \cdot 43.03)\,min. = +133.96\,min.$ for arrival. This confirms the initial temporary timeframe offset of four hours for this analysis and the new timeframe offset $[-134.72\,min.;\ +133.96\,min.]$ with a confidence interval of $99.73\,\%$ was used to gather all ADS-B trajectories available in the AIRAC cycle 1907, which cross the chosen sector (*LSAZM567*).

name	type	description
time	int	UTC timestamp of sample
icao24	string	ICAO24 identifier
lat	double	latitude position
lon	double	longitude position
velocity	double	speed over ground
heading	double	track angle
vertrate	doubler	vertical rate in m/sec

callsign	string	callsign
onground	boolean	broadcasted indicator of surface/ airborne position
alert	boolean	alert indicator
spi	boolean	spi indicator
squawk	string	transponder code
baroaltitude	double	barometric altitude
geoaltitude	double	geometric altitude
lastposupdate	double	UTC timestamp of the samples position data
lastcontact	double	UTC timestamp of last received signal from aircraft
hour	int	UTC timestamp indicating the beginning of the hour to which the data belongs

Table 3-1: Data table *state_vectors_data4 (see* [10]*)*

Flights of the 20.06.2019 are used for the test-dataset, flights of all other days are used for training the TP module. 24410 flight plans were available from DDR2 database for the training-dataset and 807 flight plans for the test-dataset. After gathering and parsing the ADS-B data with the described process, 23419 trajectories were obtained for the training-dataset and 771 trajectories for the test-dataset. This indicates a loss of less than 5 % of parsed ADS-B trajectories compared to initially filed flight plans available from DDR2 database. Reasons may be aircraft filing a flight plan, but not being equipped with ADS-B, cancelled flights and gaps in received ADS-B data. Furthermore the ADS-B data parser automatically filters ADS-B data with too low resolution to build a sufficient trajectory. During the data processing step for generating training and test data suitable for the neural networks, some additional trajectories may be filtered out because of low data quality. The general quality of parsed trajectories can be described by the metrics reliability, completeness and plausibility calculated by the data parser. The reliability-metric indicates the mean reliability of the overall trajectory in the context of density of supporting samples contained in the raw ADS-B data: trajectory-parts with gaps (no received ADS-B data) are rated with a low reliability. The completeness-metric indicates whether the beginning and end of the trajectory are near a potential departure and arrival airport. The plausibility-metric describes whether the profile of barometric altitude is typical for a passenger aircraft. Histograms of the metrics of the gathered training-dataset are shown in Figure 4 and histograms of the metrics of test-dataset are shown in Figure 5. Table 3-2 lists the numbers of trajectories per aircraft type in training and test datasets.

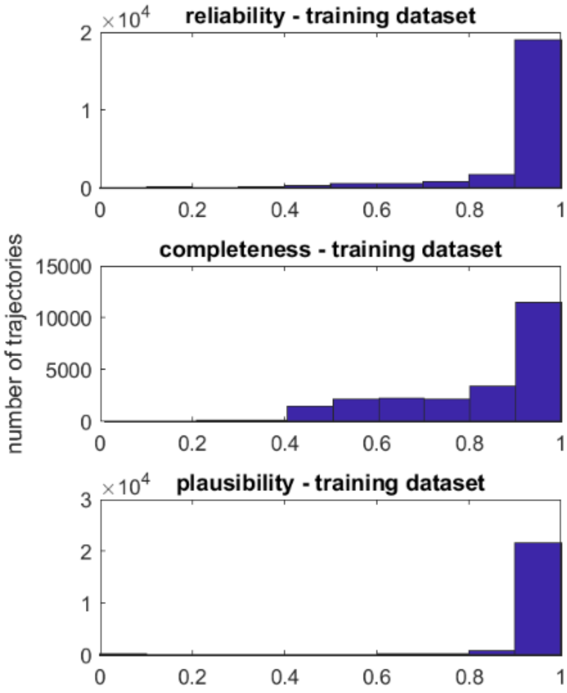


Figure 4: ADS-B data metrics of training dataset

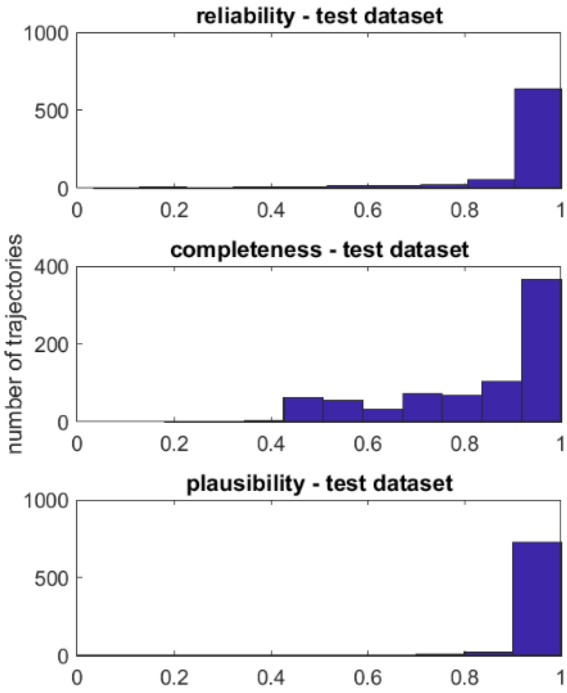


Figure 5: ADS-B data metrics of test dataset

Type	TRNG	Test	Type	TRNG	Test	Type	TRNG	Test	Type	TRNG	Test
B738	7855	(255)	**E75S**	73	(2)	**E75L**	27	(1)	**PC24**	5	(0)
A320	4648	(149)	**CL60**	72	(2)	**E195**	23	(0)	**C55B**	5	(0)
A319	2672	(97)	**CL35**	71	(5)	**CRJX**	21	(1)	**GL7T**	4	(0)
A321	1399	(38)	**B736**	70	(0)	**A35K**	20	(1)	**A310**	4	(0)
A20N	970	(28)	**B733**	70	(4)	**C525**	19	(0)	**CRJ2**	4	(0)
B737	403	(13)	**GLF6**	70	(1)	**FA8X**	18	(1)	**C560**	4	(0)
E190	381	(12)	**BCS3**	66	(4)	**C510**	17	(2)	**BCS1**	4	(0)
A332	316	(9)	**E35L**	62	(2)	**BE40**	15	(0)	**GA5C**	4	(0)
B752	286	(18)	**FA7X**	53	(2)	**B735**	15	(2)	**E50P**	4	(0)
B789	276	(11)	**C25A**	49	(1)	**GALX**	14	(0)	**C650**	3	(0)
B788	269	(8)	**C68A**	46	(1)	**C750**	13	(0)	**FA10**	3	(0)
A333	257	(6)	**GLF4**	45	(1)	**LJ75**	13	(1)	**G280**	3	(0)
A21N	247	(8)	**GL5T**	40	(1)	**A339**	12	(0)	**B764**	3	(1)
B763	201	(4)	**F900**	40	(0)	**C25B**	12	(2)	**E545**	3	(0)
A318	194	(8)	**B762**	37	(2)	**E170**	11	(0)	**LJ40**	2	(0)
A388	187	(5)	**A343**	37	(2)	**E550**	11	(0)	**B78X**	2	(0)
B753	138	(5)	**B77L**	34	(2)	**C25M**	11	(0)	**P180**	2	(1)
B77W	131	(1)	**CL30**	33	(1)	**PRM1**	10	(0)	**A342**	2	(0)
GLEX	126	(4)	**B748**	33	(2)	**H25B**	9	(0)	**C550**	2	(0)
F2TH	125	(6)	**C680**	32	(2)	**HDJT**	8	(0)	**ASTR**	1	(0)
E55P	119	(8)	**A346**	32	(1)	**G150**	7	(0)	**LJ55**	1	(1)
GLF5	104	(1)	**LJ45**	32	(3)	**FA50**	7	(0)	**C551**	1	(0)
A359	103	(4)	**C25C**	32	(0)	**MD11**	7	(1)	**C700**	1	(0)
B772	95	(5)	**CRJ9**	32	(0)	**A306**	7	(0)			
C56X	92	(2)	**B739**	30	(1)	**LJ35**	6	(0)			
B744	80	(4)	**B734**	29	(1)	**E135**	6	(0)			

Table 3-2: List of aircraft types of training and test dataset

4 Development of Neural Networks

The development of ML-TP module assesses fully connected feed-forward neural networks. Thereby adequate inputs and outputs are required to address a certain context contained in the training- and test-data. This functional context is to be modelled by the neural network. The general TP-idea of chapter 2 shall be part of this modelling-context. Due to the complexity of the envisaged TP-problem – with time-domain it's a 4D problem – it needs to be simplified to be fed to and modelled by a simple feed-forward network. This simplification is done by neglecting the vertical plane and the temporal dimension in the first step. According to the problem definition (see chapter 2) the input of the neural network shall be the initially filed flight plan and the output shall contain information about the actually flown track of the aircraft. These inputs and outputs need to be designed properly to provide the most expressive data to the neural network during training to model the desired functional context. For each potential waypoint of the considered initial flight plans, an individual input-value will be foreseen, which indicates whether the waypoint is part of the actual inputted flight plan or not. Therefore the input-values can take two values: 0 in case the waypoint is not part of the flight plan and 1 when it is. This enables a simple way of using the neural network, as only a vector of $[0,1]$ values regarding the actual flight-plan is required to feed the neural network in operation. To limit the number of required input-values on the one hand, but to not lose essential input-information regarding the functional context to be modelled, the considered waypoints for the input vector are determined with an iteratively set radius of $400\ NM$ around the airspace to be used. This means, that all official waypoints spatially laying within this $400\ NM$ radius-area are part of the input-vector with one dedicated value being 0 or 1. To gather the list of these considered waypoints the DDR2 flight plans of training and test datasets were consulted. After all a total number of 2135 waypoints were collected. Airports and flight-individual technical waypoints, such as top of climb, etc., were not considered for the input of the neural network. This is because departure and arrival airports may lie out of the $400\ NM$ area and technical waypoints would further increase the already huge input-vector to the neural network.

Contrary to the spatially differentiated DDR2-waypoints, the considered output-information to be modelled, the ADS-B track, is much more detailed and individual, as no commonly used official positions apply here, like with the waypoints of the flight plans. To obtain an adequate representation of this desired output-information within a comparable scale of number of values regarding the input-vector and by keeping or even sharpen the contextual functionality (route directs and changes between flight plan and final track) to be modelled, the individual positions of ADS-B samples will be mapped to a static list of predefined waypoints. This is done by calculating a waypoint-overfly probability for each of the static predefined waypoints regarding the finally flown trajectory represented by the ADS-B data. The waypoint-overfly probability value δ assesses whether the aircraft is passing or overflying the dedicated waypoint within a $10\ NM$ range. It is calculated for each trajectory i and waypoint j with the pass-distance $d_{pass,i,j}$:

$$\delta_{j,i} = \begin{cases} 0 & if\ d_{pass,i,j} \geq 10\ NM \\ 1 - \dfrac{d_{pass,i,j}}{10\ NM} & else \end{cases}$$

The probability values range from 0 to 1. A graphical example is shown in Figure 6. The blueish markers represent the waypoint probability values. These markers are also used to graphically represent the outputs of the designed neural networks. During the development and test of various neural networks, described in the following chapters, unavoidable shortcomings were observed, which appear in lower waypoint probabilities during testing (and operation) of the networks. These lower probability values result in low saturations within graphical representations. Therefore the outputs of neural networks are post-processed by applying the square-root and two thresholds:

$$\bar{\delta}_{j,i} = \begin{cases} 0 & if \ \sqrt{\delta_{i,j}} < 0.2 \\ 1 & if \ \sqrt{\delta_{i,j}} \geq 0.4 \\ \sqrt{\delta_{i,j}} & \quad else \end{cases}$$

Figure 6 depicts a comparison between the original and the post-processed probability values. This post-processing of output-values is only conducted for graphical representations. For further processing of the data, e.g. the position prediction in chapter 5, no such rework of the values is carried out.

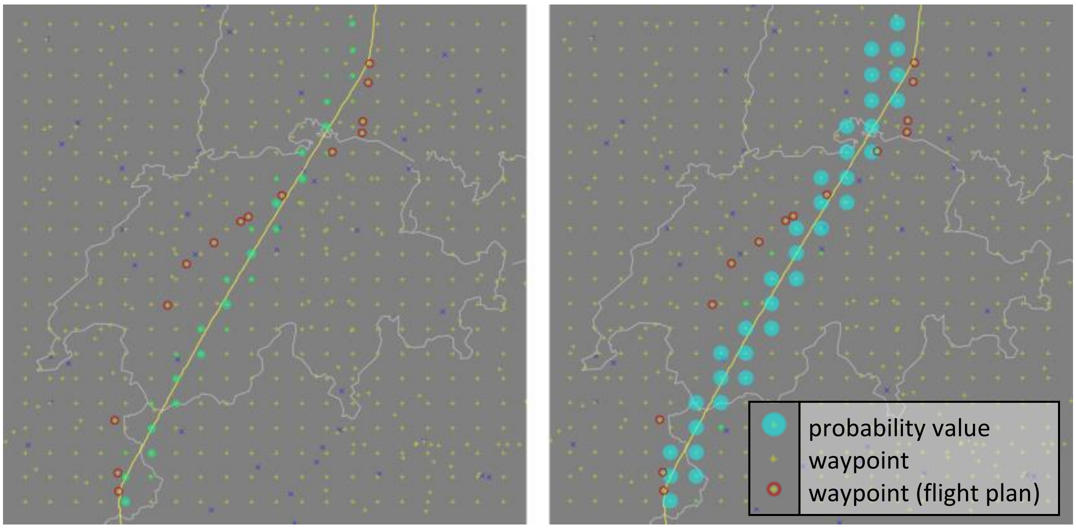


Figure 6: Comparison of rendering of neural networks waypoint probability-values (left: without post-processing, right: with post-processing)

4.1 Neural Network with official waypoints output

The first approach to predict re-routings and route-changes on flight plan level was to use the official waypoints lying within the considered geographical area for input values as well as for the output values. Hence the vector-length of in- and output and the corresponding waypoints are equal. The only difference is the kind of value calculation for in- and outputs (see chapter 4). The idea is to identify skipped waypoints of the initial flight plan and to indicate other waypoints, which have been overflown but are not part of the filed flight plan. This can be observed in the example in Figure 7. It depicts the output of a neural network with one hidden layer with 100 sigmoid neurons, which was trained with the Adagrd-algorithm and the mean absolute error as error-function. 40 epochs of the training dataset have been trained with a batch-size of 1. The final mean absolute error of the trained network was 0.0158 regarding the training dataset and it is the best result achieved in all network- and training-developments iteratively conducted with official waypoints as in- and output.

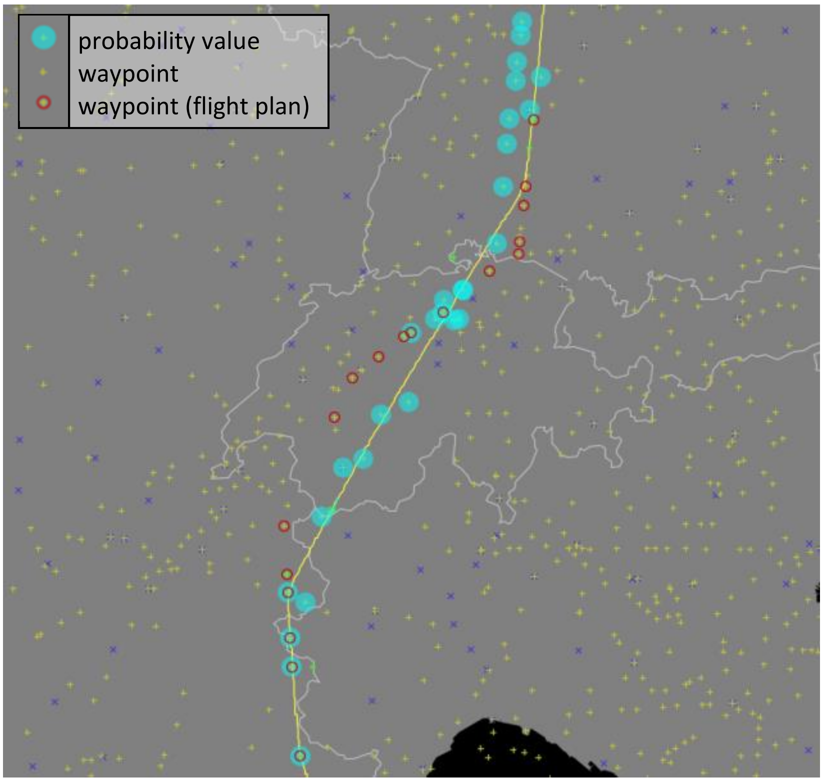


Figure 7: Example of waypoint probability with flight plan- waypoints as neural network output

During the assessment of the results of the test-dataset some trajectories with major errors in the prediction of the neural network were observed. These errors comprise generally false probability-values (an example is shown in Figure 8) and missing information or gaps in some geographical regions (see Figure 9). Even though the neural network would perfectly model the true track of the aircraft, the unequal distribution of flight plan waypoints (which are used for the output-values) has the effect of inhomogeneous prediction precisions: In regions with a low density of considered waypoints, the prediction precision is generally lower, especially if additional errors in the modelling of the neural network occur. The general idea of the approach was to identify single waypoints of the initial flight plan, which have been skipped during the flight. But the aforementioned shortcomings make it much more difficult or even impossible to derive such information from the output of the neural network.

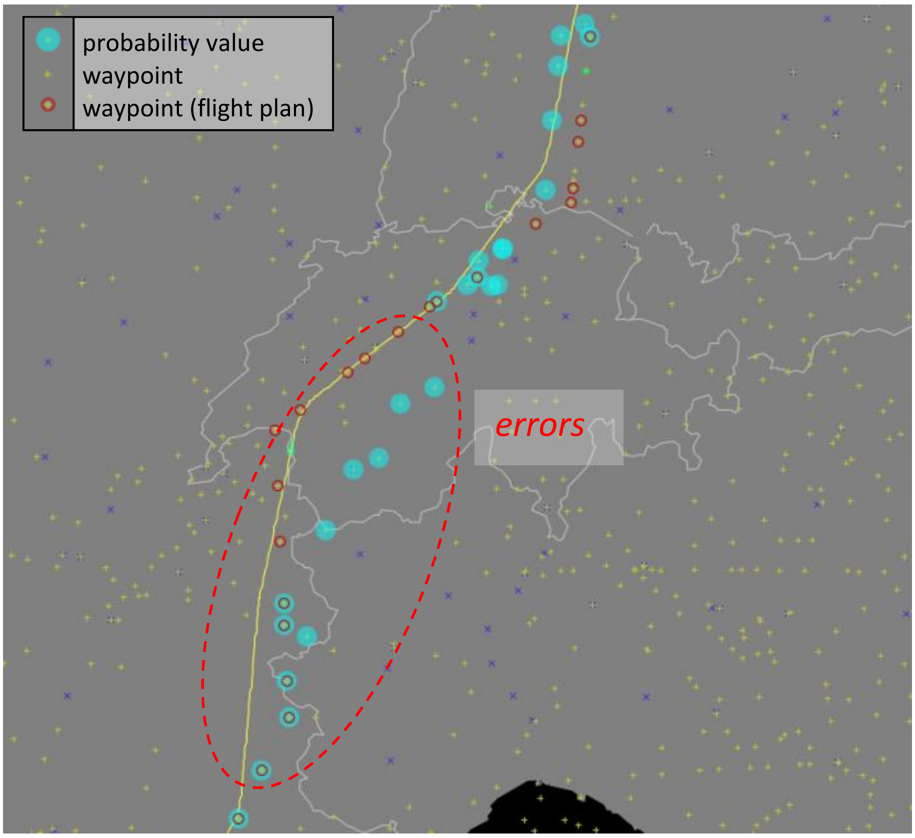


Figure 8: Example of erroneous waypoint probability predictions (flight plan waypoints used for output-values)

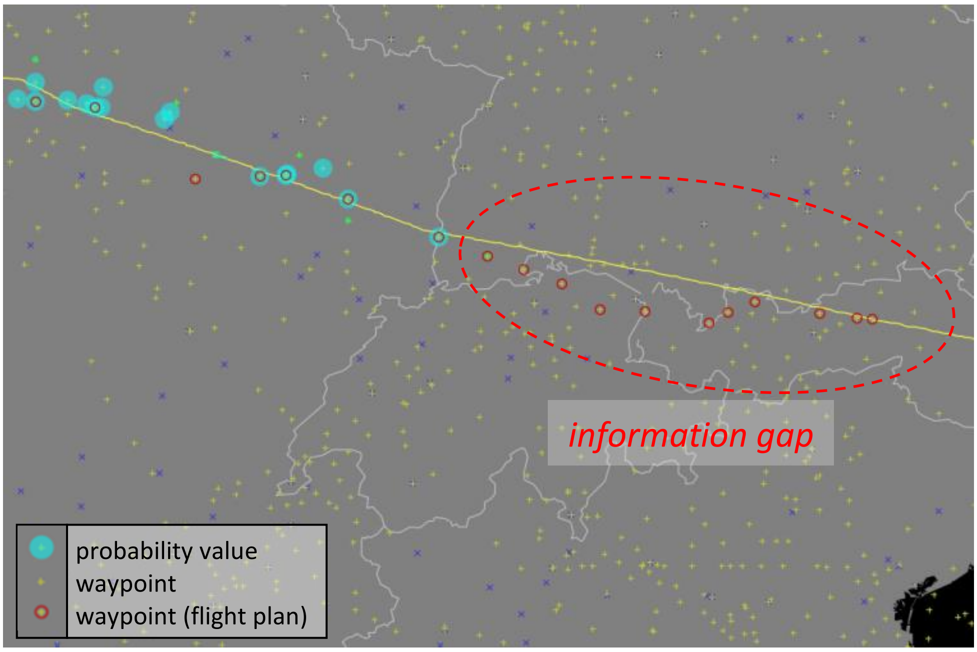


Figure 9: Example of a flight with a region with a gap in the waypoint probability prediction (flight plan waypoints used for output-values)

4.2 Neural Network with waypoints-grid output

To improve the waypoint probability prediction, especially regarding the precision of prediction resulting from the distribution of the output-waypoints, a generic waypoint grid with homogeneous distribution was tested as output of the neural network. This grid was spread over the considered airspace of Switzerland with an approximately waypoint-distance of $10\ NM$, resulting in a grid of $20x20$ waypoints. The waypoint-distance of $10\ NM$ is derived from the waypoint probability calculation introduced in chapter 4. If the true track is passing the middle of two grid-waypoints, each waypoint then will take a probability value of 0.5. A network with one hidden layer composed of 100 sigmoid neurons was observed to show the best results regarding modelling accuracy and training- and operational calculation effort. The input-vector is the same like the neural network in chapter 4.1 (2135 waypoint values, range: $[0;1]$). The output vector will be composed of the 400 grid-waypoint probability values. An example with adequate results regarding the modelling of route-changes is depicted in Figure 10. The waypoint-probabilities, which are calculated by the neural network with the flight plan as input, neatly follow the true track of the aircraft.

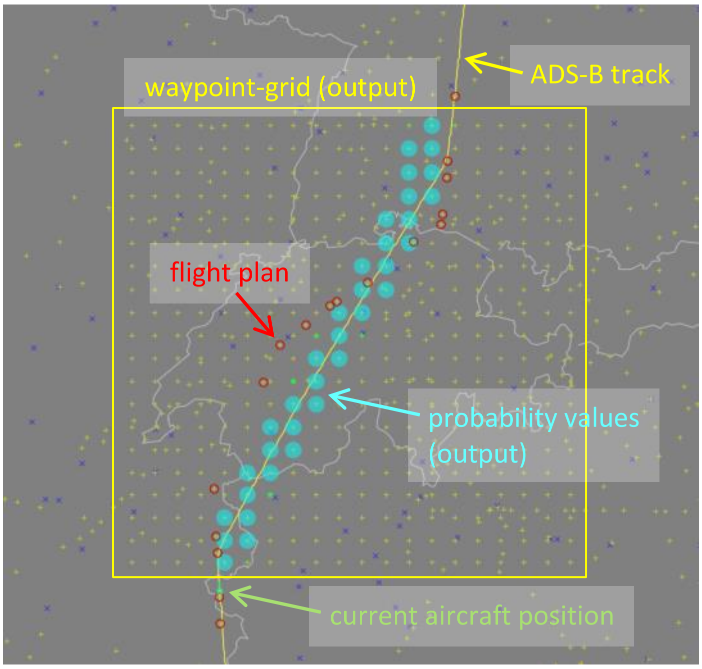


Figure 10: Example of track prediction with neural network (waypoint-grid used for output values)

The shortcomings observed in the results of the neural network of chapter 4.1 also appear in the results of this neural network, but with lower severity. It is assumed, that the smaller geographical region and higher density of output-waypoints facilitates a more detailed modelling of the functional correlations between the flight plans and true tracks. Beneath the lower informational complexity to model by the neural network, this can also be explained with the composition of the training and test datasets. These have been composed with the trajectories crossing the considered airspace above Switzerland, which results in a higher informational density in this region, than in the regions around (see test dataset in Figure 11 as an example).

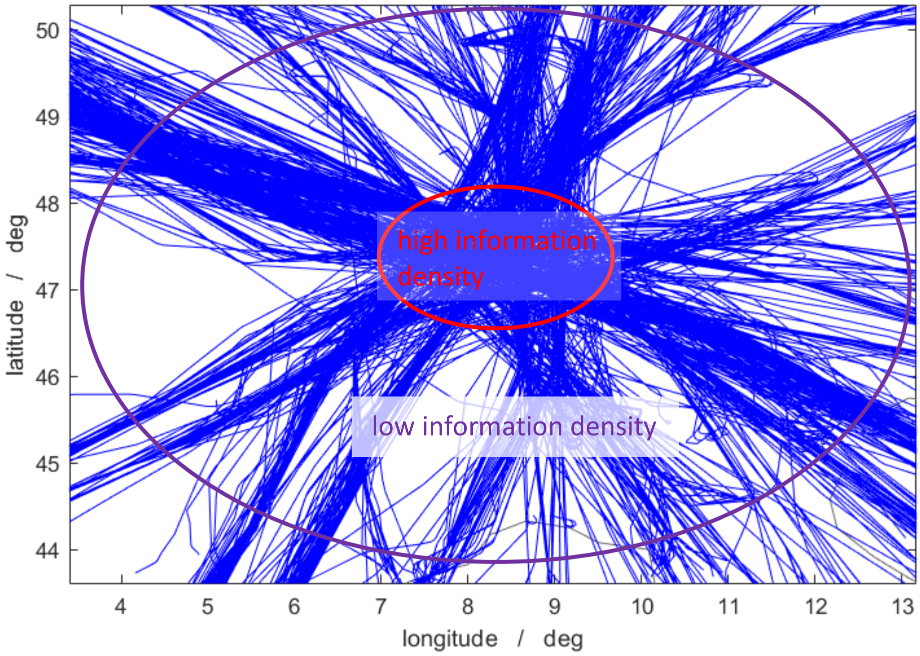


Figure 11: Trajectories of test dataset

4.3 Neural Network with waypoints-grid output and additional input data

The neural network to predict aircraft-tracks in chapter 4.2 is only using static and discrete position data without any temporal information. Also the direction of the trajectories is not explicitly indicated within the input-vector, because the waypoints are only marked as part of the flight plan with no information about the sequential order. Additional tests with adapted neural networks have been conducted to test, whether the provision of temporal information and information about the direction of the flights have beneficial effects on the results of the neural network. Thereby the output-vector and probability-value calculation of chapter 4.2 are kept, which facilitates a comparison of the neural networks with the mean absolute error of the output during training. To add the temporal information to the input-vector, each waypoint is assigned a second value, indicating the time of the day of passing the dedicated waypoint according to the flight plan. This timestamp-value ranges from 0 to 1 according to $12\,am$ and $12\,pm$. The timestamp-values of all waypoints not part of the flight plan also take 0 as a value. In the second approach the mean course $\overline{\Psi}$ of the flights was added as a single scalar value to the input-vector of the neural network. This value ranges from 0 to 2π according to the radian angle (0: north, π: south). It is calculated with the starting and ending course on a virtual orthodrome between the departure and arrival position:

$$\overline{\Psi} = \frac{\Psi_{arr\_dep} + \Psi_{dep\_arr}}{2}$$

The final mean absolute error of all trained networks is shown in Table 4-1. Although the mean absolute error of the neural network with the timestamp-data is slightly smaller, the difference is within the scope of the stochastic processes during initialization and training of the neural networks. After all, the additional input values had no relevant beneficial effect on the training of the neural network. Therefore the neural network with no additional input than the flight plan will be used in the AISA project.

network input	mean absolute error
no additional input (only flight plan)	0.0333
with waypoint timestamps (for each input-waypoint)	0.0332
with departure-arrival course (only one scalar value)	0.0339

Table 4-1: Final mean absolute errors of trained neural networks

5 Position prediction with current aircraft state

The neural network developed in chapter 4.2 is modelling the aircraft-track without any information in the time-domain. To predict the future position as a function of the current aircraft state during the flight, the waypoint probability grid is combined with an aircraft-fixed pattern (see Figure 12). This pattern indicates the possible area of the future position of the aircraft in a distance of $100\,NM$. The lower radius is set to $80\,NM$ and the higher radius is set to $120\,NM$, according to a consideration-width of $40\,NM$. To respect potential turns in the future track, the pattern has an opening-angle of $110\,°$. The predicted future position is calculated by the weighted mean of all waypoint probability-values laying in the aircraft pattern. An example is shown in Figure 13. Besides the advantage of providing a concrete position prediction, the calculation of the weighted mean is also smoothing the granular track prediction of the neural network and therefore increasing the precision of the position prediction.

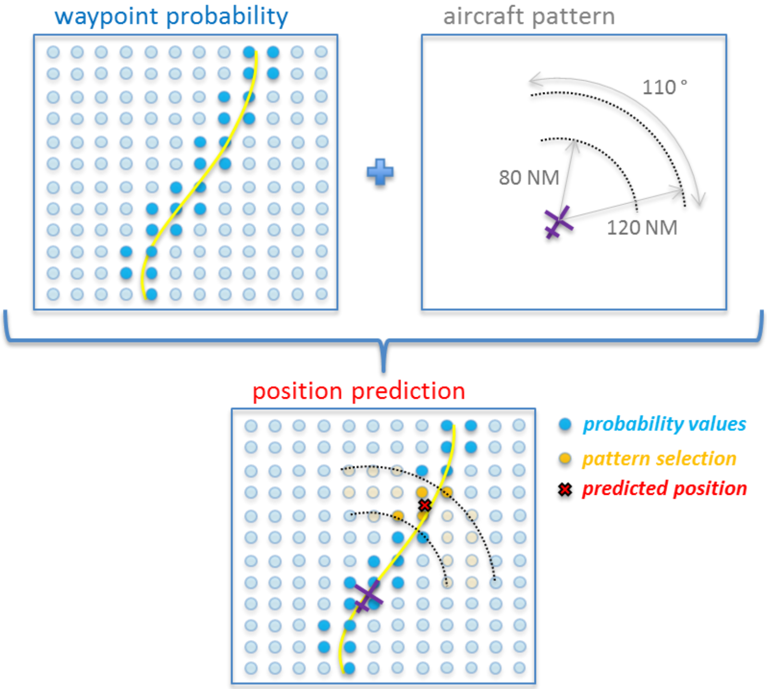


Figure 12: Position prediction with grid-waypoint probabilities and aircraft pattern

Because of the fixed distance of $100\,NM$ of the middle of the aircraft pattern, the predicted position is assumed to be a distant-fixed prediction without any dependence on the aircrafts speed and with no temporal information. A timestamp of the predicted position may be calculated with the distance and the aircrafts speed. Thereby the duration till the predicted position may vary, depending on the concrete position prediction with the weighted mean and especially depending on the speed of the aircraft. Tests with the ground-speed obtained from the ADS-B track indicated severe noise in the calculation of prediction-timestamps. Although the ADS-B track was partly already softened with the

ADS-B data parser, some trajectories comprise noisy spatiotemporal samples, which result in noisy ground-speed calculations and therefore noisy prediction-timestamps.

After calculating the predicted future position with the weighted mean of probability values, the highest probability value in the vicinity of the predicted position with a maximal distance of $40\ NM$ is selected as a metric for the reliability of the predicted position itself. Also an average-calculation of all probability values with a maximal distance of $40\ NM$ was assessed for this metric. But in this case undesirable effects would be rewarded in the metric: a precise track-prediction of the neural network with a narrow track in the waypoint-probabilities would dilute the reliability-metric, whereas a broad track-prediction would erroneously be rewarded with the average calculation. Because of the relative large consideration area with a radius of $40\ NM$ the metric can only be used as a first approach regarding the reliability of the prediction.

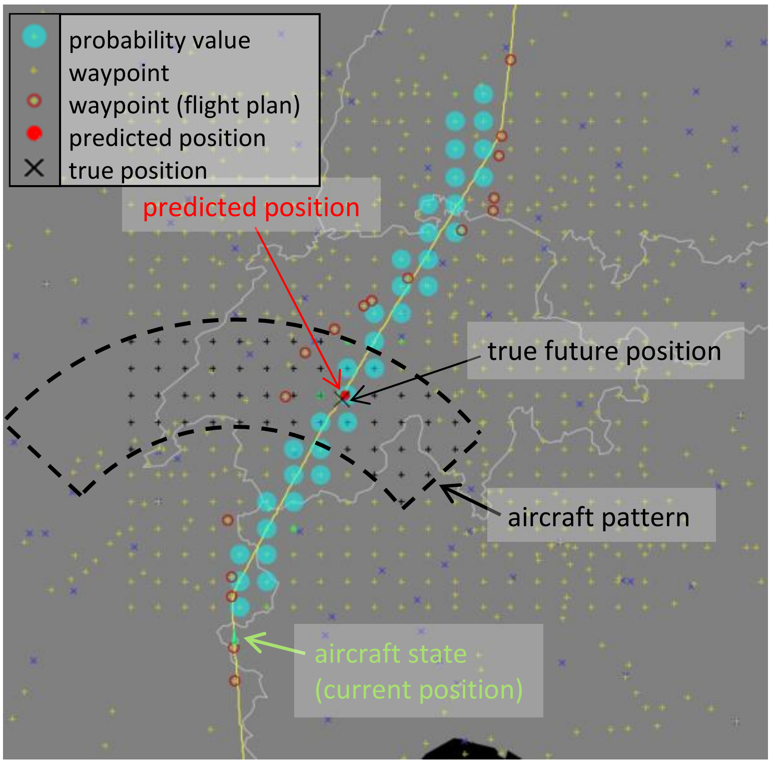


Figure 13: Example of position prediction with grid-waypoint probabilities and aircraft pattern

A predicted aircraft altitude is obtained in a separate process by mapping the planned waypoint-altitudes of the flight plan to the predicted position and by correcting this value with the latest known error-offset of the altitude prediction. The mapping of the planned flight plans altitudes to the previously predicted position is done by calculating the distance-weighted average of the altitudes of the two flight plan waypoints lying nearest to the predicted position. Afterwards the obtained altitude is corrected with the latest known error-offset. The error-offset is defined by the difference between the planned altitude and the finally flown altitude, when passing the timestamp of the predicted position. This means, that the error-offset is earliest available when the aircraft passes the predicted position timestamp itself. Therefore the latest known error-offset is used to correct the predicted position. This procedure is depicted in Figure 14.

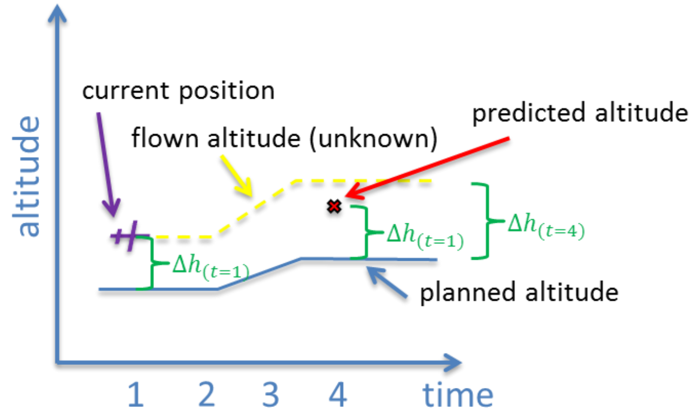


Figure 14: Altitude prediction with correction offsets Δh

6 Validation of final Trajectory Prediction

The results of the trajectory prediction of the combination of the neural network of chapter 4.2 with the position prediction method in chapter 5 are assessed. To obtain a dataset with predicted trajectory positions, the positions of all trajectories of the test-dataset are predicted for every 30 second. In case the aircraft pattern lies beyond the waypoints-grid or all probability values of the selected waypoints of the waypoints-grid are zero, no position prediction is possible. These cases are not further tracked in the validation-dataset. Figure 15 depicts the distribution of reliability-metrics of the position predictions of this dataset. About 6000 position predictions have a reliability value of less than 0.1. Only a low number of predicted positions has a value between 0.1 and 0.4, whereas the number of predicted positions is increasing with higher reliability values than 0.4. Figure 16 depicts a histogram of the correctness of all predicted positions with a reliability-metric greater or equal to 0.4 and Figure 17 summarizes the distribution of deviations of predicted positions with a reliability-metric lower or equal to 0.1. Both histograms underline the idea of the reliability-metric: The distribution of deviations with a high reliability-value have a maximum at 0 to $2.7\ NM$, whereas predicted positions with low reliability-values in general comprise higher deviations to the true future position (maximum at 65 to $75\ NM$). Nevertheless do the histograms also show, that the introduced reliability-metric is only a first approach to the reliability of the predicted position, like stated in chapter 5, as there are also some position predictions with a high reliability-value but high position deviation and vice versa.
The predicted altitudes have a mean absolute deviation of $1228\ ft.$ to the finally flown altitude levels. The histogram in Figure 18 indicates that most predicted altitudes have a deviation near to zero and some predicted altitudes comprise major deviations of thousands of feet, resulting in the relatively high mean absolute deviation.

Incorrect future position predictions generally can have two main sources. Either the predicted track of the neural network may be inadequate or the position prediction method with the aircraft pattern may be erroneous. Although the track prediction with the neural network could have any shape as output, the plotted waypoint probability values generally have the shape of a straight corridor, which is the desired output to predict the track. Besides general differences in the predicted track and the true track, especially two different kinds of failures have been observed. The first modelling-failure comprises misleading predictions with no route changes regarding the flight plan, whereby finally route changes have been applied during the flight (Figure 19). The second failure category comprises predictions with route changes by the neural network, but no re-routings in the final flight trajectory (Figure 20). Figure 21 depicts an example of a predicted aircraft track, which adequately indicates route changes regarding the initial flight plan. Prediction errors within the aircraft pattern method especially arise in the border areas of the waypoints-grid. In situations where the true future position of the aircraft is slightly outside the grid, only parts of the aircraft pattern may cover the grid-area, see Figure 22 and Figure 23. Then also low waypoint probability-values may strongly impair the predicted future position by the weighted mean calculation. It is assumed, that especially this effect is responsible for the large position deviations shown in the histograms.

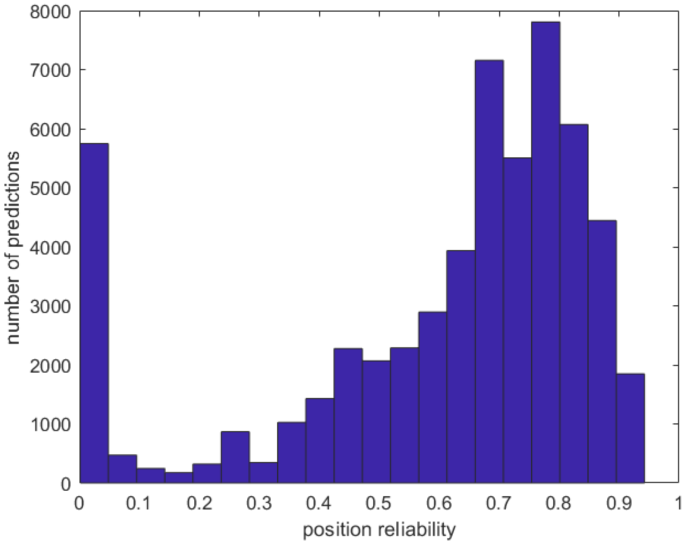


Figure 15: Histogram of reliability-metrics of position predictions (test-dataset)

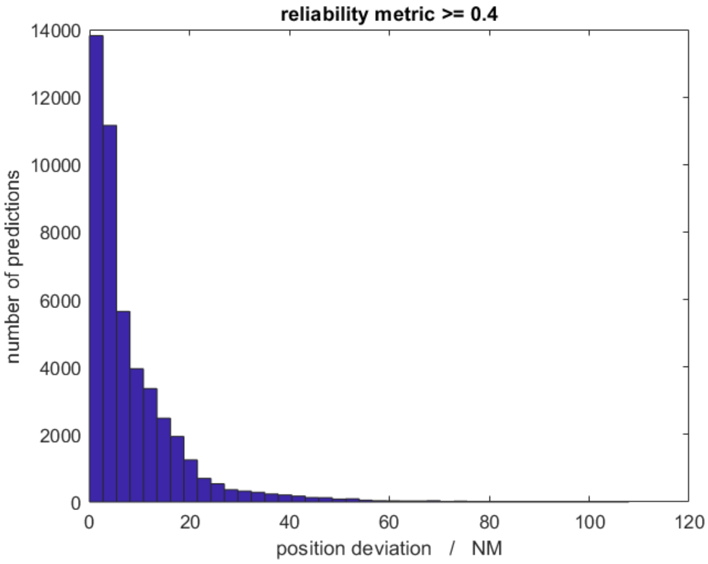


Figure 16: Histogram of predicted positions deviations (regarding true position) with a reliability value ≥ 0.4

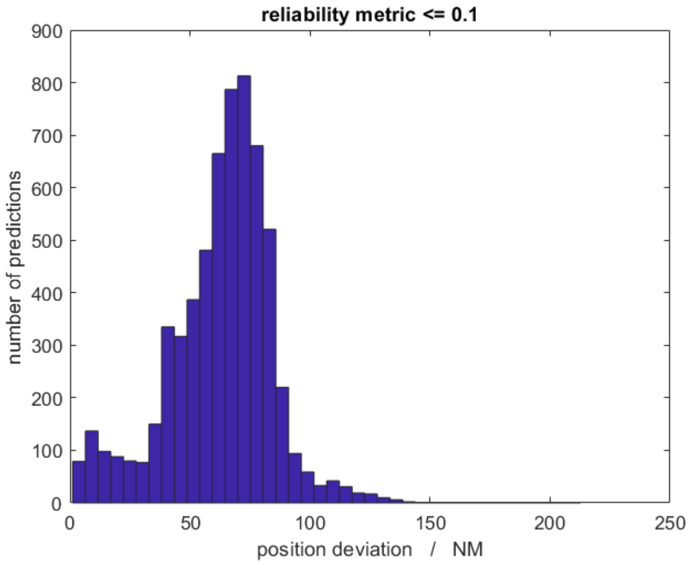


Figure 17: Histogram of predicted positions deviations (regarding true position) with a reliability value ≤ 0.1

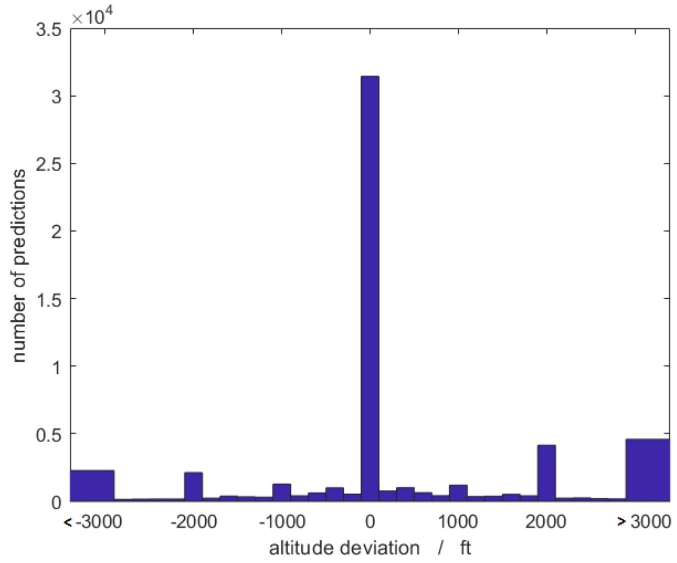


Figure 18: Histogram of predicted altitude deviations (regarding true altitude)

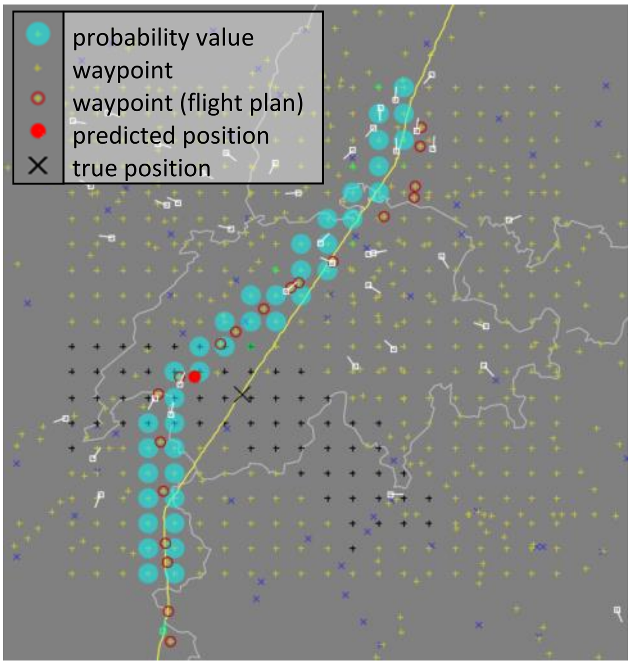


Figure 19: Track prediction failure (false negative route change)

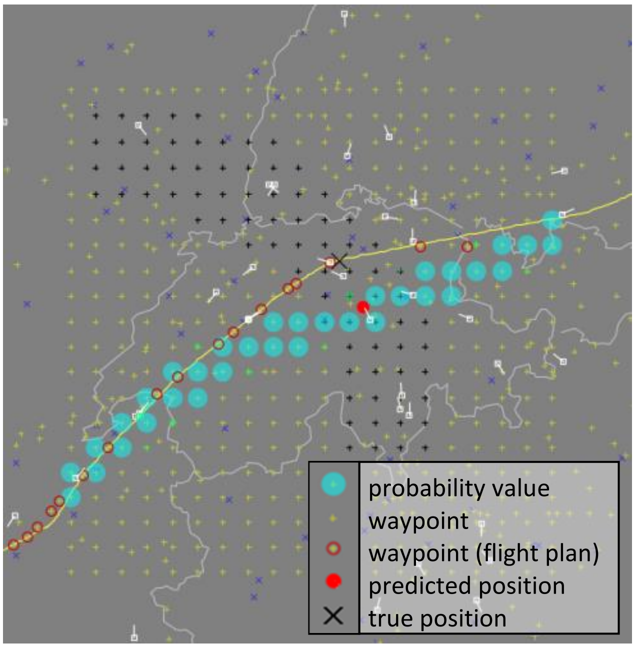


Figure 20: Track prediction failure (false positive route change)

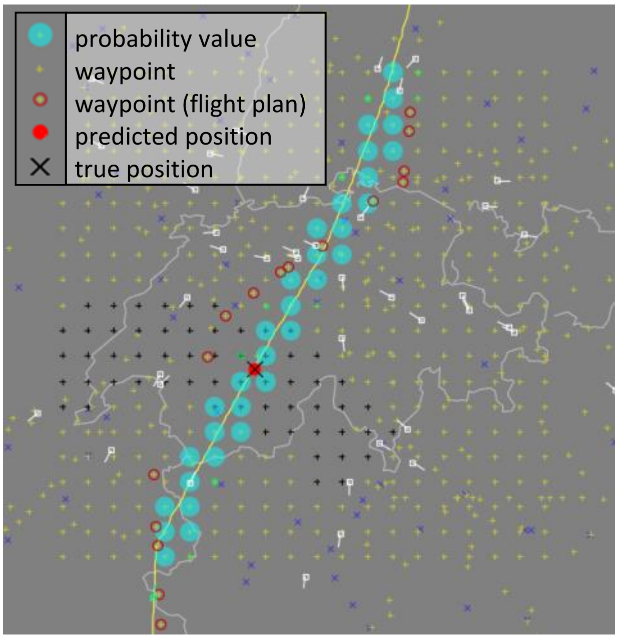


Figure 21: Adequate route change prediction

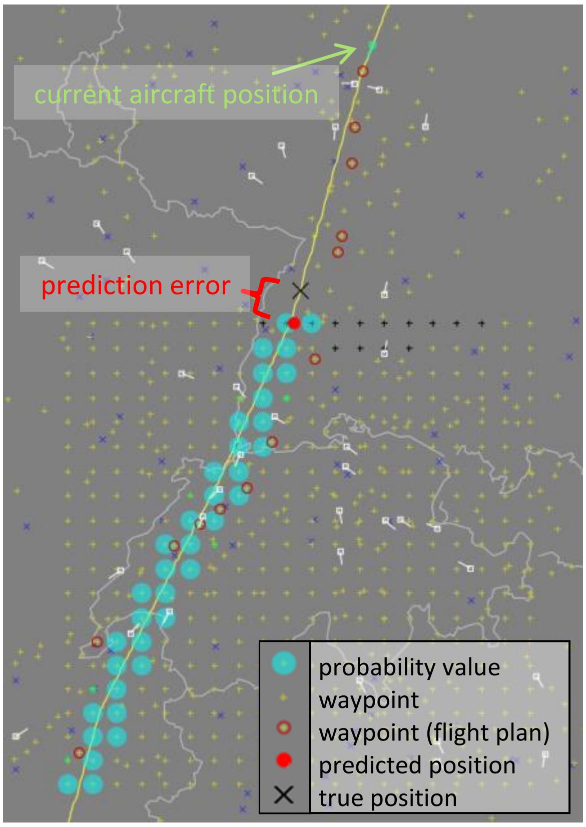


Figure 22: Position prediction error of entering aircraft reliability value: 0.84

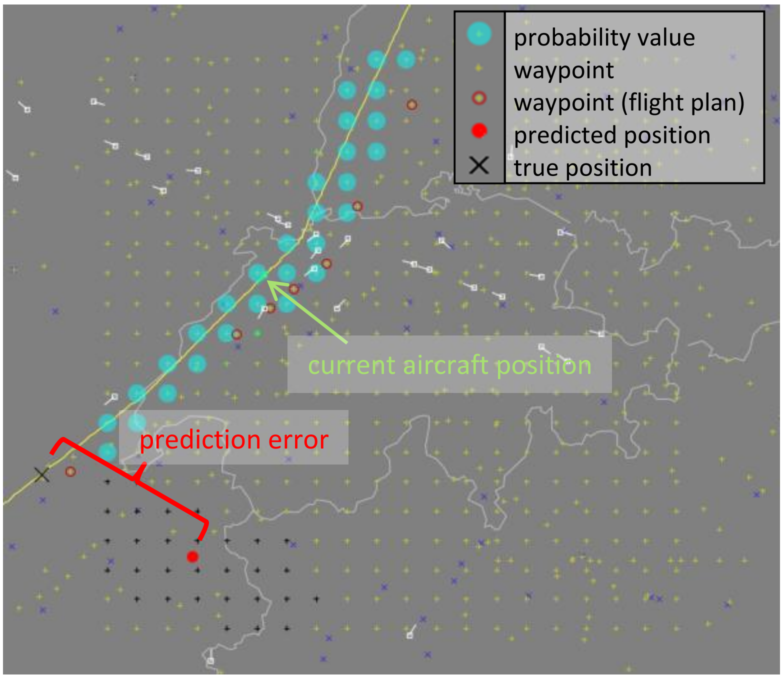


Figure 23: Position prediction error of exiting aircraft reliability value: 0.0027

7 Conclusion

A two-step trajectory prediction method has been developed. At first a neural network is predicting a granular static track of the flights. Thereby no temporal information about the track-position is provided and the filed flight plans comprise the input of the neural network. Because of the static approach with minimal requirements regarding the input, this step of the prediction can be conducted as soon as the initial flight plan is known and with any flight plan, which waypoints were parts of the input-vector during training. But it needs to be respected, that flight plans, which have similar trajectories in the training dataset, probably obtain much better prediction results than flight plans, which haven't been trained. During the development of the first trajectory prediction step, also timestamp information about the crossing of flight plan waypoints and the direction of the flights have been considered as input. But with the analysed settings no considerable benefits in the modelling of the aircraft tracks were observed. The flight plan waypoints used for the neural network input had also been considered for the output of the neural network. But it emerges, that a higher output waypoint density is required, to gather considerable information about the predicted track in the considered area. Therefore a waypoints-grid with $20x20$ waypoints was finally applied. As the dimension of the neural networks output-vector is connected to the waypoints-grid size, larger synthetic waypoint-grids regarding the number of waypoints would result in a larger neural network output-vector and therefore a more complex neural network itself.

In the second step of the trajectory prediction, the actual aircraft state in combination with an aircraft-fixed pattern is used to predict a concrete position in a distance of $100\ NM$ with the beforehand predicted aircraft track as input. As this step requires the actual aircraft position and track-angle as input, it is a dynamic part of the trajectory prediction and needs to be conducted during the operation of the flight. ADS-B data may be used to obtain the actual aircraft state during operation. If the actual ground-speed is known, also a timestamp of passing the predicted future position may be calculated. Despite previous filtering, the ground speed obtained from the ADS-B data turned out to be noisy for some aircraft. Therefore also the prediction timestamps of these flights would be noisy and the distance parameter of $100\ NM$ was used in the validation of the final trajectory prediction. The altitude is predicted by mapping the flight plans altitude levels to the predicted position and correcting it with the latest known prediction error.

Validation indicated the prediction reliability-metric, which is obtained in the second part of the trajectory prediction, to be an initial indicator about the reliability of the predicted positions. But it needs to be respected, that there may be position predictions with a high reliability-value but a high deviation regarding the true future position and vice versa. Although many predictions have a position deviation of less than $6\ NM$, a considerable amount of predictions have a deviation of more than $11\ NM$ regarding the true future position, especially for predictions with a reliability-metric lower than 0.1. These greater prediction deviations, which can grow up to $140\ NM$, presumably especially arise from prediction errors of the second part of the trajectory prediction in the border areas of the waypoints-grid. An assessment of the distance from the actual aircraft state to the predicted position may indicate deviations from the intended prediction-radius of $100\ NM$ and therefore it could probably be used to identify these prediction errors in the border-area of the waypoints-grid. This is where the upcoming research modules of AISA may take over to assess the results of the trajectory prediction and the other machine learning modules (conflict detection module and airspace complexity module).

8 Bibliography

[1] T. Radišić, M. Bazina, D. Antolović, I. Tukarić, R. Gurály, A. Kocsis, K. Moebus, I. Hajdinjak and T. Rogošić, "Concept of Operations for AI Situational Awareness," *AISA project,* 30 September 2020.

[2] R. Gurály, A. Kocsis, B. Gurály, K. Moebus, J. A. Perez Castan, B. Neumayr, D. Antolović, M. Bazina and T. Radišić, "Requirements for automation of monitoring tasks via AI SA," *AISA project,* 30 September 2020.

[3] J. A. Pérez Castán, "D3.2 Conflict detection module," *AISA project,* March 2021.

[4] I. Tukarić, "D3.3 Air traffic complexity module," *AISA project,* March 2021.

[5] R. Alligier, D. Gianazza and N. Durand, "Machine Learning and Mass Estimation Methods for Ground-Based Aircraft Climb Prediction," *IEEE Transactions on Intelligent Transportation Systems,* pp. 1-12, 30 July 2015.

[6] R. Alligier, D. Gianazza and N. Durand, "Machine Learning Applied to Airspeed Prediction During Climb," *ATM seminar 2015, 11th USA/EUROPE Air Traffic Management R&D Seminar,* 29 June 2015.

[7] Z. Wang, M. Liang and D. Delahaye, "Short-term 4D Trajectory Prediction Using Machine Learning Methods," *SID 2017, 7th SESAR Innovation Days,* 29 November 2017.

[8] Y. Liu and M. Hansen, "Predicting Aircraft Trajectories: A Deep Generative Convolutional Recurrent Neural Networks Approach".

[9] EUROCONTROL, "Demand data repository (DDR) | EUROCONTROL," [Online]. Available: https://www.eurocontrol.int/ddr. [Accessed March 2021].

[10] "The OpenSky Network - Free ADS-B and Mode S data for Research," [Online]. Available: https://opensky-network.org/. [Accessed March 2021].

[11] M. Schäfer, M. Strohmeier, V. Lenders, I. Martinovic and M. Wilhelm, "Bringing up OpenSky: A large-scale ADS-B sensor network for research," *ACM/IEEE International Conference on Information Processing in Sensor Networks,* April 2014.

[12] "GitHub - LSchmidt-TUBS/ADSbDataParser," [Online]. Available: https://github.com/LSchmidt-TUBS/ADSbDataParser. [Accessed March 2021].

University of Zagreb

Technische Universität Braunschweig

POLITÉCNICA UNIVERSIDAD POLITÉCNICA DE MADRID

skyguide

Zurich University of Applied Sciences zh aw School of Engineering ZAV Center for Aviation

JKU JOHANNES KEPLER UNIVERSITÄT LINZ

Slot Consulting