D3.2 REPORT ON LONG HAUL CONFLICT-FREE TRAJECTORY OPTIMIZATION UNDER THE INFLUENCE OF HIGH ALTITUDE WIND FOR FLIGHTS CROSSING DIFFERENT TYPES OF AIRSPACE

SECURITY: PUBLIC

Lead beneficiary: CARERI
Contractual Due Date: 31/12/2022 (M36)
Actual Submission Date: 02/02/2023 (M38)

Grant Agreement number:	875154
Project acronym:	GREAT
Project title:	GREENER AIR TRAFFIC OPERATIONS
Funding scheme:	RIA/ H2020
Start date of the project:	January 1st, 2020
Duration:	42 months
Project coordinator (organisation):	Michael Finke (DLR)
Phone:	+49 531 295-2921
E-mail:	Michael.Finke@dlr.de
Project website address:	www.project-great.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 875154 GreAT.

DOCUMENT INFORMATION

DOCUMENT NAME	D3.2 Report on long haul conflict-free trajectory optimization under the influence of high altitude wind for flights crossing different types of airspace
VERSION	VF (Final version)
VERSION DATE	02/02/2023
AUTHOR	CARERI
SECURITY	Confidential

DISTRIBUTION LIST

FULL NAME OR GROUP
GreAT Consortium EU
GreAT Consortium China
Project Officer
Other

PROPRIETARY RIGHTS STATEMENT

DISCLAIMER

TABLE OF CONTENTS

GLOSSARY

Acronym	Signification
AMDAR	Aircraft Meteorological Data Relay
QAR	Quick Access Recorder
ECMWF	European Centre for Medium-Range Weather Forecasts
EHAM	Amsterdam Airport Schiphol
EPS	Ensemble Prediction System
RBT	Reference Business Trajectory
RMSE	Root mean square errors
TBO	Trajectory based operation
BDT	Business development trajectory
SBT	Shared Business Trajectory
RBT	Reference Business Trajectory
STD	Standard Deviation
HMOA	Hybrid Meta-heuristic Optimization Algorithm
ZGGG	Guangzhou Baiyun International Airport
VHHH	Hong Kong International Airport

1. INTRODUCTION

Under the trajectory-oriented operation mode of the next-generation air traffic management system, the cooperative and collaborative trajectory planning mode will be adopted between ANSP and the airlines. In the frame of subtask 3.2, research on the robust trajectory optimization and deconfliction method is carried out, considering plan data of Sino-European flights as the reference scenario.

In the pre-tactical stage of flight operation, the trajectory planning of a single flight and the conflict detection and resolution of multiple flights are affected by the uncertainty of the forecasted high-altitude wind. The purpose of the research is to analyze the uncertainty characteristics of the high-altitude wind ensemble forecasts of ECMWF EPS. Based on quantifying the uncertainty of wind forecast data and the trajectory point passing time uncertainty, the task will then investigate the method of trajectory optimization and the robust method of conflict detection and resolution while considering the wind prediction uncertainty.

To support the uncertainty analyzing method of high-altitude wind information (provided as the required operating environment data for long-haul green trajectory optimization) the forecasts of the ECMWF EPS have been provided over selected periods. This task focus on the characterization of the air-ground provided wind information, obtaining wind uncertainty characteristics by comparing the actual wind computed by the aircraft's onboard system and the wind data predicted by the ground system.

The Ensemble Prediction System (EPS) is a method used in numerical weather prediction. Instead of making a single forecast of the most likely weather, a set of forecasts is produced, aiming to give indications of the range of possible future states of the atmosphere. The multiple simulations are conducted to account for the two sources of uncertainty: the errors introduced by imperfect initial conditions, amplified by the chaotic nature of the evolution of the atmosphere, and errors introduced because of imperfections in the model formulation, such as the approximate mathematical methods to solve the equations. Ideally, the verified future atmospheric state should fall within the predicted ensemble spread, and the amount of spread should be related to the uncertainty of the forecast. The EPS of ECMWF consists of 51 forecasts. At the same time, to analyze the error of the forecast wind data, airborne AMDAR (Aircraft Meteorological Data Relay) data, and airborne QAR (Quick Access Recorder) data are used to obtain "real" high-altitude wind data.

The main research work and the corresponding chapters are shown in Figure 1, which include:

(1) Study the operation process of collaborative trajectory planning between ANSP and airlines;

(2) Study the wind ensemble forecast uncertainty characteristics based on forecasts of the ECMWF EPS and wind forecast error when compared with both the high-altitude wind 0-Hr analysis data and AMDAR data, and research on the effectiveness of correcting high-altitude wind forecasts based on historical high-altitude wind forecast errors;

(3) Study the trajectory prediction method considering the high-altitude wind and the quantitative analysis method of the uncertainty characteristics of the point passing

time considering the wind ensemble forecast data, to provide the necessary input data for robust conflict detection and resolution;

(4) Study the optimization method of the horizontal route and vertical profile for the flight;

(5) Study the robust trajectory conflict detection and resolution method considering the predicted point passing time uncertainty.

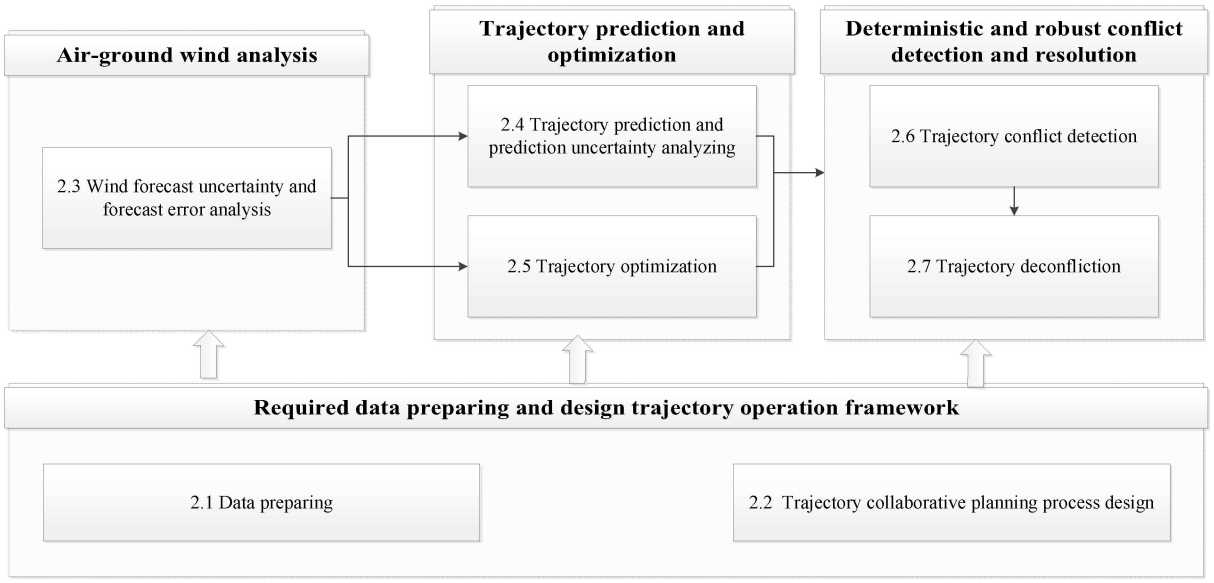


Figure 1 Research work done and corresponding chapters

2. WORK PERFORMED

2.1. INVESTIGATIONS AND RESULTS ON PREPARING THE FLIGHT PLAN AND WIND DATA

The flight plan data obtained mainly include the flight plan data of the Sino-European city pair(Hong Kong and Amsterdam) from 2019.6.1 to 2019.6.15 (provided by KLM) and the flight plan data passing through any of the same sectors as that of the flight VHHH-EHAM.

The ECMWF wind ensemble forecast and AMDAR data are collected for this research.

For the AMDAR data, the set of data covers the period of 1-7 June 2019. The geographical coverage of the data has been provided (Inside the yellow box in Figure 2):

- Latitude from 3° north to 46°north
- Longitude from 92° east to 129° east.

For the ECMWF EPS data, specifically, a first set of data has been provided, which is related to the week of 1-7 June 2020. Then, considering that COVID-19 had a relevant impact on aviation in 2020, in a successive phase a second set of data has been provided, related to the period 1-15 June 2019.
To optimize the trajectory of flights between Europe and China, GRIB data that contains the following geographical coverage has been provided (Inside the red box in Figure 2):

- Latitude: from 3° north to 72°north
- Longitude: from 9° west to 136° east

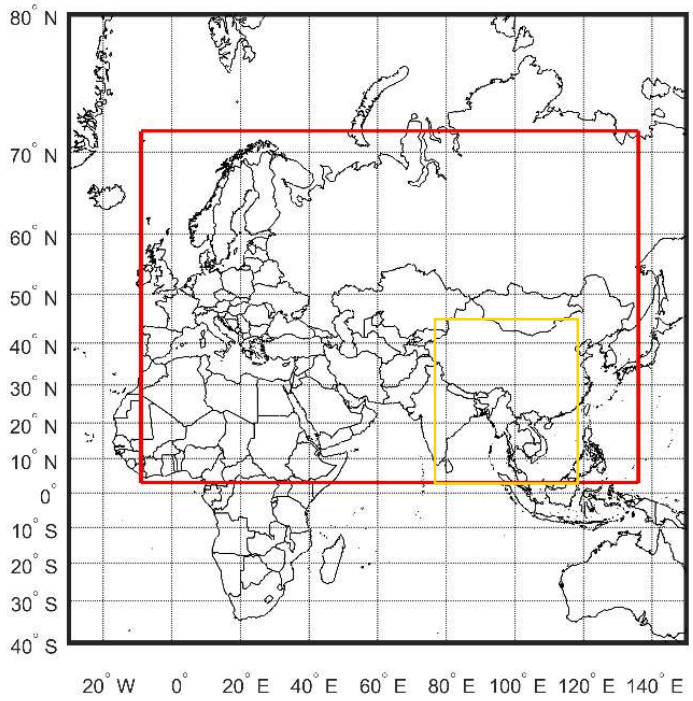


Figure 2: The computational domain considered of the AMDAR and ECMWF data

For each day of the period considered (1-7 June 2020 and 1-15 June 2019), analysis data at hour 00:00 UTC and forecast data up to 72 hours have been provided. Data are

organized in daily folders. Each folder contains 73 files (1 analysis file and 72 hourly forecasts). The generical hourly GRIB file (e.g. 20200601_0hr-fcst.grib1) contains 2040 records. There are:

- 4 variables: T, R, U, V (Temperature [K], Relative humidity [%], U velocity [m s-1], V velocity [m s-1]);
- 10 pressure levels: 100,200,250,300,400,500,700,850,925,1000 mb;
- 51 ensemble members: 1 Control forecast + 50 Perturbed forecasts;

For a total number of (51x4x10 = 2040) records.

All the records in the file share the same Long/Lat grid (321 x 171) with a spatial resolution of 0.2°.
There is also an additional folder (analysis) containing further analysis files, since at ECMWF. The Analysis data is available every 6 hours (00 06 12 18) so four analysis files for each of the days considered have been provided.

Figure 3 shows the logical organization of each folder with the files included in it.

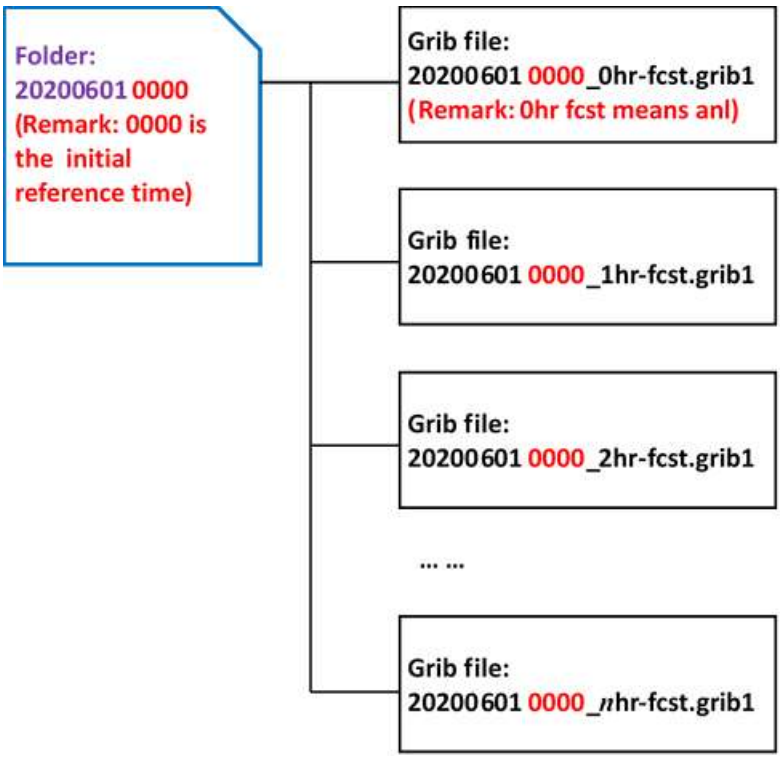


Figure 3: Logical organization of folders and files included

Moreover, all the other analysis files available at ECMWF have been provided in a separate folder "Analysis", namely at hours 06, 12, and 18 UTC of each day of the periods considered.

Figure 4 shows an example of the output that can be obtained with the software "wgrib" applied to a generical file, for a subset of the GRIB records.

1	0	D=2020010100	T	100 mb	kpds=130,100,100	anl	type=Control forecast 0	winds are N/S	Temperature [K]
2	109944	D=2020010100	R	100 mb	kpds=157,100,100	anl	type=Control forecast 0	winds are N/S	Relative humidity [%]
3	219888	D=2020010100	U	100 mb	kpds=131,100,100	anl	type=Control forecast 0	winds are N/S	U velocity [m s**-1]
4	329832	D=2020010100	V	100 mb	kpds=132,100,100	anl	type=Control forecast 0	winds are N/S	V velocity [m s**-1]
5	439776	D=2020010100	T	200 mb	kpds=130,100,200	anl	type=Control forecast 0	winds are N/S	Temperature [K]
6	549720	D=2020010100	R	200 mb	kpds=157,100,200	anl	type=Control forecast 0	winds are N/S	Relative humidity [%]
7	659664	D=2020010100	U	200 mb	kpds=131,100,200	anl	type=Control forecast 0	winds are N/S	U velocity [m s**-1]
8	769608	D=2020010100	V	200 mb	kpds=132,100,200	anl	type=Control forecast 0	winds are N/S	V velocity [m s**-1]
…	…	…	…	…	…	…	…	…	…
41	4397760	D=2020010100	T	100 mb	kpds=130,100,100	anl	type=Perturbed forecast 1	winds are N/S	Temperature [K]
42	4507704	D=2020010100	R	100 mb	kpds=157,100,100	anl	type=Perturbed forecast 1	winds are N/S	Relative humidity [%]
43	4617648	D=2020010100	U	100 mb	kpds=131,100,100	anl	type=Perturbed forecast 1	winds are N/S	U velocity [m s**-1]
44	4727592	D=2020010100	V	100 mb	kpds=132,100,100	anl	type=Perturbed forecast 1	winds are N/S	V velocity [m s**-1]
…	…	…	…	…	…	…	…	…	…
85	9235296	D=2020010100	T	200 mb	kpds=130,100,200	anl	type=Perturbed forecast 2	winds are N/S	Temperature [K]
86	9345240	D=2020010100	R	200 mb	kpds=157,100,200	anl	type=Perturbed forecast 2	winds are N/S	Relative humidity [%]
87	9455184	D=2020010100	U	200 mb	kpds=131,100,200	anl	type=Perturbed forecast 2	winds are N/S	U velocity [m s**-1]
88	9565128	D=2020010100	V	200 mb	kpds=132,100,200	anl	type=Perturbed forecast 2	winds are N/S	V velocity [m s**-1]
…	…	…	…	…	…	…	…	…	…
2037	223845984	D=2020010100	T	1000 mb	kpds=130,100,1000	anl	type=Perturbed forecast 50	winds are N/S	Temperature [K]
2038	223955928	D=2020010100	R	1000 mb	kpds=157,100,1000	anl	type=Perturbed forecast 50	winds are N/S	Relative humidity [%]
2039	224065872	D=2020010100	U	1000 mb	kpds=131,100,1000	anl	type=Perturbed forecast 50	winds are N/S	U velocity [m s**-1]
2040	224175816	D=2020010100	V	1000 mb	kpds=132,100,1000	anl	type=Perturbed forecast 50	winds are N/S	V velocity [m s**-1]

Figure 4: Example of output provided by "wgrib" for a subset of the GRIB records

2.2. INVESTIGATIONS AND RESULTS OF THE TRAJECTORY COLLABORATIVE PLANNING PROCESS

With the forthcoming TBO concept, a transition in ATM from control by tactical clearance to management by reference to a trajectory emphasizes the importance of efficient trajectory planning and optimization.

Under the trajectory operation mode, one key aspect is the migration to Trajectory-Based Operations (TBO) under which each flight is represented with a trajectory that is shared, managed, and used as a common plan for the flight. The interactive trajectory design working process is adopted in the pre-tactical stage, aligned with the CDM paradigm of air traffic flow management which is shown in Figure 5. Under the collaborative and interactive trajectory operation mode, airlines can submit their preferred trajectory to ATM, then ATM would subsequently perform conflict resolution between individual trajectories considering the benefit needs of the airlines. The specific interaction process can be done in a way similar to the following:

1) Airlines will submit the initially scheduled trajectories;

2) ATM will then make adaptive reconfiguration of the flexible en-route airspace according to the airlines' initial planned trajectory;

3) Then airlines can make optimization of the initial trajectory based on the forecast weather and the reconfigured airspace. When the hot-spot constraint avoidance information provided by ATM is not considered, the designed trajectory is called the BDT (Business Development Trajectory), otherwise, the designed trajectory is called the SBT (Shared Business Trajectory);

4) ATM will make trajectory de-confliction based on the airline's submitted trajectories and then generate the RBT (Reference Business Trajectory) which deviates from the BDT as little as possible.

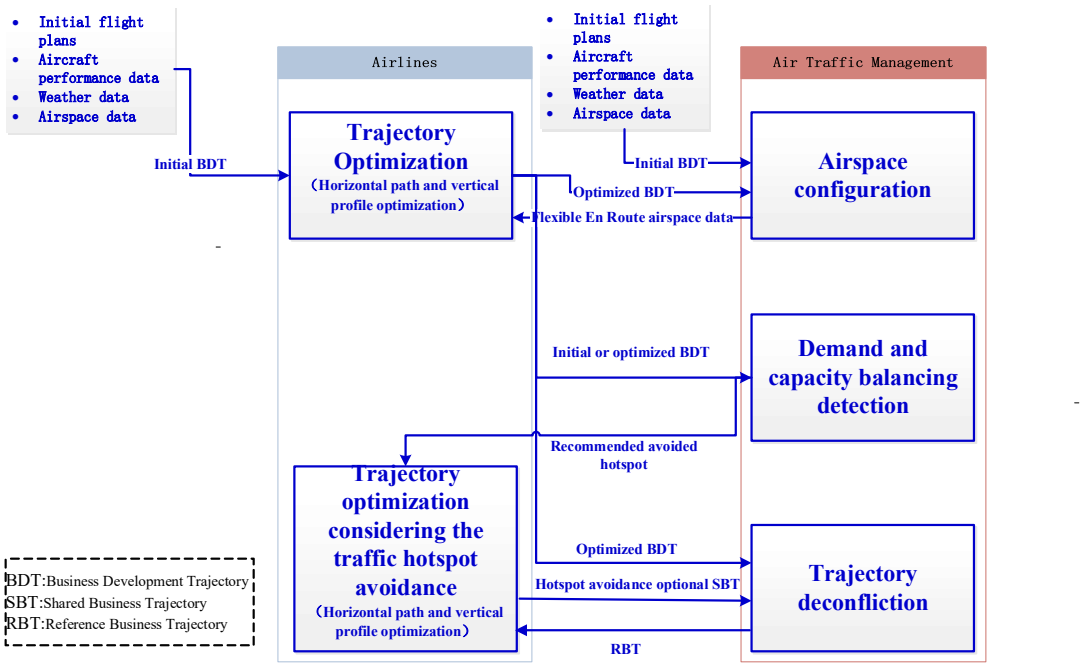


Figure 5: Collaborative and interactive trajectory design process

In the above-mentioned trajectory collaborative planning process, airlines can express their trajectory preferences in a variety of ways: (1) they can submit their preferred BDT as the initial trajectory to the air traffic manager, and the air traffic manager can minimize the deviation from the airline preferred trajectory when they have to adjust the flights for resolving the conflict; (2) airlines can also submit alternative hotspot avoidance SBT after the air traffic manager announces the congestion area to the airlines; (3) the airline submits the trajectory adjustment methods preferred by the airline when the air traffic manager needs to carry out trajectory adjustment to deconflict the trajectory, such as ground delay, airborne waiting, speed regulation, flight altitude adjustment, horizontal route adjustment, etc. Airlines can provide their preference information through the above preference expression ways, to participate in air traffic management decision-making.

2.3. INVESTIGATIONS AND RESULTS ON WIND FORECAST UNCERTAINTY AND FORECAST ERROR

2.3.1. WIND ENSEMBLE FORECAST VARIATIONS

To analyze the uncertainty of high-altitude wind ensemble forecasts, standard deviations of the ensemble forecasts of ECMWF EPS over different geographic locations and look-ahead time are examined. Route from Guangzhou Baiyun International Airport (ZGGG) or the Hong Kong International Airport (VHHH) in China to Schiphol Airport (EHAM) in the Netherlands is selected as the typical China-EU flight. Standard deviations of the 51 wind forecast ensemble members are calculated for each waypoint of the route.

The results of U and V components under forecast look-ahead time of 0h, 12h, 24h, 48h, and 72h are shown in Figure 6. Figure 7 shows the standard deviations of U and V components for the route points within China and outside of China. The mean value of the standard deviation of U and V components of the high-altitude wind is shown in Table 1 and the maximum and minimum standard deviation of U and V components is shown in Table 2. It can be seen that the dispersion degree of the ensemble forecast increases with the increase of the look-ahead time. It suggests that the forecast uncertainty increases with look-ahead time increases. Regional differences exist for the standard

deviation of different geographic locations. When the look-ahead time is over 24 hours, the uncertainty is increased apparently for route points in some geographical areas.

Robust conflict detection and resolution will be made after Shared Business Trajectory (SBT), which is then finalized into a Reference Business Trajectory (RBT) in the pre-tactical stage. Wind forecast uncertainty level changes with the forecast look-ahead time and the robust trajectory deconfliction will also be impacted when the deconfliction is made at a different look-ahead time.

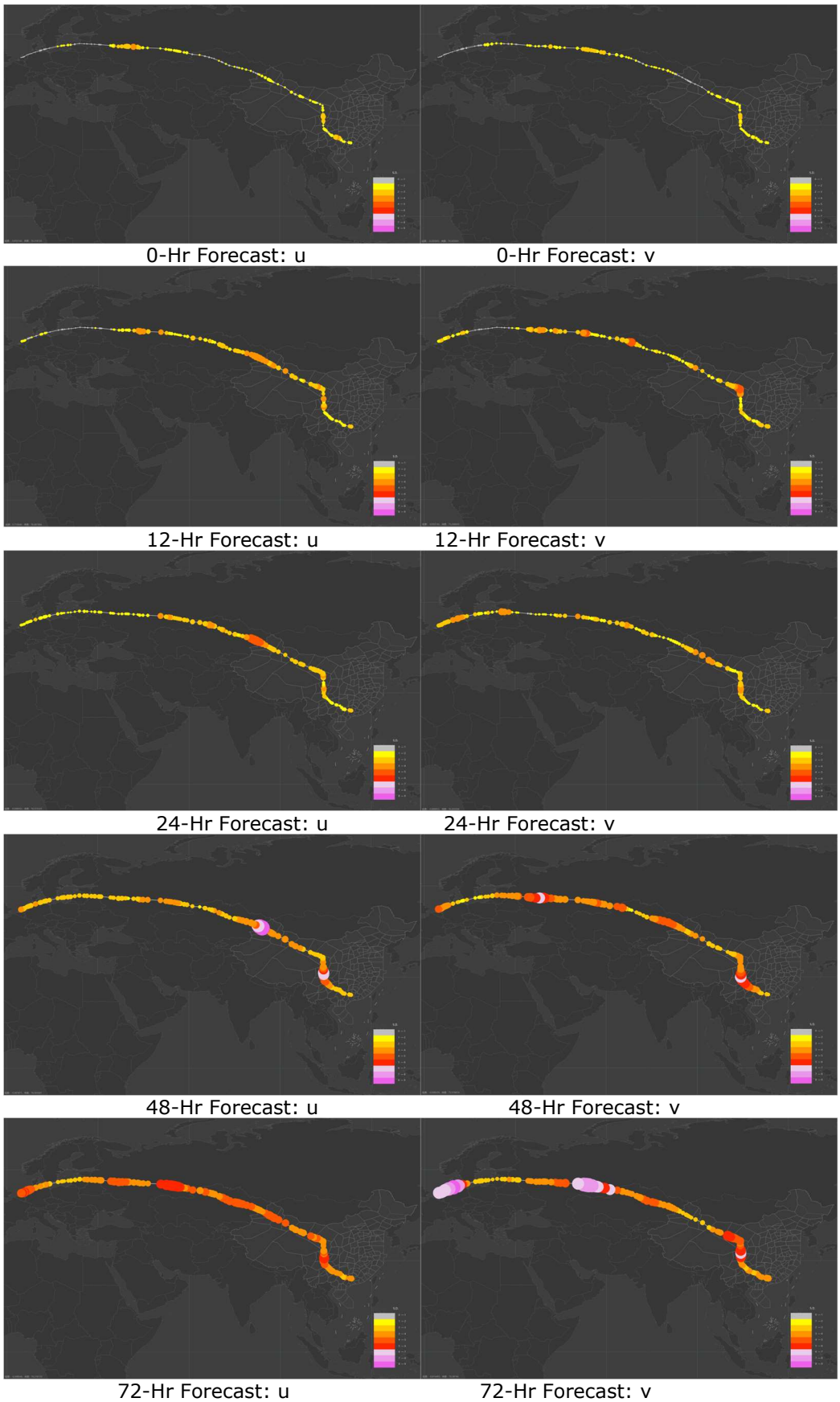
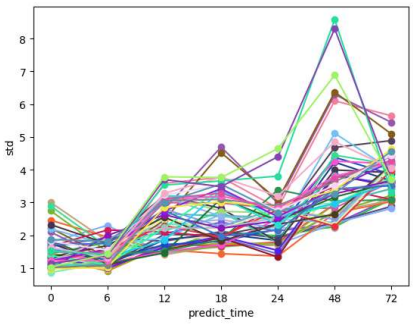
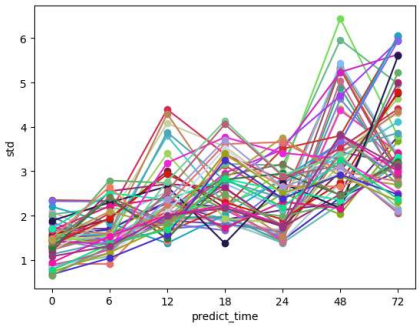


Figure 6: STDs at each route point for the route between ZGGG and EHAM

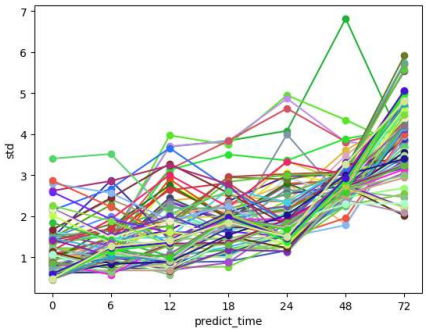
for 250mb Pressure Altitude and 5 different look-ahead times



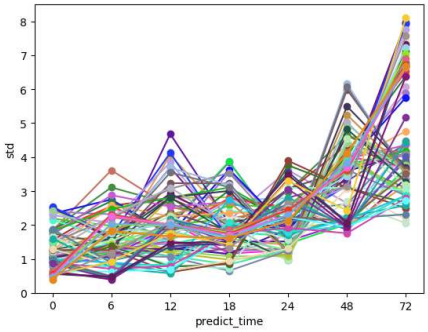
(a)STD of U-component at route points within China



(b) STD of V-component at route points within China



(a)STD of U-component at route points outside of China



(b) STD of V-component at route points outside of China

Figure 7: Standard Deviation (STD) of U-Component and V-Component at route points within and outside of China

Table 1: Mean value of the standard deviation of U and V components of the high-altitude wind for different LAT

LAT	Mean value of STD of U components within China	Mean value of STD of V components within China	Mean value of STD of U components outside of China	Mean value of STD of V components outside of China
0h	1.531	1.398	1.164	1.232
6h	1.447	1.678	1.357	1.542
12h	2.394	2.338	1.701	1.973
18h	2.605	2.567	1.944	1.975
24h	2.459	2.331	2.113	2.113
48h	3.714	3.549	2.85	3.496
72h	3.837	3.620	4.029	4.658

Table 2: Maximum and minimum value of the standard deviation of U and V components of the high-altitude wind for different LAT

	Maximum STD of U components within China	Minimum STD of U components within China	Maximum STD of V components within China	Maximum STD of V components within China	Maximum STD of U components outside of China	Minimum STD of U components outside of China	Maximum STD of V components outside of China	Maximum STD of V components outside of China
0h	3.005	0.863	2.347	0.65	3.401	0.451	2.54	0.398

6h	2.303	0.907	2.789	0.908	3.515	0.569	3.613	0.395
12 h	3.785	1.406	4.394	1.373	3.968	0.568	4.682	0.576
18 h	4.693	1.44	4.135	1.372	3.837	0.762	3.874	0.654
24 h	4.647	1.337	3.757	1.380	4.948	1.108	3.892	0.966
48 h	8.586	2.234	6.441	2.029	6.816	1.778	6.155	1.758
72 h	5.643	2.821	6.065	2.052	5.914	2.009	8.107	2.054

2.3.2. WIND FORECAST ERROR

To calculate the high-altitude wind forecast error, both the forecast data and the "true" data of the high-altitude wind are needed. The high-altitude forecast data used is from the ECMWF EPS. The "true" wind data is based on ECMWF high-altitude wind 0h-analysis data and AMDAR data separately.

The calculation is carried out from the following three aspects:

(1) Based on forecast data of the ECMWF EPS and AMDAR data for the same period, the forecast errors at different look-ahead times are calculated, and the error characteristics are analyzed.

$$\Delta W^{EP-AMDAR}(r,t|t_0) = W^{Ensemble\,\mathrm{Predict}}(r,t|t_0) - W^{AMDAR}(r,t) \quad (2.3.1)$$

where $\Delta W^{EP-AMDAR}(r,t|t_0)$ represents the forecast error related to AMDAR data. $W^{Ensemble\,\mathrm{Predict}}(r,t|t_0)$ is the forecast data determined by interpolation of the wind grid data of time t forecasted at time t_0 . $W^{AMDAR}(r,t)$ is the "true" winds at time t for the geographic region r got from the AMDAR data.

For the period considered (from 00:00 to 01:00 on 4 June 2019), the results are listed in Table 3. In the first row of the table, the first member of the 0-Hr forecast data of the ECMWF EPS is used. In the second row of the table, the 51 members of the 0-Hr forecast data are used. Root mean square error (RMSE) for the AMDAR data with that for the 0-Hr analysis data indicate a high degree of consistency between the two kinds of "true" data of the high-altitude wind.

Table 3: Mean and RMSE of the wind forecast errors compared with AMDAR data

Look-ahead time	U-Component Mean Forecast error	U-Component RMSE	V-Component Mean Forecast error	U-Component RMSE
0-Hr (member 0)	0.333	2.563	-0.061	2.176
0-Hr(51 members)	0.334	2.72	-0.061	2.404
24-Hr	0.487	3.287	0.31	3.469
48-Hr	0.463	4.07	0.524	4.37
72-Hr	0.431	4.931	0.598	5.223

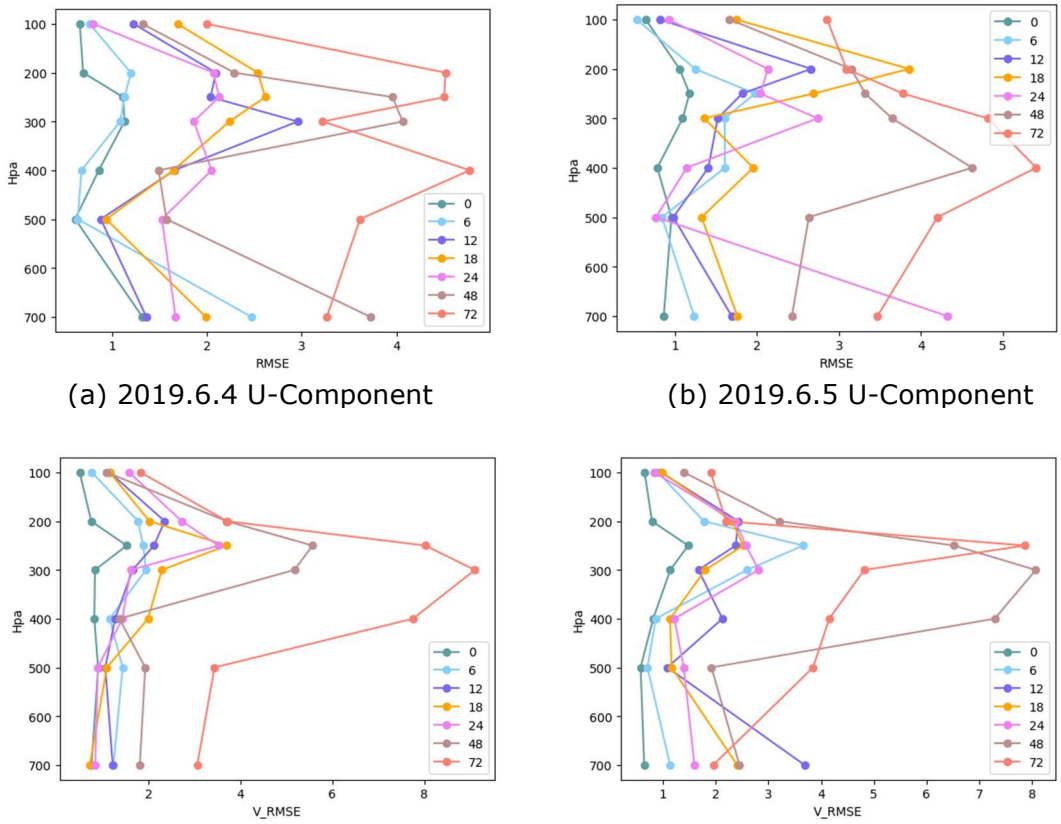
(2) Based on both forecast data and analysis data of the ECMWF EPS, wind's RMSE is obtained for route points at different pressure levels and look-ahead time.

$$\Delta W^{EP-ANA}(r,t|t_0) = W^{Ensem\,Predict}(r,t|t_0) - W^{Analysis}(r,t|t) \quad (2.3.2)$$

where $\Delta W^{EP-ANA}(r,t|t_0)$ represents the forecast error related to the analysis data of the ECMWF. $W^{Analysis}(r,t|t)$ is the interpolation of the "true" winds from the analysis grid data at time t for the geographic region r.

SARIN is selected as an example, which is an entry/exit point to China for the flights between ZGGG and EHAM. The data on 4 and 5 June 2019 are selected and RMSE is calculated at 7 pressure levels (100, 200, 250, 300, 400, 500, 700 hPa) and 7 different look-ahead times (0, 6, 12, 18, 24, 48, 72 Hr). The results are shown in Figure 8.

Readily apparent in Figure 8 is the consistent decrease in upper-level wind forecast RMSE as forecast length is decreased. The 0-Hr RMSE not equal to 0 is because of the different values of the ensemble members of the 0-Hr forecast data. RMSE varies as the pressure level changes but does not exhibit a clear pattern.



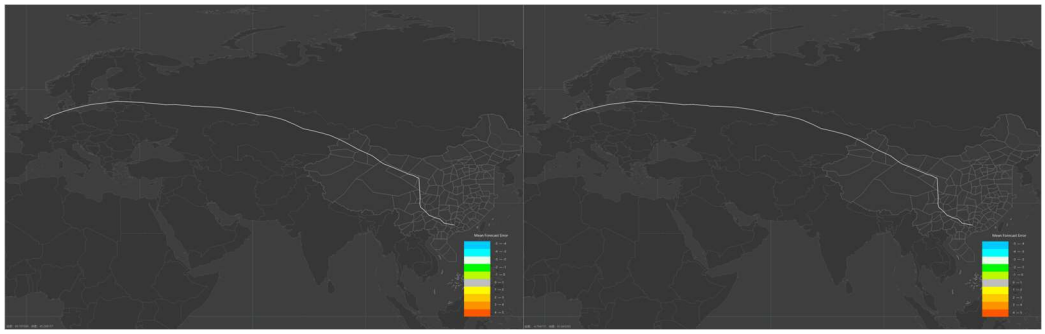
(a) 2019.6.4 U-Component (b) 2019.6.5 U-Component

(c) 2019.6.4 V-Component (b) 2019.6.5 V-Component

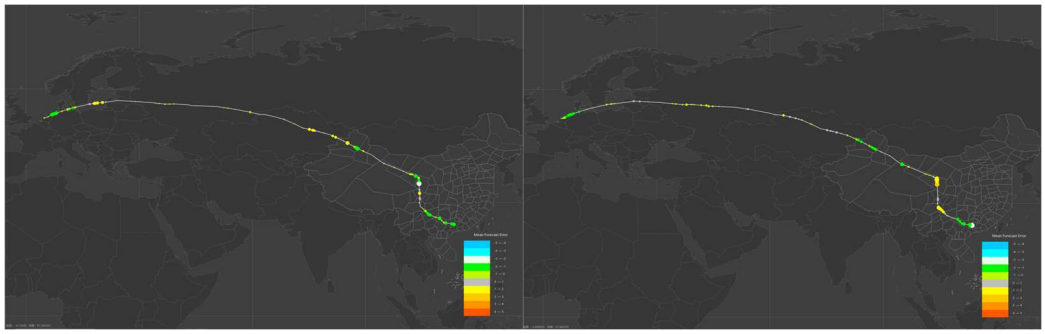
Figure 8: RMSE at route point SARIN verified against 0-Hr analysis data for different look-ahead times (0,6,12,18,24,48,72 hours) and different pressure levels (100,200,250,300,400,500,700mb)

(3) Based on the forecast data and analysis data of the ECMWF EPS, the average forecast error within a period is calculated for different route points to investigate the geographic distribution of the wind forecast error.

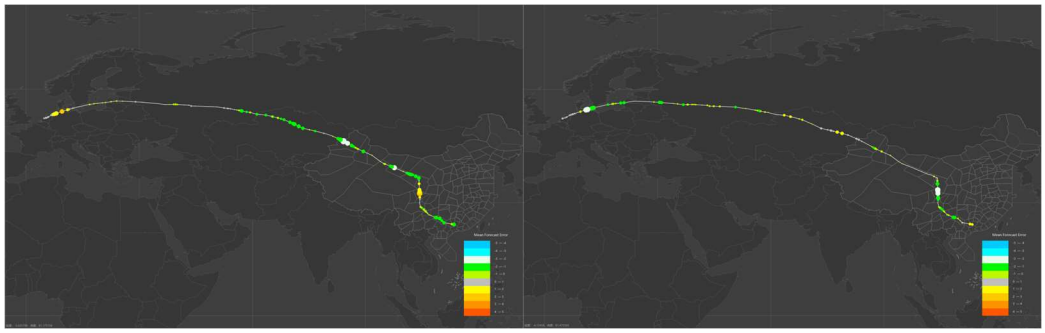
The mean forecast error is calculated for each waypoint on the route of the ZGGG-EHAM. The period is from June 1 to June 12 in 2019, covering 12 days of wind ensemble forecast values and wind 0-Hr analysis data. Results are shown in Figure 9. The maximum and minimum mean error is listed in Table 4. The actual wind for this period was almost entirely from west to east. Positive U-component errors suggest forecast westerlies are greater than analyzed westerlies, while negative U-component errors suggest the forecast result is smaller than that is analyzed. Positive errors also indicate that the trajectory based on the forecast wind could be too fast, while negative errors indicate the trajectory would be too slow. For flights flying from ZGGG to EHAM, negative U-component errors are more frequent than positive errors. For the V-component, positive errors and negative errors show alternatively.



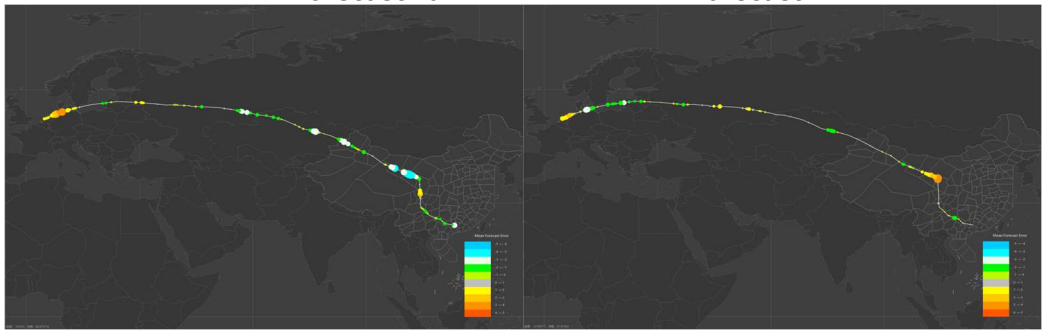
6-Hr Forecast: u 6-Hr Forecast: v



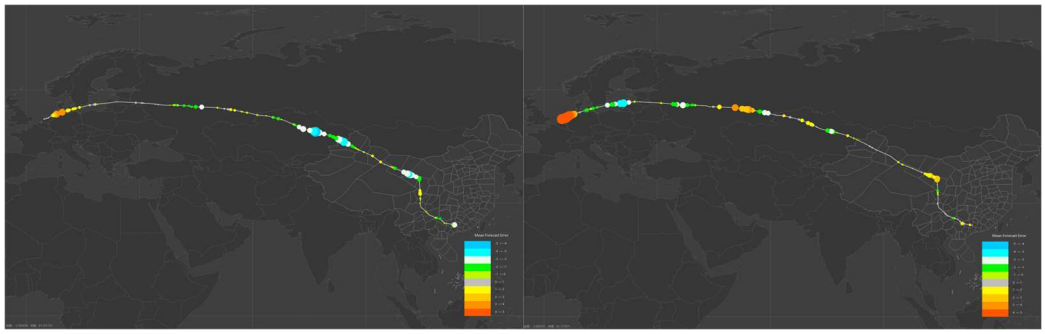
12-Hr Forecast: u 12-Hr Forecast: v



24-Hr Forecast: u 24-Hr Forecast: v



48-Hr Forecast: u 48-Hr Forecast: v



72-Hr Forecast: u 72-Hr Forecast: v

Figure 9: Mean forecast errors at each route point for the airlines between ZGGG and EHAM for 250mb pressure altitude and 5 different look-ahead times

Table 4: Maximum and minimum mean forecast error compared with the 0h analysis data

	Maximum mean forecast error of U components	Minimum mean forecast error of U components	Maximum mean forecast error of V components	Minimum mean forecast error of V components
0h	0.000435	-0.00047	0.000410	-0.00060
6h	0.739	-1.250	1.0719	-1.898
12h	1.679	-2.184	2.008	-2.261
18h	1.795	-2.424	1.285	-2.407
24h	2.49	-2.696	1.503	-2.402
48h	3.382	-3.541	3.698	-2.582
72h	3.085	-4.364	4.557	-3.453

2.3.3. WIND FORECAST CORRECTION WITH HISTORICAL WIND FORECAST ERROR

To verify the wind forecast corrected with the historical wind forecast error, forecast errors averaged over several days for each route point between ZGGG and EHAM, and for each ensemble forecast, are used to correct.

Revise the high-altitude wind forecast value of June 13 in 2019 with the average forecast errors. RMSE of the forecast data before and after the correction is used to analyze whether the wind forecast correction with historical forecast errors can reduce the error of high-altitude wind forecasts.

Table 5 lists RMSE for U-component and V-component forecast data before and after the correction on 13 June 2019. There's no significant reduction in the RMSE.

Table 5: RMSE of the wind forecast before and after the correction

Look-ahead time	RMSE for U before correction	RMSE for V before correction	RMSE for U after correction	RMSE for V after correction
0-Hr	1.597	1.615	1.670	1.679
6-Hr	2.204	2.245	2.241	2.316
12-Hr	3.831	2.499	3.874	2.666
18-hr	3.09	3.017	3.268	3.083
24-Hr	3.458	3.276	3.738	3.388
48-Hr	4.081	4.921	4.341	4.986
72-Hr	7.150	6.898	7.718	6.488

2.4. INVESTIGATIONS AND RESULTS ON TRAJECTORY PREDICTION METHOD AND PREDICTION UNCERTAINTY CHARACTERIZATION ANALYZING

2.4.1. ENSEMBLE TRAJECTORY PREDICTION BASED ON ENSEMBLE FORECASTING OF HIGH-ALTITUDE WINDS

In the pre-tactical stage, trajectory prediction will be affected by various uncertain factors, including the input parameters, aircraft intent, aircraft performance parameters, and the assumed atmospheric conditions, etc. Among them, the uncertain trajectory prediction method considering the ensemble forecast of high-altitude wind mainly includes the following two kinds:

(1) Ensemble Trajectory Forecast method: Each forecast member in the ensemble weather forecast product is used as the input of the deterministic trajectory prediction, and the predicted trajectory corresponding to each member of the weather ensemble forecast will constitute an ensemble of the predicted trajectories. The uncertainty of the trajectory prediction is obtained by quantifying method applied to the trajectory ensemble.

(2) Probabilistic Transforming method: The probability distribution is fitted according to the members of the wind ensemble forecast. The probabilistic information of the wind is used as the input of the probabilistic trajectory prediction. Through the probabilistic transforming method, that is, the PDF of the sum of stochastic variables is the convolution of the PDF of each addition. Thus, the PDF of flight time and so on can be obtained according to the PDF of the wind speed and aircraft ground speed.

The ensemble trajectory forecast method is applied for this project, the schematic diagram of the method and its application is shown in Figure 10:

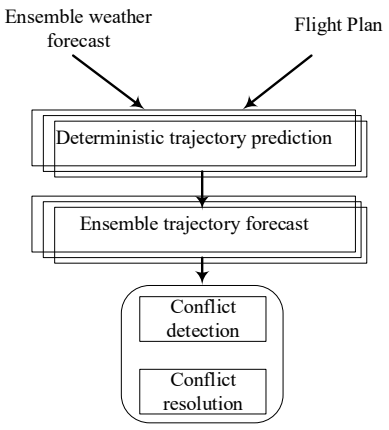


Figure 10: Schematic diagram of the ensemble trajectory forecast method and its application

(1) Deterministic trajectory prediction

Trajectory prediction is the premise and basis of trajectory conflict detection and resolution. The flying stage of the aircraft can be divided into five parts, namely, take-off, climbing, cruising, descending, and landing. Specifically, the aircraft takes off at the take-off airport, climbs from the ground to cruising altitude, and then cruises at the cruising altitude. When the aircraft reaches the top of descent, it begins to descend until landing at the destination airport. Four-dimensional trajectory flying is the fine control of the time dimension and the space dimension composed of horizontal position and altitude during the whole flight phase.

4D trajectory prediction is to predict the corresponding flight trajectory at each time in the future. It is based on the flight plan and flight intention, weather information in the forecast area and time, real-time radar data of aircraft, kinematic and dynamic characteristics based on aircraft performance, and so on to analyze and predict the motion state and situation of the flight.

4D trajectory prediction can be made for different types of aircraft. Based on the above-mentioned information, the trajectory state information of the aircraft at each time step (including the longitude and latitude of the position, flying altitude, heading, speed, attitude, etc.) is simulated, and the time when the aircraft passes through each route point is given.

The 4D trajectory prediction algorithm combines the method based on the orthodrome/loxodrome route and the method based on the aircraft motion model and dynamic model. The planned flight altitude is obtained based on historical statistical data. The overall framework of the trajectory simulation model is shown in Figure 11. Based on the initial state of the aircraft, weather environment, and aircraft performance, the four-dimensional trajectory is obtained through recursive simulation.

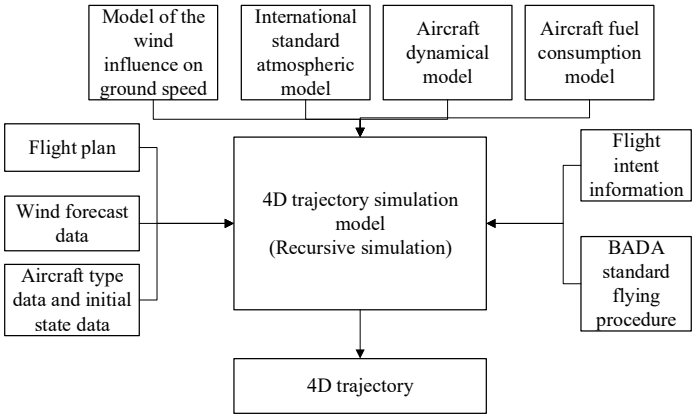


Figure 11: Trajectory simulation modeling methodology framework

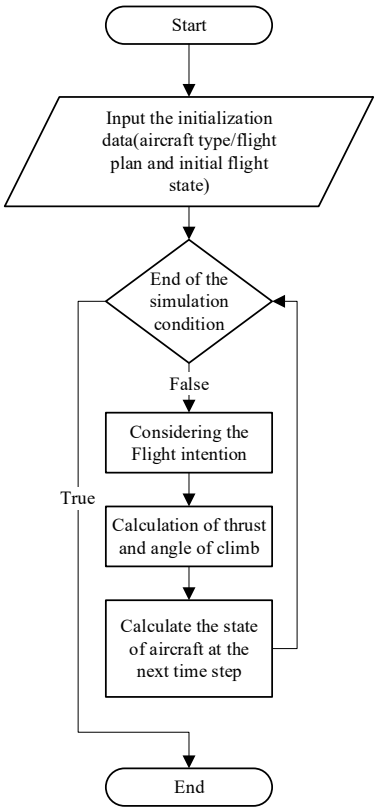


Figure 12: Flowchart of the calculation process of trajectory simulation derivation

The simulation derivation process of the trajectory is shown in Figure 12. Firstly, input the aircraft type information, flight plan, and the initial state of the aircraft (takeoff weight, take-off speed, etc.), and infer the flying intention of the aircraft according to the current state of the aircraft and the BADA standard flying procedures during the flight. The flight intention can be classified according to the aircraft's altitude variation and speed variation. After obtaining the flying intention at a certain time, calculate the thrust required by the aircraft and the Angle of Climb (AOC) according to the BADA standard flying procedure. For example, if the aircraft is climbing at a constant Indicated Air Speed (IAS), the thrust is the maximum available thrust during the climbing phase, and the AOC is calculated by the energy share factors during the climb at a constant IAS. After obtaining the required thrust and AOC, the next moment flight state is derived according to the formula after discretizing the state equation, which is as follows.

$$v_{k+1} = v_k + \Delta t\left(\frac{Thr_k - D_k}{m_k} - g\sin\gamma_k\right) \tag{2.4.1}$$

$$s_{k+1} = s_k + \Delta t\left(\sqrt{\left(v_k\cos\gamma_k\right)^2 - W_x(s_k,h_k)^2} + W_s(s_k,h_k)\right) \tag{2.4.2}$$

$$h_{k+1} = h_k + \Delta t\left(v_k\sin\gamma_k\right) \tag{2.4.3}$$

$$m_{k+1} = m_k + \Delta t\left(-FF_k\right) \tag{2.4.4}$$

In the calculation process of the conflict resolution algorithm, a simplified calculation method of deterministic trajectory prediction based on the members of high-altitude wind ensemble forecast is as follows: the speed triangle formula is used to calculate the aircraft ground speed, and then the trajectory prediction for the next sampling point is made according to the calculated ground speed.

To keep the aircraft along a given trajectory, the effect of the lateral wind on the trajectory must be corrected by changing the direction of the True Air Speed (TAS) by the amount determined by TAS and the heading as well as the wind speed and direction.

The wind triangle formula is shown in Figure 13, and the calculation method is shown in the formula (2.4.5)-(2.4.7), where $\theta_{heading}$ is the heading of the ground trajectory to the next waypoint, θ_{wind} is wind direction, WTA is wind angle, WCA is wind calibrated angle. The arriving time of the aircraft to the next trajectory sample point can be calculated according to the trajectory ground distance and ground speed to the point, as shown in the formula (2.4.8), where s_{seg} is the Great Circle distance to the position of the next point.

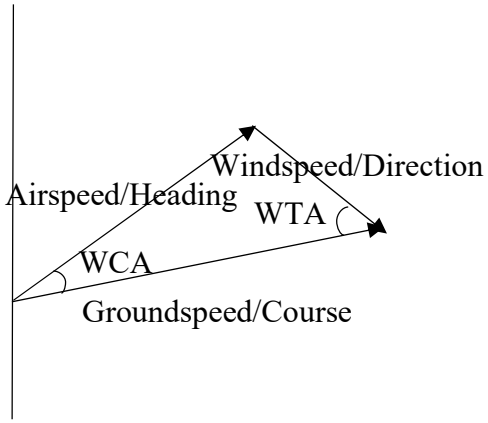


Figure 13: Wind triangle diagram

$$WTA = \theta_{heading} - \theta_{wind} \quad (2.4.5)$$

$$WCA = \arcsin(V_{wind} * \frac{\sin(WTA)}{TAS}) \quad (2.4.6)$$

$$V_{ground} = V_{TAS} * \cos(WCA) + V_{wind} * \cos(WTA) \quad (2.4.7)$$

$$t_{seg} = \frac{s_{seg}}{V_{ground}} \quad (2.4.8)$$

(2) Trajectory Ensemble prediction based on high-altitude wind ensemble forecasts

The trajectory ensemble prediction method based on the high-altitude wind ensemble forecasts is shown in Figure 14. Each member of the trajectory ensemble is predicted based on the deterministic trajectory prediction method by utilizing each member of the high-altitude wind ensemble forecasts. All the predicted trajectories constitute the trajectory ensemble, and each member of the trajectory ensemble represents a possible predicted flight path. Analyzing the statistical characteristics of the trajectory ensemble can obtain information concerning the uncertainty of the predicted trajectory.

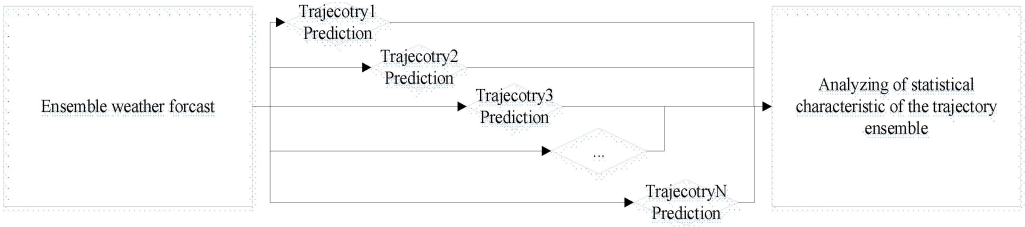


Figure 14: Ensemble trajectory prediction methods

2.4.2. UNCERTAINTY ANALYSIS OF TRAJECTORY PREDICTION BASED ON COMBINED HIGH-ALTITUDE WIND FORECASTING

Uncertainties of the high-altitude wind ensemble forecasts vary under different look-ahead times, leading to different effects on the uncertainty of trajectory prediction. The predicted waypoints passing time uncertainty characteristics are analyzed by the time deviation range of all the other members from member 0 in the trajectory ensemble.

Analyze the predicted point passing time deviations of the city pairs of ZGGG-EHAM and VHHH-EHAM flights on 2019.6.8(UTC+8) which is a typical city-pair flight between Sino-European and other flights in the same sector as the Sino-European flight (that is, flights that may have potential conflict with the Sino-European flight), respectively.

(1) Sino-European flights point passing time prediction deviation analysis

Flight plan information of ZGGG-EHAM and VHHH-EHAM for June 8, 2019, is shown in Table 6:

Table 6: Sino-European Flight Plan Information

Flight number	Aircraft type	Departure Airport	Landing Airport	Estimated take-off time(UTC)	Estimated landing time(UTC)
CSN307	A333	ZGGG	EHAM	2019.6.7 16:05	2019.6.7 04:11
KLM888	B744	VHHH	EHAM	2019.6.8 04:20	2019.6.8 16:20

The high-altitude wind data from 00:00:00(UTC) on 2019.6.6 and 00:00:00(UTC) on 2019.6.7 is used to predict the trajectory point passing time for flight CSN307. And the wind data from 00:00:00(UTC) on 2019.6.8 is used to predict the point passing time for flight KLM888. Predict the passing time under different members of the wind ensemble forecasts, and then calculate the deviation of the predicted passing time relative to the time $t_n^{f,0}$ that is predicted with the ensemble member 0 in the high-altitude wind ensemble forecasts. As is shown in Figure 15, $\delta t_n^{f,\max}$ is the maximum positive time deviation, $\delta t_n^{f,\min}$ is the maximum negative time deviation.

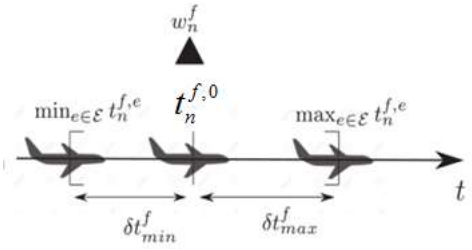


Figure 15: Flight point passing time deviation under different high-altitude wind ensemble forecast

The predicted waypoint passing time of flight CSN307 under different forecast LAT based on the ensemble member 0 wind forecast data is shown in Table 7. The predicted passing time of flight KLM888 based on the ensemble member 0 wind forecast data is shown in Table 8. The statistics of the time deviation relative to the waypoint passing time predicted with ensemble member 0 wind forecast data of these two flights are shown in Table 9.

Table 7: Predicted point passing time based on member 0 of the wind ensemble forecast data under different high-altitude wind forecast look-ahead times for the ZGGG-EHAM airline

Forecasting time (UTC)	Waypoint name	Forecast point passing time (UTC)	Wind forecast LAT(h)	Flying time relative to the departure time(hh:mm:ss)
2019.6.6 00:00:00	YIN	16:20:00	40	00:15:00
	ELKAL	16:56:29	41	00:51:29
	FJC	17:36:28	42	01:31:28
	OMBON	18:00:34	42	01:55:34
	BUKPU	18:29:34	43	02:24:34
	P407	18:52:52	43	02:47:52
	IBANO	19:28:01	43	03:23:01
	KEXAB	20:07:01	44	04:02:01
	OMKEN	20:23:58	44	04:18:58
	SARIN	20:50:27	45	04:45:27
2019.6.7 00:00:00	YIN	16:20:00	16	00:15:00
	ELKAL	16:56:21	17	00:51:21
	FJC	17:36:20	18	01:31:20
	OMBON	18:00:35	18	01:55:35
	BUKPU	18:29:53	19	02:24:53
	P407	18:52:56	19	02:47:56
	IBANO	19:28:06	19	03:23:06
	KEXAB	20:07:16	20	04:02:16
	OMKEN	20:24:18	20	04:19:18
	SARIN	20:50:47	21	04:45:47

Table 8: Predicted point passing time based on member 0 of the wind ensemble forecast data for the VHHH-EHAM airline

Forecasting time (UTC)	Waypoint name	Forecast point passing time (UTC)	Wind forecast LAT(h)	Flying time relative to the departure time(hh:mm:ss)
2019.6.8 00:00:00	YIN	05:02:32	5	00:42:32
	ELKAL	05:39:00	6	01:19:00
	FJC	06:19:33	6	01:59:33
	OMBON	06:43:55	7	02:23:55

	BUKPU	07:12:28	7	02:52:28
	P407	07:35:06	8	03:15:06
	IBANO	08:08:02	8	03:48:02
	KEXAB	08:45:59	9	04:25:59
	NIRAV	09:07:30	9	04:47:30
	GOPTO	09:31:24	10	05:11:24

Table 9: Statistical deviation value of the predicted point passing time of other ensemble members relative to the time of ensemble member 0 under different high-altitude wind forecast look-ahead time

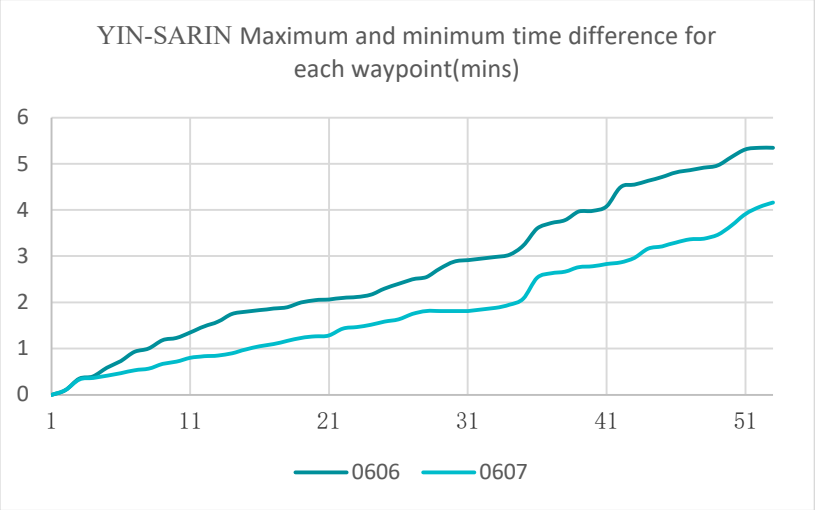
Forecasting time (UTC)	Flight	Waypoint name	Maximum positive time deviation relative to the 0-ensemble member (s)	Maximum negative time deviation relative to the 0-ensemble member (s)	Mean transition time deviation relative to the 0-ensemble members(s)	The standard deviation of the time deviation relative to the 0-ensemble members(s)
2019.6.6 00:00:00	CSN307	YIN	0	0	0	0
		ELKAL	36	-24	-1.373	12.238
		FJC	47	-67	-7.941	21.712
		OMBON	56	-74	0.392	27.479
		BUKPU	81	-83	5.255	36.163
		P407	80	-114	-15.059	48.575
		IBANO	83	-156	-32.706	59.991
		KEXAB	100	-173	-41.451	70.699
		OMKEN	108	-187	-40.922	74.543
		SARIN	111	-210	-46.137	80.568
2019.6.7 00:00:00	CSN307	YIN	0	0	0	0
		ELKAL	17	-17	-2.471	7.941
		FJC	29	-41	-4.118	15.029
		OMBON	48	-43	3.02	18.604
		BUKPU	71	-38	11.843	25.138
		P407	74	-51	11.902	32.136
		IBANO	87	-80	-0.137	41.633
		KEXAB	75	-103	-6.529	45.345
		OMKEN	84	-119	-10.353	48.17
		SARIN	106	-144	-13.392	51.475
2019.6.8 00:00:00	KLM888	YIN	8	-7	0.2549	2.4318
		ELKAL	25	-15	1.4118	7.3978
		FJC	34	-28	1.2941	12.7268
		OMBON	39	-34	1.1569	15.1253
		BUKPU	48	-44	-1.4314	19.9482
		P407	48	-58	-6.6863	23.2159
		IBANO	66	-72	-4.3922	27.1751
		KEXAB	62	-87	-3.5294	28.3177

		NIRAV	66	-97	-6.6078	30.2556
		GOPTO	81	-112	-7.0392	33.9405

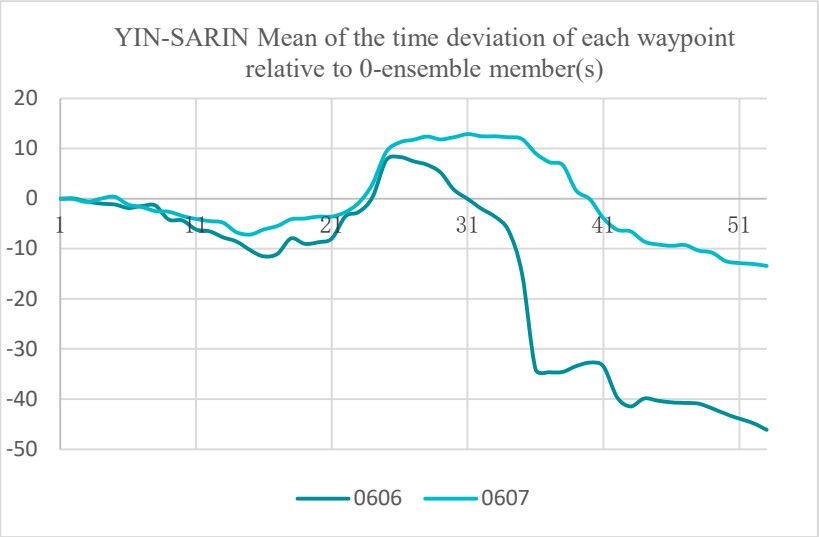
The calculation results of the statistical deviation value of the predicted point passing time under different high-altitude wind forecast look-ahead times for the ZGGG-EHAM airline are shown in Figure 16. The horizontal axis in Figure 16 is the serial number of each waypoint from YIN-SARIN. Comparing the results calculated based on the predicted high-altitude wind data forecasted at 00:00 on 2019.6.6 and 00:00 on 2019.6.7 respectively, it is found that:

1) The dispersion of waypoint passing time increases with the increasing look-ahead time of the wind forecast and the increasing flying time after departure, and the maximum positive and negative deviation value of the passing time, as well as the standard time deviation value relative to the passing time of 0-ensemble member, also increase.

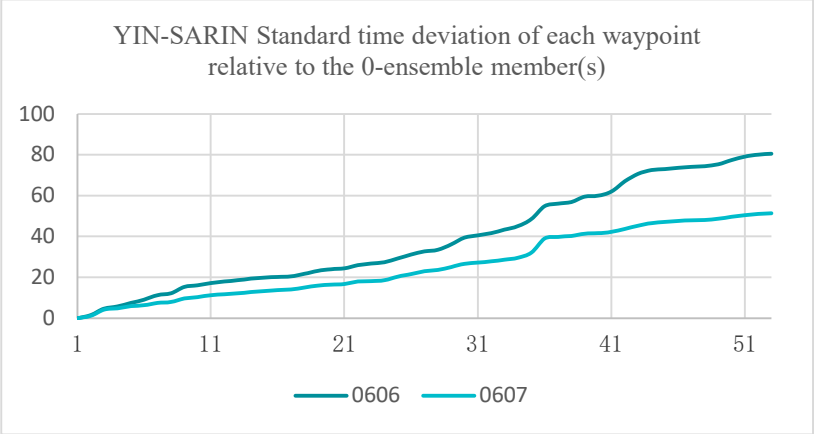
2) On the whole, the absolute value of the mean deviation value relative to the time of 0-ensemble member increases with the increase of the look-ahead time of the high-altitude wind forecast and the increase of flying time after departure. But for the relationship with the forecast look-ahead time, there are also exceptional for a small number of route points that do not conform to this law. And for the relationship with the flying time after departure, the increasing tendency of the absolute value of the mean deviation value is not so obvious for the forecast date of 2019.6.7.



(a)Maximum and minimum time deviation for each waypoint



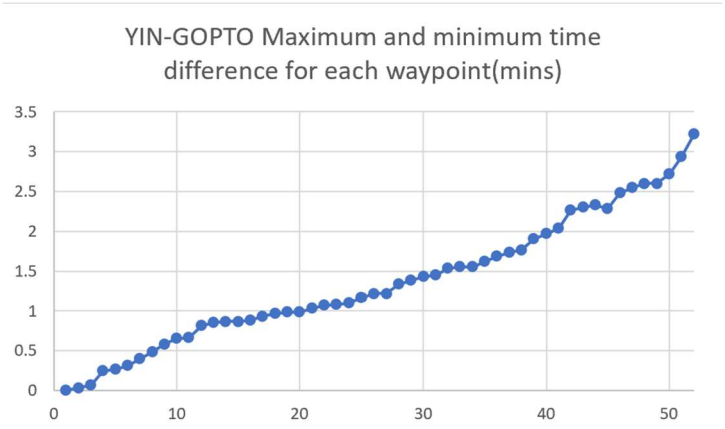
(b)Mean of the time deviation of each waypoint relative to the waypoint time of 0-ensemble member



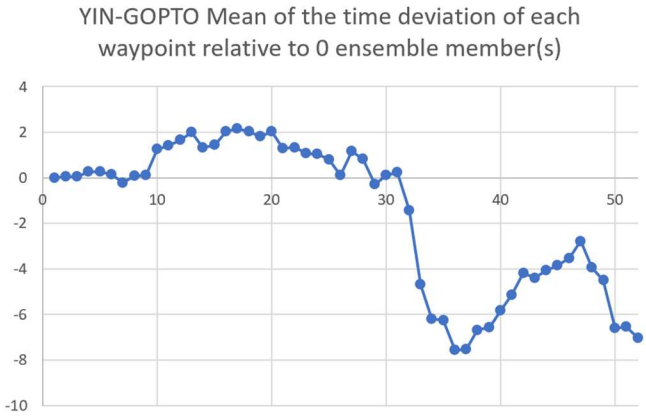
(c)Standard deviation of the time deviation of each waypoint relative to the waypoint time of 0-ensemble member

Figure 16: The deviations of passing time for different high-altitude wind forecasting look-ahead time and waypoint of the airline ZGGG-EHAM

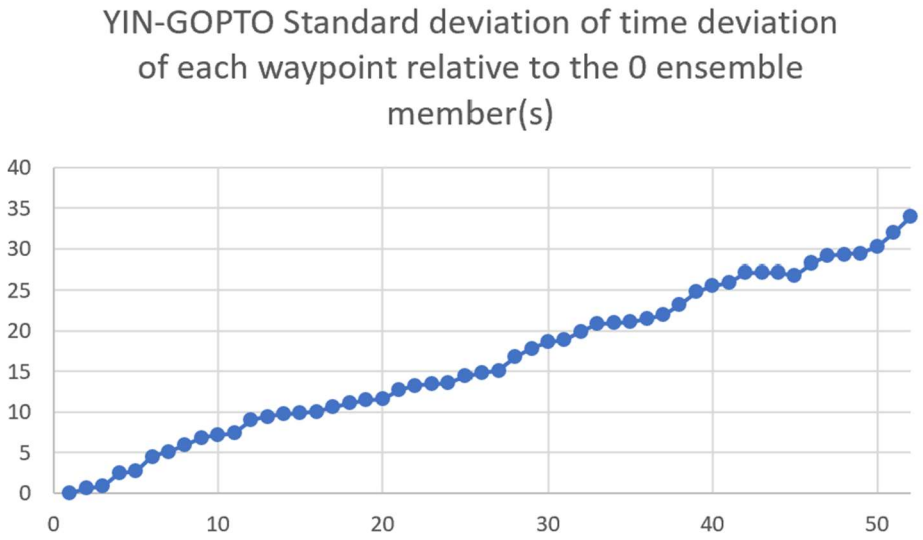
Figure 17 shows the calculation results of the time deviation statistical value for the VHHH-EHAM airline using the high-altitude wind forecast data at 00: 00:00 (UTC) on 2019.6.8.



(a)Maximum and minimum time deviation for each waypoint



(b)Mean of the time deviation of each waypoint relative to the waypoint time of 0-ensemble member



(c)Standard deviation of the time deviation of each waypoint relative to the waypoint time of 0-ensemble member

Figure 17: The deviations of passing time for different high-altitude wind forecasting look-ahead time and waypoint of the airline VHHH-EHAM

(2) Regression analysis of flight waypoint passing time deviations

Since the robust trajectory conflict resolution study intends to use the high-altitude wind data based on the 0-ensemble member of high-altitude wind ensemble forecasting for trajectory prediction. At the same time, to reflect the uncertainty characteristics of trajectory prediction, the deviation of time under different high-altitude wind ensemble members relative to the 0-ensemble member will be fitted by regression to quantify the uncertainty of the waypoint passing time based on the historical data.

To obtain the time uncertainty data needed by the robust conflict resolution algorithm, high-altitude wind data predicted at 00:00:00 (UTC) of 2019.6.8 and was used for regression analysis of the maximum positive and negative deviations of the predicted passing time relative to the predicted time of the 0-ensemble member for the affected flights passing one of the same sectors as the sectors passing by the flight between VHHH-EHAM.

1) Regression analysis of maximum positive time deviation influenced by forecast look-ahead time of the wind and flying time after departure

Using the predicted trajectory ensemble of the 3500 flights throughout the day on 2019.6.8 at different wind forecasting look-ahead times(expressed as 'LAT') and the flying time after departure(expressed as 'FlightTime') as independent variables, the SPSS software was used to establish the multiple linear regression equation of the maximum positive time deviation of the predicted waypoint passing time with regards to the 0-ensemble member with regards to LAT as well as FlightTime, and the results are shown in Table 10. The regression equation is significant with P<0.001. The independent variable of forecast look-ahead time LAT($\beta = 0.132$, $P < 0.001$) and the flying time FlightTime ($\beta = 0.816$, $P < 0.001$) significantly positively predict the maximum positive deviation of the point passing time. $R^2 = 0.707$, indicates that 70.7% of the variation of the positive time deviation can be explained by changes in the independent variables.

The established multiple linear regression equation for the maximum positive time deviation of the predicted waypoint passing time relative to the 0-ensemble member is given in Eq. (2.4.9) :

$$\delta t_{\max} = 1.094 LAT + 0.011 FlightTime - 26.815 \quad (2.4.9)$$

Table 10: SPSS maximum positive deviation multiple regression analysis results

	B	β	t	p	Adjusted R^2
Constant	-26.815		-58.175	<0.001	0.707
LAT(IV)	1.094	0.132	75.869	<0.001	
FlightTime(IV)	0.011	0.836	467.313	<0.001	

2) Regression analysis of the maximum negative time deviation influenced by forecast look-ahead time of the wind and flying time after the departure

Using the same 3500 flights and the same independent variables as the maximum positive time deviation for making the regression analysis, the results are shown in Table 11. The regression equation is significant with P<0.001. The independent variable of forecast look-ahead time LAT ($\beta = -0.105$, $P < 0.001$) and the flying time FlightTime ($\beta = -0.825$, $P < 0.001$) significantly negatively predict the maximum negative deviation of the point passing time. $R^2 = 0.711$, indicates that 71.1% of the negative deviation can be explained by changes in the independent variables.

The established multiple linear regression equation for the maximum negative deviation of the predicted waypoint passing time relative to the 0-ensemble member is given in Eq. (2.4.10) :

$$\delta t_{\min} = \mathbf{-0.885} LAT - 0.012 FlightTime + 24.590 \quad (2.4.10)$$

Table 11: SPSS maximum negative deviation multiple regression analysis results

	B	β	t	p	Adjusted R^2
Constant	24.59		52.583	<0.001	0.711

LAT(IV)	-0.885	-0.105	-60.486	<0.001	
FlightTime(IV)	-0.012	-0.825	-475.751	<0.001	

The established maximum positive and maximum negative time deviation multiple regression equations can be used to determine the range of individual trajectory point passing time, thus providing the required input parameters for robust trajectory conflict detection that takes into account the uncertainty of the predicted trajectory.

2.5. INVESTIGATIONS AND RESULTS ON TRAJECTORY OPTIMIZATION

2.5.1. HORIZONTAL PATH OPTIMIZATION

2.5.1.1 MODEL BUILDING

The horizontal aircraft path planning model (HAPPM) is established based on the following assumptions:

(1) The flying speed of the aircraft is constant on the route;

(2) The flying path from one waypoint to another is a great circle route, and the flying time is the length of the flying path divided by the ground speed;

(3) The time-varying nature of the wind at certain waypoints is considered, which is related to the moment of arrival at the waypoint;

(4) The wind speed components of the north-south and east-west directions along the path are the average value of the wind speed components at the start and end of the path segment, without considering the time-varying nature of the wind during the flight path segment;

(5) There is an upper boundary for the total flying time of the flying path, denoted as t_m. This upper limit can help limit the size of the model.

The HAPPM is established with the notations listed in Table 12.

Table 12: Notation for HAPPM formulation

Notation	Description
V	Set of waypoints in the airspace
$Bcon_j$	Set of waypoints from the waypoint j
$Econ_j$	Set of waypoints to the waypoint j
v_s / v_e	The start/end waypoint of the whole path
t_s	Time the flight arrives at the stary waypoint v_s
t_m	The upper boundary of flying time from v_s to v_e (in hours)
c_{ijt}	The flying time from waypoint i to j under the wind forecast for the LAT of t

c'_{ij}	The flying time from waypoint *i* to *j* for the optimized path
T_j	Time the flight arrives at waypoint *j*
δ_{it}	A binary variable, the value of 1 indicates the arrival time at waypoint *i* is within time slot *t* , and 0 otherwise
$\overline{\delta}_{it}$	A binary variable, the value of 1 indicates the arrival time at waypoint *i* is not earlier than *t*, 0 otherwise
$\underline{\delta}_{it}$	A binary variable, the value of 1 indicates the arrival time at waypoint *i* is earlier than *t*+1, 0 otherwise
ζ_{ijt}	Auxiliary variables to transform the model into linear
$T_{ij}^{(0)}$	Auxiliary variables to transform the model into linear
x_{ij}	A binary variable, the value of 1 indicates the route segment to fly from waypoint *i* to *j* is selected to form the optimized route, 0 otherwise

The HAPPM is formulated as Eq. (2.5.1)-(2.5.11). Eq. (2.5.1) minimizes the time arrival at the end point v_e subjected to Constraint (2.5.2)- (2.5.11). Eq. (2.5.2) guarantees the connectivity of the generated flying path, that is, each waypoint (except for the start and the end waypoint) in the generated path is connected with the same number of entry and exit route segments. Eq. (2.5.3) and (2.5.4) limit the segments to or from any waypoint of the generated path at most one. Eq. (2.5.5) and (2.5.6) limit the flight must depart from v_s and arrive at v_e. Eq. (2.5.7) describes the relationship between c'_{ij} and c_{ijt}. The flying time from waypoint i to j depends on the value of the auxiliary binary variables δ_{it}, that is, the flying time is calculated based on the wind under the arrival time at waypoint j. Eq. (2.5.8) and (2.5.9) limit the value of δ_{it}. As stated in the assumptions, there is an upper boundary of the flying time for the whole route, denoted as t_m. Thus, the time-varying nature of the wind forecast will not be considered when the arrival time to one of the waypoints is later than t_m. Eq. (2.5.10) defines the arrival time of the start waypoint as t_s. Eq. (2.5.11) shows that the arrival time of waypoint *j* depends on the arrival time of the start waypoint *i* and the flying time of the route segment from waypoint *i* to waypoint *j*, which is a nonlinear function.

$$\min T_{v_e} \tag{2.6.1}$$

$$\sum_{i \in Econ_j} x_{ij} = \sum_{k \in Bcon_j} x_{jk} \quad \forall j \in V, j \neq v_s, j \neq v_e \tag{2.6.2}$$

$$\sum_{i \in Econ_j} x_{ij} \leq 1 \quad \forall j \in V, j \neq v_s \tag{2.6.3}$$

$$\sum_{k \in Bcon_j} x_{jk} \leq 1 \quad \forall j \in V, j \neq v_e \tag{2.6.4}$$

$$\sum_{i \in Econ_{v_e}} x_{iv_e} = 1 \tag{2.6.5}$$

$$\sum_{k \in Bcon_{v_s}} x_{v_s k} = 1 \tag{2.6.6}$$

$$c'_{ij} = \sum_{t=0}^{t_m - 1} c_{ijt} \delta_{it} \quad \forall i \in V, \forall j \in V, i \neq v_e, j \neq v_s \tag{2.6.7}$$

$$\delta_{it}=\begin{cases}1 & t\le T_i<t+1\\ 0 & otherwise\end{cases}\quad \forall i\in V,\forall t\in Z,0\le t\le t_m-2 \tag{2.6.8}$$

$$\delta_{i,t_m-1}=\begin{cases}1 & T_i\ge t_m-1\\ 0 & otherwise\end{cases}\quad \forall i\in V \tag{2.6.9}$$

$$T_{v_s}=t_s \tag{2.6.10}$$

$$T_j=\sum_{i\in Econ_j}x_{ij}\left(c'_{ij}x_{ij}+T_i\right)\quad \forall j\in V,j\ne v_s \tag{2.6.11}$$

Nonlinear constraints exist in the above model, such as Eq. (2.5.7), (2.5.8), (2.5.9), and (2.5.11). These nonlinear constraints will increase the solution time and cause difficulties in solving the model, especially in the case of large-scale problems with a large number of waypoints. Therefore, a linear transformation of these nonlinear constraints is necessary.

For δ_{i,t_m-1}, it is only necessary to restrict its boundary, i.e.

$$\delta_{i,t_m-1}>\left(T_i-t_m-1\right)/M\quad \forall i\in V \tag{2.6.12}$$

$$\delta_{i,t_m-1}\le 1+\left(T_i-t_m-1\right)/M\quad \forall i\in V \tag{2.6.13}$$

For δ_{it}, both the upper and the lower bound need to be restricted at the same time. In this case, extra binary variables $\overline{\delta}_{it}$ and $\underline{\delta}_{it}$ ($i\in V,t\in Z,0\le t\le t_m-2$) are introduced, satisfying the following constraints.

$$\overline{\delta}_{it}=\begin{cases}1 & T_i\ge t\\ 0 & T_i<t\end{cases}\quad \forall t\in Z,0\le t\le t_m-2 \tag{2.6.14}$$

$$\underline{\delta}_{it}=\begin{cases}1 & T_i<t+1\\ 0 & T_i\ge t+1\end{cases}\quad \forall t\in Z,0\le t\le t_m-2 \tag{2.6.15}$$

Constraints can be reformulated as formula (2.5.16)- (2.5.19) introducing the big M (M represents a positive number that is large enough).

$$\overline{\delta}_{it}>\left(T_i-t\right)/M\quad \forall i\in V,\forall t\in Z,0\le t\le t_m-2 \tag{2.6.16}$$

$$\overline{\delta}_{it}\le 1+\left(T_i-t\right)/M\quad \forall i\in V,\forall t\in Z,0\le t\le t_m-2 \tag{2.6.17}$$

$$\underline{\delta}_{it}\ge\left(t+1-T_i\right)/M\quad \forall i\in V,\forall t\in Z,0\le t\le t_m-2 \tag{2.6.18}$$

$$\underline{\delta}_{it}<1+\left(t+1-T_i\right)/M\quad \forall i\in V,\forall t\in Z,0\le t\le t_m-2 \tag{2.6.19}$$

Then $\delta_{it}=1\Leftrightarrow(\overline{\delta}_{it}=1\wedge\underline{\delta}_{it}=1)$ is reformulated by the following linear constraints:

$$\delta_{it}\le\left(\overline{\delta}_{it}+\underline{\delta}_{it}\right)/2\quad \forall i\in V,\forall t\in Z,0\le t\le t_m-2 \tag{2.6.20}$$

$$\delta_{it} \geq \overline{\delta}_{it} + \underline{\delta}_{jt} - 1 \quad \forall i \in V, \forall t \in Z, 0 \leq t \leq t_m - 2 \tag{2.6.21}$$

For there are nonlinear parts in Eq. (2.5.11), the auxiliary variables ζ_{ijt} and $T_{ij}^{(0)}$ are introduced, defined by Eq. (2.5.22) and (2.5.23).

$$\zeta_{ijt} = \delta_{it} x_{ij} \quad \forall i \in V, \forall j \in V, \forall t \in Z, i \neq v_e, j \neq v_s, 0 \leq t \leq t_m - 1 \tag{2.6.22}$$

$$T_{ij}^{(0)} = x_{ij} T_i \quad \forall i \in V, \forall j \in V, i \neq v_e, j \neq v_s \tag{2.6.23}$$

In this way, Eq. (2.5.11) can be reformulated as Eq. (2.5.24).

$$\begin{aligned} T_j &= \sum_{i \in Econ_j} x_{ij} \left(c'_{ij} x_{ij} + T_i \right) \\ &= \sum_{i \in Econ_j} \left(c'_{ij} x_{ij} + x_{ij} T_i \right) \\ &= \sum_{i \in Econ_j} \left(\sum_{t=0}^{t_m - 1} c_{ijt} \delta_{it} x_{ij} + x_{ij} T_i \right) \\ &= \sum_{i \in Econ_j} \sum_{t=0}^{t_m - 1} c_{ijt} \delta_{it} x_{ij} + \sum_{i \in Econ_j} x_{ij} T_i \\ &= \sum_{i \in Econ_j} \sum_{t=0}^{t_m - 1} c_{ijt} \zeta_{ijt} + \sum_{i \in Econ_j} T_{ij}^{(0)} \end{aligned} \tag{2.6.24}$$

Formula (2.5.22) can be reformulated as the following constraints:

$$\zeta_{ijt} \leq \left(\delta_{it} + x_{ij} \right) / 2 \quad \forall i \in V, \forall j \in V, \forall t \in Z, i \neq v_e, j \neq v_s, 0 \leq t \leq t_m + 1 \tag{2.6.25}$$

$$\zeta_{ijt} \geq \delta_{it} + x_{ij} - 1 \quad \forall i \in V, \forall j \in V, \forall t \in Z, i \neq v_e, j \neq v_s, 0 \leq t \leq t_m + 1 \tag{2.6.26}$$

Formula (2.5.23) can be reformulated as the following constraints:

$$T_{ij}^{(0)} \leq T_i \quad \forall i \in V, \forall j \in V, i \neq v_e, j \neq v_s \tag{2.6.27}$$

$$T_{ij}^{(0)} \geq T_i - M \left(1 - x_{ij} \right) \quad \forall i \in V, \forall j \in V, i \neq v_e, j \neq v_s \tag{2.6.28}$$

$$T_{ij}^{(0)} \leq M x_{ij} \quad \forall i \in V, \forall j \in V, i \neq v_e, j \neq v_s \tag{2.6.29}$$

HAPPM is then reformulated as a mixed integer linear programming model. (Formula (2.5.30) -(2.5.51)).

$$\min T_{v_e} \tag{2.6.30}$$

$$\sum_{i \in Econ_j} x_{ij} = \sum_{k \in Bcon_j} x_{jk} \quad \forall j \in V, j \neq v_s, j \neq v_e \tag{2.6.31}$$

$$\sum_{i \in Econ_j} x_{ij} \leq 1 \quad \forall j \in V, j \neq v_s \tag{2.6.32}$$

$$\sum_{k\in Bcon_j} x_{jk} \le 1 \quad \forall j \in V, j \ne v_e \tag{2.6.33}$$

$$\sum_{i\in Econ_{v_e}} x_{iv_e} = 1 \tag{2.6.34}$$

$$\sum_{k\in Bcon_{v_s}} x_{v_s k} = 1 \tag{2.6.35}$$

$$c'_{ij} = \sum_{t=0}^{t_m-1} c_{ijt}\delta_{it} \quad \forall i \in V, \forall j \in V, i \ne v_e, j \ne v_s \tag{2.6.36}$$

$$T_{v_s} = 0 \tag{2.6.37}$$

$$T_j = \sum_{i\in con_j}\sum_{t=0}^{t_m-1} c_{ijt}\zeta_{ijt} + \sum_{i\in con_j} T_{ij}^{(0)} \tag{2.6.38}$$

$$\zeta_{ijt} \le \left(\delta_{it} + x_{ij}\right)/2 \quad \forall i \in V, \forall j \in V, \forall t \in Z, i \ne v_e, j \ne v_s, 0 \le t \le t_m - 1 \tag{2.6.39}$$

$$\zeta_{ijt} \ge \delta_{it} + x_{ij} - 1 \quad \forall i \in V, \forall j \in V, \forall t \in Z, i \ne v_e, j \ne v_s, 0 \le t \le t_m - 1 \tag{2.6.40}$$

$$T_{ij}^{(0)} \le T_i \quad \forall i \in V, \forall j \in V, i \ne v_e, j \ne v_s \tag{2.6.41}$$

$$T_{ij}^{(0)} \ge T_i - M\left(1 - x_{ij}\right) \quad \forall i \in V, \forall j \in V, i \ne v_e, j \ne v_s \tag{2.6.42}$$

$$T_{ij}^{(0)} \le M x_{ij} \quad \forall i \in V, \forall j \in V, i \ne v_e, j \ne v_s \tag{2.6.43}$$

$$\overline{\delta}_{it} > \left(T_i - t\right)/M \quad \forall i \in V, \forall t \in Z, 0 \le t \le t_m - 2 \tag{2.6.44}$$

$$\overline{\delta}_{it} \le 1 + \left(T_i - t\right)/M \quad \forall i \in V, \forall t \in Z, 0 \le t \le t_m - 2 \tag{2.6.45}$$

$$\underline{\delta}_{it} \ge \left(t + 1 - T_i\right)/M \quad \forall i \in V, \forall t \in Z, 0 \le t \le t_m - 2 \tag{2.6.46}$$

$$\underline{\delta}_{it} < 1 + \left(t + 1 - T_i\right)/M \quad \forall i \in V, \forall t \in Z, 0 \le t \le t_m - 2 \tag{2.6.47}$$

$$\delta_{it} \le \left(\overline{\delta}_{it} + \underline{\delta}_{it}\right)/2 \quad \forall i \in V, \forall t \in Z, 0 \le t \le t_m - 2 \tag{2.6.48}$$

$$\delta_{it} \ge \overline{\delta}_{it} + \underline{\delta}_{it} - 1 \quad \forall i \in V, \forall t \in Z, 0 \le t \le t_m - 2 \tag{2.6.49}$$

$$\delta_{i,t_m-1} > \left(T_i - t_m + 1\right)/M \quad \forall i \in V \tag{2.6.50}$$

$$\delta_{i,t_m-1} \le 1 + \left(T_i - t_m + 1\right)/M \quad \forall i \in V \tag{2.6.51}$$

2.5.1.2 CASE SOLVING

The model is applied in the horizontal flying path of VHHH-EHAM on June 8th, 2019. The flight enters the airspace via PASLU and leaves the airspace at SARIN. The arrival time at PASLU is 00:00 (UTC)and 05 :35(UTC) separately, and the high-altitude wind data at PASLU is UTC+0h and UTC+5h respectively. The flight is flying at 9800m altitude and 920km/h speed. Instantiated with western airspace data, the horizonal path between PASLU and SARIN in the western airspace is optimized with the model. The model is solved by the Gurobi solver with a computer configuration of 128 GB RAM, 2.4 GHz, core

20, logic processor 40, and a termination condition of 12 hours of solution time for the iteration.

The optimized flying path is shown in Table 13 corresponding to different PASLU arrival times, the flying time required for the optimized path is 3.1258 hours and 3.185 hours separately. Both the planned route and the two optimized route is visually shown in Figure 18.

Table 13: Optimized flight path for different PASLU arrival time

Point serial of optimized path 1	Longitude (E)	Latitude (N)	Point serial of optimized path 2	Longitude (E)	Latitude (N)
PASLU	105.0305557	28.48083305	PASLU	105.0305557	28.48083305
D37	103.4891896	28.52886428	D37	103.4891896	28.52886428
D65	102.3370287	29.8842799	D65	102.3370287	29.8842799
D84	101.5682496	30.79322737	D84	101.5682496	30.79322737
D103	99.92427003	32.28993945	D101	100.1803282	32.38031312
LUVAR	99	32.601665	D132	94.21759159	38.43718798
D135	90.05205994	38.23767851	D179	92.47362054	39.98832194
D205	89.40208865	40.53683717	D201	90.45298416	41.67030339
D217	86.90407261	42.76357784	D222	87.98275135	43.59906044
D238	84.88285816	44.27091775	D219	88.08282347	43.75324174
TCH	83.34388733	46.66916656	P221	86.28333282	45.125
SARIN	82.88833618	46.86500168	TCH	83.34388733	46.66916656
			SARIN	82.88833618	46.86500168

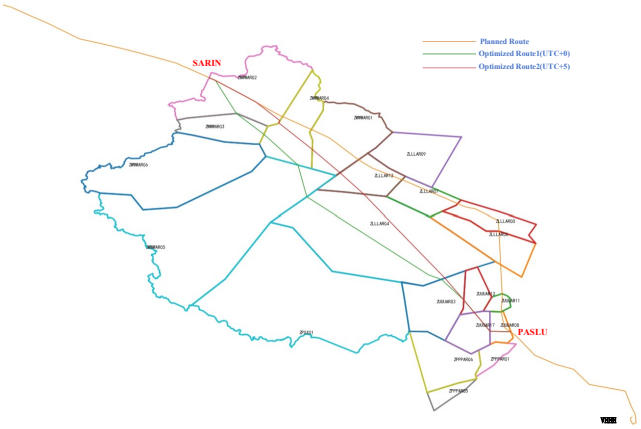


Figure 18: The optimized horizontal flying path

2.5.2. VERTICAL PROFILES OPTIMIZATION

In the aspect of trajectory vertical profile optimization model construction, considering the influence of high-altitude wind on the trajectory and the constraints of dynamics, we study the optimization of the flight trajectory under the influence of wind in two flying scenarios: flying at any selected altitude and flying by flight level. The optimal control model to optimize the complete trajectory of "runway-runway" is established with optimization objectives of minimizing both the flight time and the fuel consumption.

In terms of solving the constructed model, the model is firstly transformed into a Nonlinear Programming (NLP) problem by using the collocation point method, and the Pareto front solution is obtained by using the constraint method to solve the multi-objective NLP model, and considering the effect of the initial solution on the NLP solver, a two-stage solution method is designed for the flight scenario by flight level to avoid the locally optimal problem.

2.5.2.1 MODEL BUILDING

The trajectory optimization model considering the wind impact is established based on the optimal control framework. The overall framework of the model and the required basic models are shown in Figure 19.

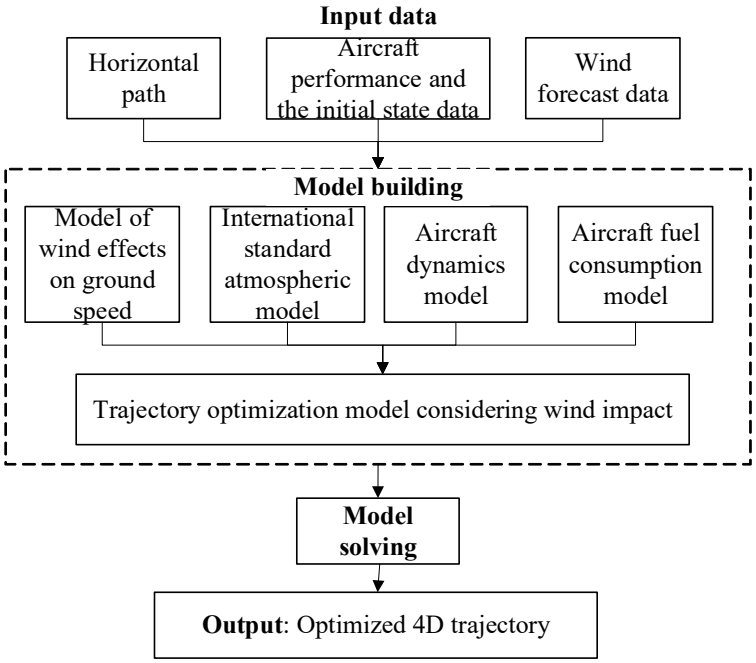


Figure 19: 4D trajectory optimization model framework

The wind-to-ground speed impact model, aircraft fuel consumption model, aircraft dynamics model involved in the trajectory simulation model and the trajectory optimization model itself are separately described as follows:

1) Model of wind effects on ground speed

During operation, the influence of wind on aircraft cannot be ignored, especially during the cruising phase. The flying time and fuel consumption of the aircraft are both related

to the wind. To obtain the optimal four-dimensional trajectory, the influence of wind on the trajectory must be taken into account by establishing the wind effects model.

According to the wind triangle principle, ground speed $\overrightarrow{V_{GS}}$ is calculated by true airspeed $\overrightarrow{V_{TAS}}$ and wind speed $\overrightarrow{V_W}$.

$$\overrightarrow{V_{GS}} = \overrightarrow{V_{TAS}} + \overrightarrow{V_W} \tag{2.6.52}$$

In the following formula (2.5.53), the Wind vector can be divided into a downwind parallel to the ground trajectory V_{Ws} and a crosswind perpendicular to the ground trajectory V_{Wx}. As shown in Figure 20, the ground speed calculation equation is expressed as follows:

$$V_{GS} = \sqrt{V_{TAS}{}^2 - V_{Wx}{}^2} \pm V_{Ws} \tag{2.6.53}$$

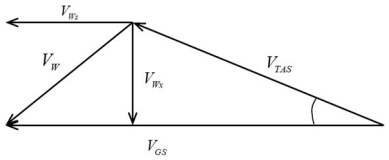


Figure 20: Schematic diagram of the effect of side wind on the trajectory

2) Aircraft fuel consumption model

The fuel flow rate model in BADA is used to calculate the fuel flow rate for jet aircraft.

The fuel flow rate formulate for all flight phases except the cruising phase and idle thrust phase is as follows:

$$\begin{cases} \eta = C_{f1}\left(1 + \dfrac{v}{C_{f2}}\right) \\ FF = \eta \cdot Thr \end{cases} \tag{2.6.54}$$

In equation(2.5.54): η is fuel consumption per unit of thrust; C_{f1} , C_{f2} is the thrust-specific fuel consumption coefficient; FF is the fuel flow rate, and Thr is thrust.

The fuel flow rate in the Cruising phase without considering the idle thrust is calculated as follows:

$$FF_{cr} = \eta \cdot Thr \cdot C_{fcr} \tag{2.6.55}$$

As is shown in equation(2.5.55), C_{fcr} is the cruising fuel flow rate factor.

In the above formula, the coefficient values of C_{f1}, C_{f2}, C_{fcr} can be obtained from the BADA database according to the aircraft type.

3) Aircraft Dynamics Model

The full energy model equalizes the power made by the force acting on the aircraft to the increased rate of potential energy and kinetic energy, i.e.:

$$(Thr - D) \cdot v = mg \cdot \frac{dh}{dt} + mv \cdot \frac{dv}{dt} \tag{2.6.56}$$

In equation (2.5.56)**Error! Reference source not found.**: Thr is thrust; D is the drag; v is true airspeed; m is aircraft mass; g is the acceleration of gravity; and d/dt represents the differential of time. To simplify the force analysis of aircraft, according to the point mass model, it is assumed that all the forces act on the center of gravity of the aircraft, and only the vertical profile of the aircraft is considered (the horizontal route is optimized by the method in 2.5.1). So that the motion of the aircraft is reduced to three degrees of freedom, and the variation of all three degrees of freedom can be described as a differential equation about time.

D can be calculated by the equation as follows:

$$D = \frac{S}{2} C_D \rho v^2 \tag{2.6.57}$$

In equation(2.5.57): S is the reference wing area; C_D represents the resistance coefficient; ρ is air density; and C_D is a function of climb coefficient C_L. C_L and C_D are calculated by equations as follows:

$$C_L = \frac{2mg}{\rho v^2 S \cos\phi} \tag{2.6.58}$$

$$C_D = C_{D0} + C_{D2} C_L^2 \tag{2.6.59}$$

In equations (2.5.58)**Error! Reference source not found.**-(2.5.59): ϕ is bank angle of aircraft, with the value of 0 in this project; C_{D0} is coefficient of parasite drag; C_{D2} is coefficient of induced drag.

In the atmospheric state without considering the wind, the aircraft's ground speed is equal to the true airspeed, and its dynamics model can be represented by the following set of differential equations:

$$\frac{dv}{dt} = \dot{v} = \frac{1}{m}(Thr - D - mg\sin\gamma) \tag{2.6.60}$$

$$\frac{ds}{dt} = \dot{s} = v\cos\gamma \tag{2.6.61}$$

$$\frac{dh}{dt} = \dot{h} = v\sin\gamma \tag{2.6.62}$$

$$\frac{dm}{dt} = \dot{m} = -FF \tag{2.6.63}$$

In equations(2.5.60)**Error! Reference source not found.**-(2.5.63)**Error! Reference source not found.**: v is true airspeed; s is the distance flown by the aircraft on the horizontal path; h is flying altitude; m is aircraft mass; Thr is thrust; γ is flight angle of climb; FF is the fuel flow rate.

Combined with wind model, equation**Error! Reference source not found.** (2.5.61)**Error! Reference source not found.** is calibrated as equation(2.5.64)**Error! Reference source not found.**:

$$\frac{ds}{dt} = \dot{s} = \sqrt{(v\cos\gamma)^2 - {V_{Wx}}^2} \pm V_{Ws} \tag{2.6.64}$$

In equation (2.5.64)：W_x is the wind component perpendicular to the flight path; W_s the wind component along the flight path.

4) Trajectory optimization model with consideration of wind

According to the basic model mentioned above, aircraft speed, altitude, thrust, etc. can be expressed as functions of time. The aircraft operation process can be considered as a controlled dynamic system, and the aircraft 4D trajectory optimization problem can be formulated as a dynamic optimal control problem, to better model the nonlinear dynamics equations and nonconvex objective functions.

To simplify the problem, the established trajectory optimization model is based on the following assumptions :

- Aircraft is in a clean configuration.
- Excluding human factors such as controller intervention on the aircraft;
- Without considering Terminal Area's STAR and STD procedures.

The optimization objective of the optimal control model is defined by the metrics that can evaluate the effectiveness of the control, in this case, the flying time J_1 and the fuel consumption J_2 are used as the optimization objectives, i.e.：

$$\min J_1 = t_f - t_0 \tag{2.6.65}$$

$$\min J_2 = \int_{t_0}^{t_f} FF(t)dt \tag{2.6.66}$$

In the above equation: t_0 is the starting moment of the optimization process, i.e. the departure time of the aircraft i; t_f is the end time of the optimization process, that is, the moment when the aircraft landed.

The model constraints are as follows：

(1)Equation of state constraints

The equation of state is described by a set of differential equations, that is, the kinetic equations described by formulas **Error! Reference source not found.**(2.5.60), (2.5.62)**Error! Reference source not found.**-(2.5.64)**Error! Reference source not found.**. Among them, $[Thr, \gamma]$ is the control vector of the model and $[v, s, h, m]$ is the state vector of the model. All are functions of time t.

(2)Two-end constraints

Two-end constraints are used to represent the state limit constraints at the initial and final moments of the aircraft, including the following constraints：

a) Speed state constraint

It is stipulated that the speed of the aircraft at the time of off-ground and at the time of on-ground is 1.3 times the stall speed, i.e.：

$$v(t_0) = v(t_f) = 1.3 \cdot v_{stall} \tag{2.6.67}$$

b) Distance state constraint

It is stipulated that the horizontal distance flown by aircraft at the time of departure from the ground is 0, and the whole range horizontal distance is flown at the time of landing, i.e.:

$$s(t_0) = 0, s(t_f) = S \tag{2.6.68}$$

In equation (2.5.68): S indicates the whole distance flown by aircraft on the horizontal path.

c) Flying altitude state constraints

It is specified that the altitude of the aircraft at the time both off-ground and on-ground is 0, that is:

$$h(t_0) = h(t_f) = 0 \tag{2.6.69}$$

d) Weight state constraints

The weight of the aircraft at the time of departure from the ground is specified as the aircraft take-off weight, i.e.:

$$m(t_0) = m_{take-off} \tag{2.6.70}$$

(3) Control constraints

The control constraint is the constraint on the range of values of the aircraft control vector based on the aircraft performance, i.e.:

$$Thr(t) < Thr_{\max}(h(t)) \tag{2.6.71}$$

$$\gamma_{\min} < \gamma(t) < \gamma_{\max} \tag{2.6.72}$$

Equation (2.5.71) indicates that during flight the aircraft thrust should not exceed the maximum thrust corresponding to the current altitude. Equation(2.5.72) Indicates that the AOC(Angle of Climb) of the aircraft should not exceed the range of available AOC.

(4)Flight level constraint

Aircraft in the cruising phase must fly at the specific flight level according to the flight level regulations, to meet the operational requirements, namely:

$$h \in \begin{cases} FL_{\text{east}}, aircraft \text{ fly eastward} \\ FL_{\text{west}}, aircraft \text{ fly westward} \end{cases} \tag{2.6.73}$$

In formula (2.5.73), FL_{east} is the collection of optional flight levels for flights flying eastward, and FL_{west} is the collection of optional flight levels for flights flying westward.

2.5.2.2 MODEL SOLVING

The solution methods for optimal control problems are usually divided into two kinds of methods, indirect and direct methods. The indirect method in optimal control is generally derived by using the variational method and the minimum principle, which requires a

prior guess of the structure of the solution. For the optimal control problem of aircraft trajectory, there are complex coupling relationships between the various state variables of the controlled system, which makes it difficult to apply the indirect method to solve this problem. So the direct method is used here, and the optimal control problem is discretized into an NLP problem using the trapezoidal collocation method before solving. Firstly, determine the number of collocation points N, then the trajectory is discretized to obtain the set of a finite number of decision variables at each collocation point. And the continuous system dynamics equations in **Error! Reference source not found.** (2.5.60), (2.5.62)**Error! Reference source not found.**-(2.5.64) are discretized into a set of nonlinear constraints using the trapezoidal collocation method, such that the change of state between two collocation points is equal to the integral of the continuous dynamics. Take equation**Error! Reference source not found. Error! Reference source not found.**(2.5.63)as an example：

$$\int_{t_k}^{t_{k+1}} \dot{m} dt = \int_{t_k}^{t_{k+1}} -FF dt \tag{2.6.74}$$

$$m_{k+1} - m_k = \frac{\Delta t_k}{2}\left[\left(-FF_{k+1}\right)+\left(-FF_k\right)\right] \tag{2.6.75}$$

Equation (2.5.74) is got **Error! Reference source not found.**after the integration of both sides of the equation (2.5.63). The right side of the equation (2.5.75) is the integral approximation using the trapezoidal formula where $\Delta t_k = t_{k+1} - t_k$. Based on the same idea, each state equation is discretized by using the trapezoidal collocation method, expressed as follows：

$$v_{k+1} - v_k = \frac{\Delta t_k}{2}\left[\left(\frac{Thr_{k+1} - D_{k+1}}{m_{k+1}} - g\sin\gamma_{k+1}\right)+\left(\frac{Thr_k - D_k}{m_k} - g\sin\gamma_k\right)\right] \tag{2.6.76}$$

$$s_{k+1} - s_k = \frac{\Delta t_k}{2}\cdot\left[\left(\sqrt{\left(v_{k+1}\cos\gamma_{k+1}\right)^2 - W_x(s_{k+1},h_{k+1})^2} + W_s(s_{k+1},h_{k+1})\right)+\left(\sqrt{\left(v_k\cos\gamma_k\right)^2 - W_x(s_k,h_k)^2} + W_s(s_k,h_k)\right)\right] \tag{2.6.77}$$

$$h_{k+1} - h_k = \frac{\Delta t_k}{2}\left[\left(v_{k+1}\sin\gamma_{k+1}\right)+\left(v_k\sin\gamma_k\right)\right] \tag{2.6.78}$$

The four sets of constraints from equation (2.5.76) to equation (2.5.78) ensure that the variation of the state vector $[v, s, h, m]$ at the discrete points satisfies the state equation constraints.

Correspondingly, the optimization objectives in the optimal control problem are described as:

$$\min J_1 = t_N - t_0 \tag{2.6.79}$$

$$\min J_2 = \sum_{k=1}^{N}\frac{1}{2}\left(FF_k + FF_{k+1}\right)\cdot\Delta t_k \tag{2.6.80}$$

The control constraints are described as：

$$Thr_{1\ldots N} < Thr_{\max}(h_{1\ldots N}) \tag{2.6.81}$$

$$\gamma_{\min} < \gamma_{1\ldots N} < \gamma_{\max} \tag{2.6.82}$$

To ensure the logical correctness between the discrete decision variables, the following constraint is added：

$$s(t_{k+1}) > s(t_k) \tag{2.6.83}$$

$$m(t_{k+1}) < m(t_k) \tag{2.6.84}$$

$$t_{k+1} > t_k \tag{2.6.85}$$

Equation(2.5.83) indicates that the horizontal distance flown by aircraft during the flight always increases. Equation(2.5.84) indicates that the fuel consumption is always positive during the flight, i.e., the mass of the aircraft is decreasing. Equation(2.5.85) indicates the time increasing progressively at discrete points.

After transforming the continuous optimal control problem into an NLP problem, the non-linear programming solver can be used to solve the model. Commonly used nonlinear programming solvers can generally only solve the single-objective problem, so a method for transforming multi-objective optimization problems into single-objective optimization problems is required, including the linear weighting method, $\varepsilon-constraint$ method and so on. For the 4D trajectory optimization problem, the feasible domain space is complex, and using the traditional linear weighting method may converge to local optimality, and no Pareto Front solution can then be obtained, while the $\varepsilon-constraint$ method narrows the range of feasible solutions and reduces the computational complexity by restricting the values of other objective functions with the constraints, and the Pareto Front solution can be obtained. Therefore, the $\varepsilon-constraint$ method is adopted. The fuel consumption is treated as the optimization objective, while the flight time objective is treated as the constraint, i.e., the model objective function is：

$$\min J = \sum_{k=1}^{N} \frac{1}{2}\left(FF_k + FF_{k+1}\right)\cdot \Delta t_k \tag{2.6.86}$$

Adding constraints at the same time：

$$t_N \le T_{cons} \tag{2.6.87}$$

In the above formula (2.5.87)： T_{cons} indicates the set flight time. The original multi-objective optimization problem is then transformed into a single-objective optimization problem, and the Pareto Front will be obtained by solving the model multiple times with a changed value of T_{cons} .

In addition, the commonly used nonlinear programming solvers are sensitive to the initial solution. For the 4D trajectory optimization problem, the initial solution can greatly influence the model-solving time and the optimization result. For the scenario that flight flies by flight level, the model solving is performed in two stages. In the first stage, the flight level constraint described in Eq. (2.5.73) is not considered, and the optimized 4D

trajectory is obtained for the freely selected altitude scenario. Then in the second stage, the optimal solution obtained in the previous stage is taken as the initial solution and Eq. (2.5.73) is added to the model. The model is solved again to obtain the optimal solution of the trajectory optimization problem for flying by flight level scenario.

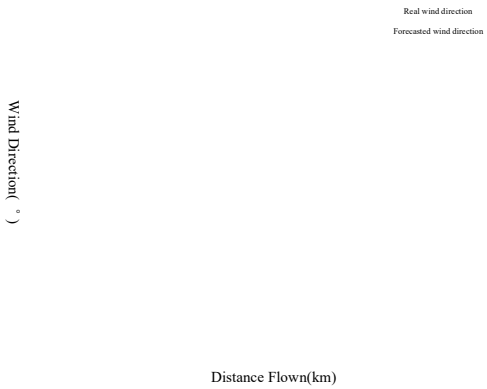
2.5.3. TRAJECTORY OPTIMIZATION EFFECT SIMULATION VERIFICATION

Using the high-altitude wind forecast data, taking the Sino-European long-haul flight as an example, the simulated optimized trajectory and planned trajectory are and compared to verify the optimization effect of the trajectory optimization method. To make the verification closer to reality, firstly, the accuracy of both the wind forecast data relative to the "real" wind and the trajectory simulation model is verified. On this basis, the simulation model is used to simulate the optimized trajectory to verify the rationality, effectiveness, efficiency and optimization effect of the proposed trajectory optimization method.

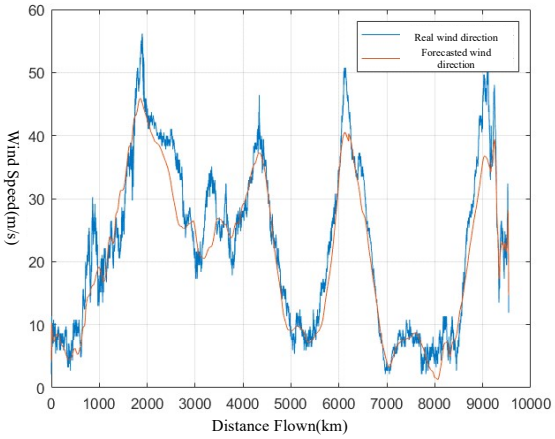
2.5.3.1 WIND FORECAST DATA VERIFICATION

Trajectory optimization is performed in advance of flight departure based on forecasted high-altitude winds, and optimized trajectory flying in actual high-altitude winds. There is some discrepancy between the predicted and actual high-altitude winds. If the discrepancy is too drastic, the actual effectiveness of the optimized trajectory will be affected, so the accuracy of the wind forecast needs to be verified.

Actual wind data is got from the Quick Access Recorder (QAR) for flight CSN307 departing at 16:05 (UTC) on 2019-06-07. Obtain the wind forecast data of the grid of the same latitude, longitude, and altitude as that of the QAR and compare it with the recorded QAR wind data. The result is shown in Figure 21. In general, the difference in average wind speed between QAR data and forecast data is 1.84 m/s, and the average wind direction difference is 3.19°. During the cruising phase, which is highly influenced by high-altitude winds, the average wind speed difference is 1.66 m/s and the average wind direction difference is 3.85°. This indicates that the forecast data adopted has high accuracy and the optimized trajectory based on the forecasted high-altitude wind forecast can be operated in actual high-altitude winds with good performance.



(a) Wind direction Comparison



(b) Wind Speed Comparison

Figure 21: ECMWF Forecast wind comparison with actual wind

2.5.3.2 TRAJECTORY SIMULATION MODEL VALIDATION

Since simulation is used to obtain flight time and fuel consumption for comparison optimization effect of the wind-optimized trajectory with the initial planned trajectory , the effectiveness of the trajectory simulation model needs to be verified.

Take flight KLM888 as the simulation object, its planned flight time is 11h14min. Set aircraft take-off weight as 370 tons, and ground release speed as 90m/s. The simulation results show that it took 11h19min flying time and 118.92 tons of fuel for the aircraft from off-ground to on-ground, and the 4D trajectory obtained is shown in Figure 22. The difference between the simulated flight time and the planned flight time is 5min, accounting for 0.73% of the total flight time. It is shown that the adopted the 4D trajectory simulation method is almost consistent with the initial planned flight time and can be used to simulate 4D trajectory of the flight for calculating and making an assessment of the trajectory-related metrics.

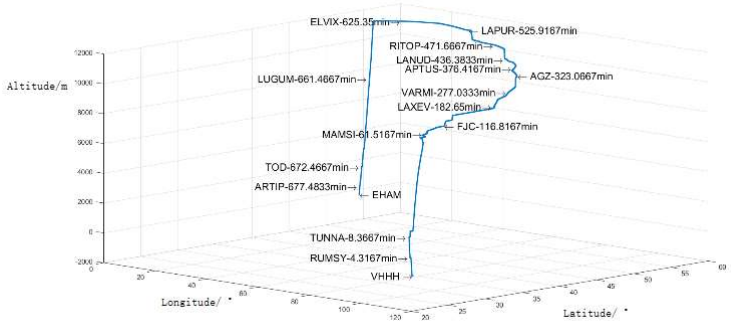


Figure 22: Flight Plan 4D Trajectory Simulation Results

2.5.3.3 TRAJECTORY OPTIMIZATION EFFECT SIMULATION VERIFICATION ANALYSIS

The long-haul flight from Hong Kong to Amsterdam (VHHH-EHAM) is selected for the verification which is greatly affected by the west wind at the mid-latitude(30 ° N-60 ° N). The whole flying range is 9,969.9km, and the optimized Horizontal Route2 generated in 3.5.1. is used as the input for optimization of the vertical profiles. The four-dimensional

flight trajectory optimization does not consider the operation details at the airport, and regards the take-off and landing airport as the same nodes as the route point. It is assumed that the aircraft is not restricted by the standard approach and departure procedures in the terminal area, and the cruising altitude is not lower than 8400m. For the flight flying by flight level scenario, the flight level adopted is in metric units, and the problem of imperial units and metric units' conversion for flight level in different flight information regions is not considered. The chosen flight KLM888 took off at 12:20 Beijing time on June 8, 2019. The aircraft type is B747-400. The flight performance database used is BADA3.11, and the planned trajectory of KLM is used for comparison with the optimized trajectory.

(1) Verification of trajectory simulation effects under ensemble member 0 of the wind ensemble forecast

Distinguished by whether to consider the flight level restrictions, the Pareto front solution is obtained by the $\varepsilon-constraint$ method for the two scenarios of flying at any selected altitude and flying by flight level, respectively. The adjustment step of the parameter ε is 0.5min. Experiments using the computer with Intel Core i7-10700 2.90GHz processor, 16GB RAM, Windows 11 OS. Solved using the function 'fmincon' in MATLAB R2019b. The average solving time is 151 seconds for every single model. A comparison of the Pareto front results obtained for each of the two cases is shown in Figure 23. The top leftmost position in the Pareto frontier represents the shortest flight time trajectory, the bottom rightmost position represents the minimum fuel consumption trajectory, and the remaining points indicate the optimal trajectory obtained under different ε values for restricting the maximum flight time. And the planned trajectory simulation results are at the positions marked by the black dots in the figure.

It can be seen that flying by flight level compresses the searching space of the 4D trajectory due to the additional flight level restriction. The flight time and fuel consumption data for the minimum flight time trajectory and minimum fuel consumption trajectory of the two scenarios and the planned trajectory are all shown in Table 14. Compared to planned trajectory results, the minimum fuel consumption was reduced by 7.8% and 6% for flying at any altitude and flying by flight level respectively, and the minimum flight time was both reduced by 10.9%.

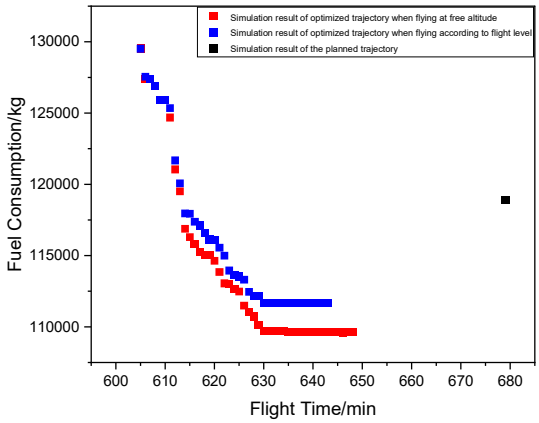


Figure 23: KLM888 Flight Trajectory Optimization Pareto Frontier Solution

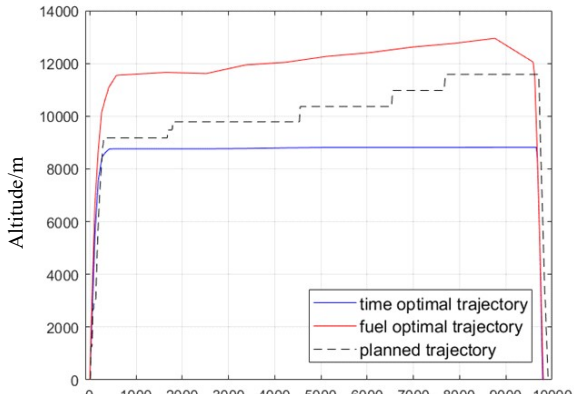
Table 14: Comparison of 4D trajectory optimization results for flight KLM888

	Flight time/min	Fuel

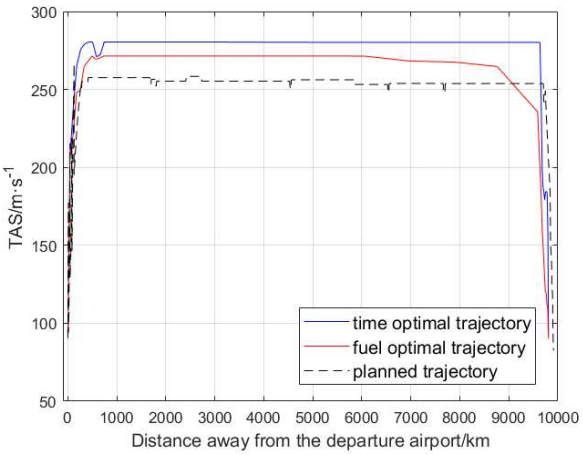
		consumption/kg
Minimum flight time trajectory for the scenario of flying at any selected altitude	605	129520.11
Minimum fuel consumption trajectory for the scenario of flying at any selected altitude	643	109630.84
Minimum flight time trajectory for the scenario of flying by flight level	605	129520.06
Minimum fuel consumption trajectory for the scenario of flying by flight level	643	111673.48
Planned Trajectory	679	118920

(2) Results analysis for the scenario of flying at any selected altitude

The altitude profiles and TAS profiles for minimum flight time trajectory, minimum fuel consumption trajectory, and flight plan simulation trajectory for the scenario of flying at any selected altitude are shown in Figure 24. It is shown that the cruising phase accounts for the largest proportion of flight time and fuel consumption of the whole flying stage, which reaches more than 90%. The lowest possible altitude (8,738m) was selected for the minimum flight time trajectory as the altitude for the cruising phase, which is the crossover altitude corresponding to the maximum Mach number (M0.92) and the maximum IAS (365kt) of the B747-400. At this altitude, the aircraft obtains the maximum TAS. The minimum fuel consumption trajectory is made by taking into account fuel consumption on the climbing phase, 11,600m was chosen as the initial flight level for the cruising phase and the long-range speed of the current flight level was selected as the cruising speed. As fuel is consumed, the mass of the aircraft gradually becomes lighter and the cruising altitude increases accordingly. The reason for this is that with the increase in altitude, air density decreases, the drag force decreases, and the miles that can be flown per unit of fuel consumed increases. The optimization of the trajectory by the freely chosen altitude does not consider the flight level constraint, so the minimum fuel consumption trajectory adopts a variable cruising altitude with a small angle for climbing, instead of the step climbing cruising currently used in actual air traffic operation.



(a) Altitude profile

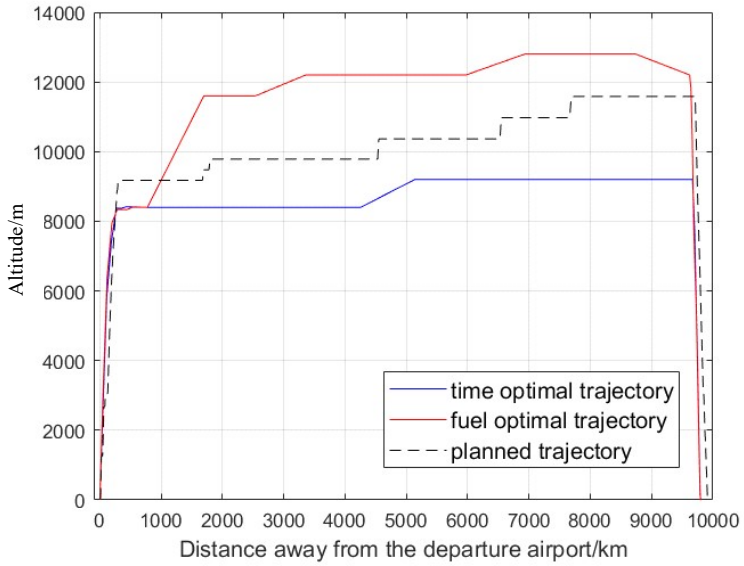


(b)TAS profile

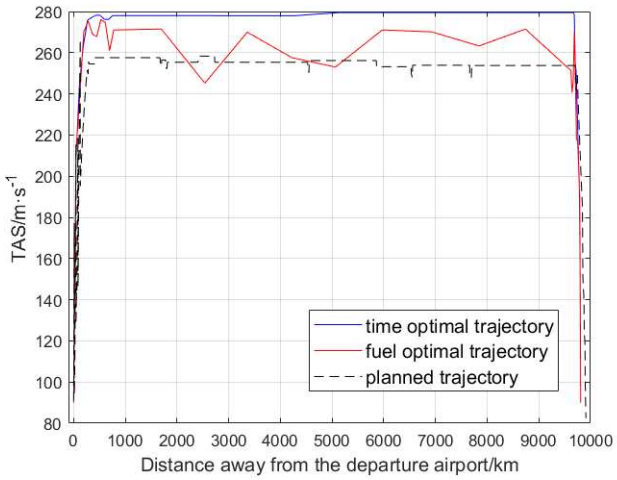
Figure 24: Comparison of optimal trajectory profiles for random flight level

(3) Results analysis for the scenario of flying by flight level

The altitude profile and TAS profile of the minimum flight time trajectory, minimum fuel consumption trajectory and flight plan simulation trajectory for the scenario of flying by flight level are shown in Figure 25. As is shown in the figure, the minimum flight time trajectory chooses the 8400m flight level closest to the 8,738m optimal altitude in the first half of the flight(before the flying distance of 4255m), and the 9200m flight level in the second half of the flight and maintains the maximum flight speed throughout the whole flight stage. The minimum fuel consumption trajectory selects 11,600m as the cruising altitude after a short stay at 8400m, which is the same as the optimization result of flying at any selected altitude scenario. The difference is that due to the flight level constraint, the trajectory takes the way of step-climbing cruising rather than the theoretically optimal small angle climbing altitude varying cruising. The comparison with the flight plan trajectory simulation data shows that the flight plan trajectory is closer to the minimum fuel consumption trajectory than to the minimum flight time trajectory, which indicates that the airline makes the flight plan with a lower cost index by giving more weight to the unit fuel costs.



(a)Altitude profile



(b)TAS profile

Figure 25: Comparison of optimal flight path profile by flight level

(4) Optimal trajectory selection based on the cost

Airlines usually determine the optimal flight trajectory according to the minimum total flight cost from the perspective of economic optimization. The total flight cost is composed of time cost and fuel cost. Referring to the European Airline Delay Cost Reference Values document, aircraft strategic cost is divided into fuel cost and time cost, in which time cost includes: 1) maintenance cost; 2) fleet cost; 3) crew cost. The fuel cost is mainly related to the oil price. The time cost can be queried in the above delay cost reference document concerning the B744 aircraft model for flying the flight between the Sino-European cities. As shown in Table 15, the various time costs of the B744 model in the basic scenario are listed in the table. It can also be adjusted according to the actual situation of the airlines.

Table 15: Composition of time cost for the aircraft B744

	Cost value(€/h)	Cost value（€/min）
Maintenance cost	1610	26.8
Fleet cost	1090	18.2
Crew cost	870	14.5
Total cost	3570	59.5

Cost Index (CI) is the ratio of unit time cost to unit fuel cost. The two indexes of fuel consumption and flying time are unified through flight cost. The calculation formula of flight cost without considering fixed cost is flight cost = oil price * fuel consumption + unit time cost * flight time = oil price * (fuel consumption + CI* flight time). Set the CI index range as between 60kg/min-100kg/min. Based on the multi-objective trajectory optimization Pareto frontier got under ensemble member 0 wind forecast data for both flying at free altitude and flying by flight level scenarios in Figure 23, the trajectory of the minimum cost is separately calculated with different CI values within the set CI value range. The minimum cost trajectory is the same for different CI values when in the range from 60kg/min to 100kg/min. The flying time and fuel consumption of the cost-optimized trajectory are listed in Table 16.

Table 16: The minimum cost trajectory for CI in the range of (60-100)kg/min

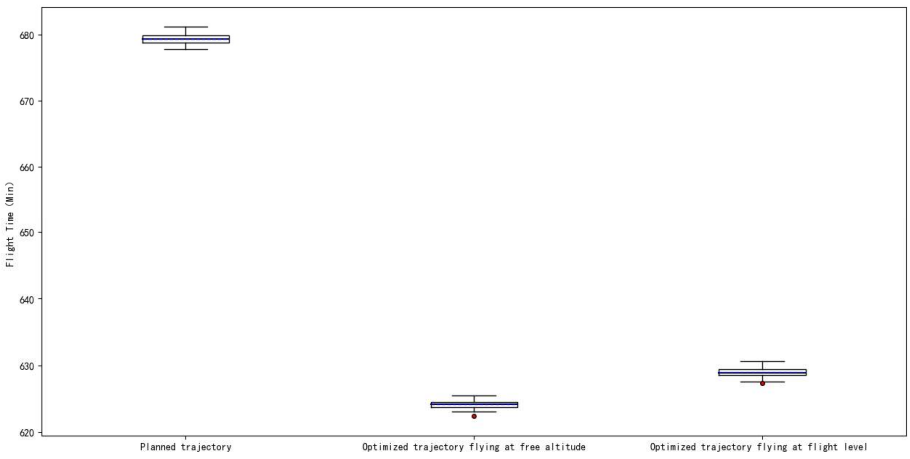
Scenario	CI (kg/min)	Flying time(min)	Fuel consumption（kg）
Flying at Free altitude	60-100	630	109747.112731926
Flying by flight level	60-100	630	111673.492883

(5) Verification of the trajectory simulation metric under each member of the wind ensemble forecast

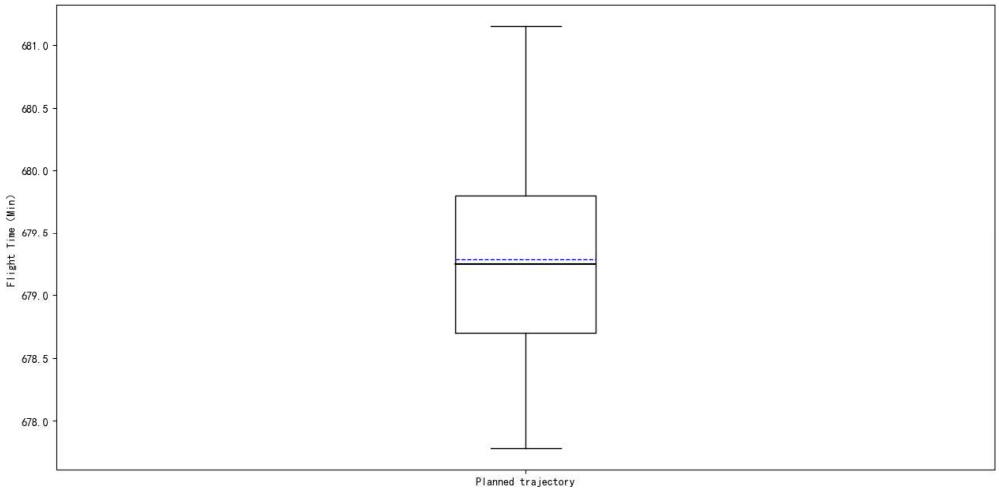
The flight time and fuel consumption are calculated by simulation for verifying the optimization effect. Simulation under ensemble member 0 of the wind ensemble forecast, the flight time, and fuel consumption simulation results of planned trajectory and optimized trajectory are listed in Table 17. Simulation under 51 ensemble members of the wind ensemble forecast, the flight time simulation results of planned trajectory and optimized trajectory are shown in Figure 26, and the flight fuel consumption simulation results of planned trajectory and optimized trajectory are shown in Figure 27.

Table 17: Simulation results under ensemble member 0

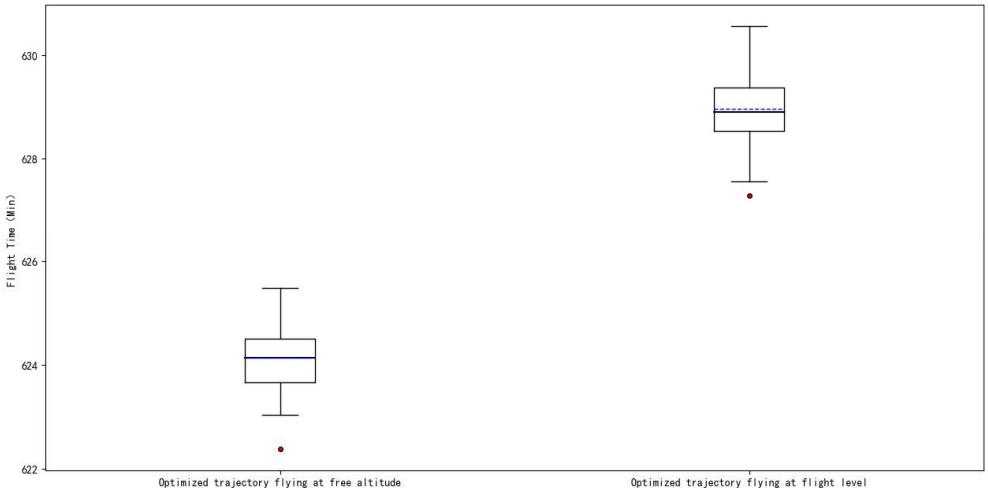
	Flying time(min)	Fuel consumption（kg）
Planned trajectory	679.28	118931.7289
The optimized trajectory for flying at the freely chosen altitude	624.27	114020.4104
The optimized trajectory for flying by flight level	629.15	116854.1591



(a) Flight time distribution for both the planned and optimized trajectories

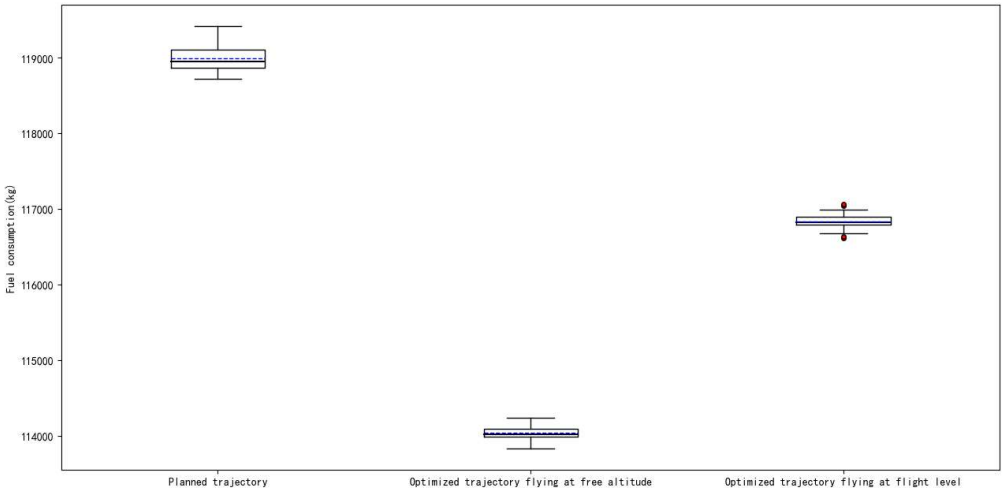


(b) Flight time distribution for the planned trajectory

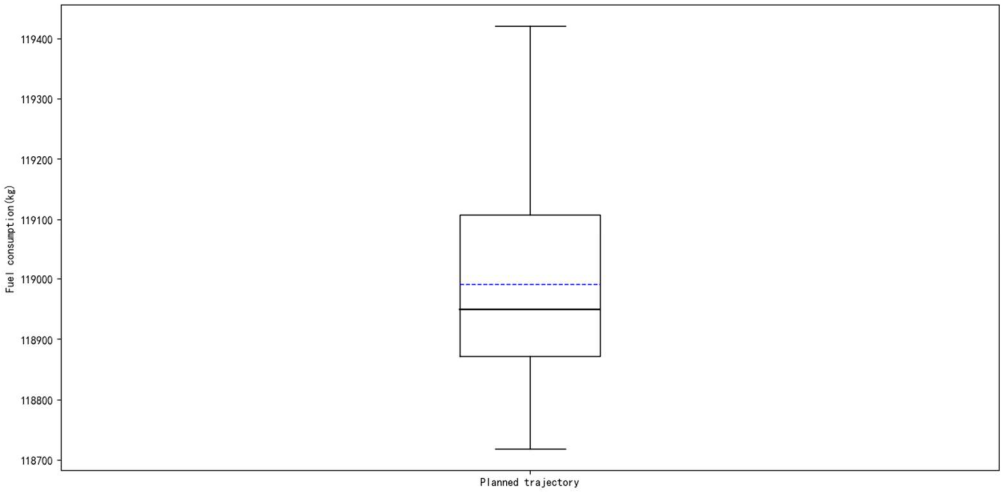


(c) Flight time distribution for the optimized trajectories

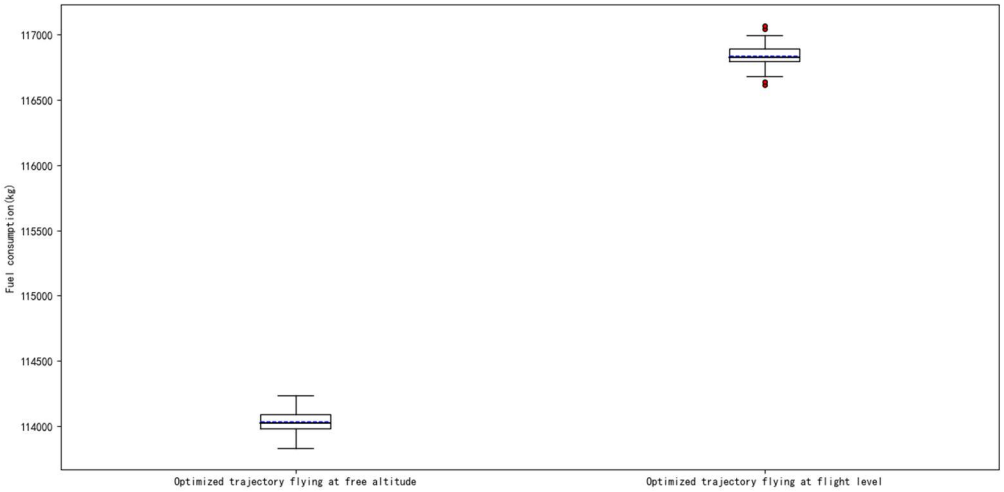
Figure 26: Simulated flight time distribution results under 51 ensemble members



(a) Fuel consumption distribution for both the planned and optimized trajectories



(b) Fuel consumption distribution for the planned trajectory



(c) Fuel consumption distribution for the optimized trajectories

Figure 27: Simulated fuel consumption distribution results under 51 ensemble members

Compared with the originally planned trajectory, the maximum and minimum decreasing percentages of flight time and fuel consumption are shown in Table 18.

Table 18: Decreasing percentage in flight time and fuel consumption

	Decreasing percentage of flying at the freely chosen altitude		Decreasing percentage of flying by flight level	
	Flight time	Fuel consumption	Flight time	Fuel consumption
Maximum value	8.3%	4.4%	7.7%	2.1%
Minimum value	7.9%	3.9%	7.2%	1.6%
Average value	8.1%	4.2%	7.4%	1.8%

2.6. INVESTIGATIONS AND RESULTS ON TRAJECTORY CONFLICT DETECTION

2.6.1. DETERMINISTIC TRAJECTORY CONFLICT DETECTION METHOD

The conflict detection methods studied include conflict detection method based on grid and conflict detection methods based on conflict types for both structured airspace and free-route airspace.

2.6.1.1 DETERMINISTIC CONFLICT DETECTION METHOD BASED ON GRIDS

The traditional pair-wise conflict detection method can be used in the free route airspace scenario, which determines whether there is a conflict by calculating the distance between the sampling points of each flight's trajectory. For n flight trajectories, the pair-wise conflict detection method needs to compare $N(N-1)/2$ trajectories in pairs. In the actual conflict detection scene, considering that some trajectories have large intervals in time and space dimensions, and there is no possibility of conflict, to improve the efficiency of conflict detection for large-scale flights and reduce unnecessary computing time, a conflict detection algorithm based on four-dimensional grids is adopted. First, the airspace is discretized using a four-dimensional grid (3D space +time) as shown in Figure 28. The four-dimension grid can be seen as a time series of 3D grids sampled with a discretization time step Δt . The size of each cell in the 3D grids is defined by the minimum safety separation requirements with horizontal separation N_h as 5 nautical miles and vertical separation N_v as 1000 feet. Identify the grid cells corresponding to each sampling point of each trajectory, that is , map to the corresponding grid cell $C_{m.n.l.k}$ according to its four-dimensional coordinates $(x_{i,j}, y_{i,j}, z_{i,j}, t_{i,j})$ for the point $P_{i,j}$ which is the j discrete sampling point of the trajectory i . The index number of a grid cell $C_{m.n.l.k}$ is calculated as follows.

$$m = \left\lceil \frac{x_{i,j} - x_{\min}}{N_h} \right\rceil + 1 \qquad (2.6.1)$$

$$n = \left\lceil \frac{y_{i,j} - y_{\min}}{N_h} \right\rceil + 1 \qquad (2.6.2)$$

$$l = \left\lceil \frac{z_{i,j} - z_{\min}}{N_v} \right\rceil + 1 \qquad (2.6.3)$$

$$k = \left\lceil \frac{t_{i,j} - t_{\min}}{\Delta t} \right\rceil + 1 \qquad (2.6.4)$$

In the above formula, $x_{\min}$, $y_{\min}$, $z_{\min}$, $t_{\min}$ is respectively the minimum value of the studied airspace in each space and time dimension.

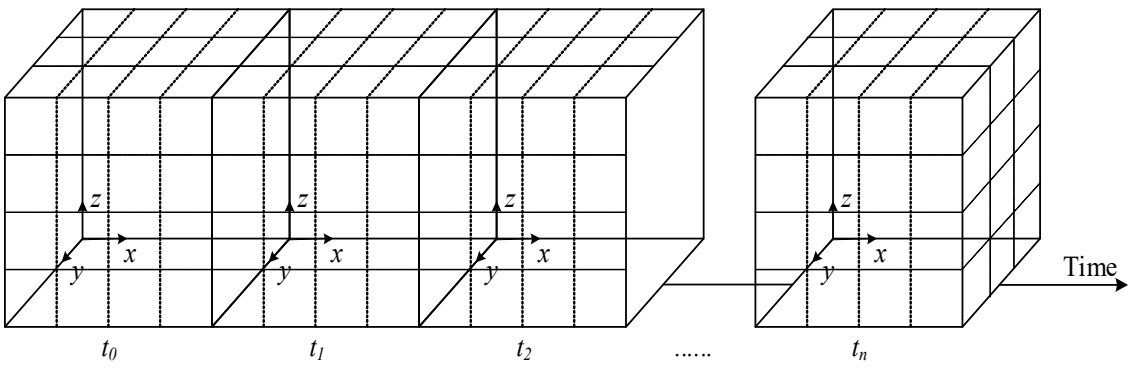


Figure 28: Four-dimension space and time grid for the conflict detection

As shown in Figure 29, the neighborhood of the grid cell $C_{m.n.l.k}$ in the spatial three-dimension consists of 26 grid cells around it, marked as $\mathbf{A}_{m,n,l,k}=\left[\mathbf{a}_{m,n,l-1,k} \quad \mathbf{a}_{m,n,l,k} \quad \mathbf{a}_{m,n,l+1,k}\right]$, where:

$$\mathbf{a}_{m,n,l-1,k}=\begin{bmatrix} C_{m-1,n-1,l-1,k} & C_{m,n-1,l-1,k} & C_{m+1,n-1,l-1,k} \\ C_{m-1,n,l-1,k} & C_{m,n,l-1,k} & C_{m+1,n,l-1,k} \\ C_{m-1,n+1,l-1,k} & C_{m,n+1,l-1,k} & C_{m+1,n+1,l-1,k} \end{bmatrix}_{3\times 3} \tag{2.6.5}$$

$$\mathbf{a}_{m,n,l,k}=\begin{bmatrix} C_{m-1,n-1,l,k} & C_{m,n-1,l,k} & C_{m+1,n-1,l,k} \\ C_{m-1,n,l,k} & C_{m,n,l,k} & C_{m+1,n,l,k} \\ C_{m-1,n+1,l,k} & C_{m,n+1,l,k} & C_{m+1,n+1,l,k} \end{bmatrix}_{3\times 3} \tag{2.6.6}$$

$$\mathbf{a}_{m,n,l+1,k}=\begin{bmatrix} C_{m-1,n-1,l+1,k} & C_{m,n-1,l+1,k} & C_{m+1,n-1,l+1,k} \\ C_{m-1,n,l+1,k} & C_{m,n,l+1,k} & C_{m+1,n,l+1,k} \\ C_{m-1,n+1,l+1,k} & C_{m,n+1,l+1,k} & C_{m+1,n+1,l+1,k} \end{bmatrix}_{3\times 3} \tag{2.6.7}$$

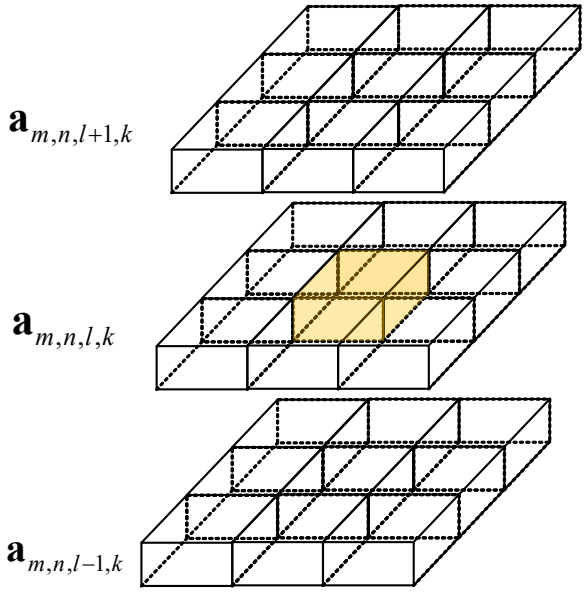


Figure 29: 3D schematic diagram of the neighborhood of the grid cell $C_{m.n.l.k}$

To make an initial detection of the potential conflicts of trajectories, the algorithm examines a total of 27 cells in turn of the cell $C_{m.n.l.k}$ itself and its 26 neighborhood cells. If there are sampling points of other trajectories in any of these cells, it indicates that there may be potential flight conflicts that need more accurate distance calculation between these sample points. Otherwise, the discrete sampling point $P_{i,j}$ of the trajectory i has no potential conflicts with others and there is no need to calculate the distance.

The grid-based conflict detection method can effectively improve the efficiency of the conflict detection process and reduce computational complexity by making initial filtering of potential conflicts. The problem of trajectory planning in the strategic stage involves a large number of aircraft flying across a large area of airspace which needs a large amount of computer memory to process and store the data when programmed to realize

the grid-based conflict detection method. Therefore, it is necessary to use an appropriate data structure to store the corresponding trajectory and airspace information data, to avoid the lack of the ability of large-scale collision detection due to the limitation of computer memory.

Time-Space Data Structure (TSDS) represents each four-dimensional time-space grid cell through a separate memory position. For a given grid cell, the relevant information stored in TSDS can be accessed by simple mathematical formulas according to its index number, including space information (such as discrete trajectory points) and non-space information (such as flight codes).

In the process of conflict detection, TSDS can be considered as a network composed of grid cells in the space-time dimensions, and the granularity of TSDS represents the distance between the grid cells. Too small granularity will affect computing performance and require a lot of computer memory, while sparse granularity will not have the effect of potential conflict detection. Therefore, it is necessary to determine the appropriate TSDS granularity, and its influencing factors include the size of the airspace, the flights to be de-conflicted, the speed of the flights, the available memory of the computer, the space complexity of the designed algorithm, and so on.

The example in Figure 30 shows what TSDS used for conflict detection contains. TSDS can be thought of as a table, in which each row represents a four-dimensional grid cell in the airspace of concern, the number of rows is the same as the total number of grid cells in the space, and each column represents the aircraft trajectory sample points in the grid cells, and the number of columns is the same as the total number of aircraft trajectory sample points. Because the rows are arranged in order, so the position in the rows can be accessed by the following expression according to the coordinates of the trajectory sample points:

$$SDSpos = x \cdot Y \cdot Z \cdot T + y \cdot Z \cdot T + z \cdot T + t \qquad (2.6.8)$$

Where $SDSpos$ is the position stored in the database of the four-dimensional grid cell (x, y, z, t), with $x \in [0, X-1]$, $y \in [0, Y-1]$, $z \in [0, Z-1]$, $t \in [0, T-1]$. X, Y, Z, T is the maximum number of discrete coordinates of each space-time dimension, respectively, which is obtained by dividing the length of each dimension by the size of the grid cell (that is, the granularity of TSDS).

	Pointer 1				Pointer 2				⋯	Pointer n			
Coordinate	Flight	x	y	z	Flight	x	y	z	…	Flight	x	y	z
(0,0,0,0)													
(0,0,0,1)													
(0,0,0,2)	15	453543	546345	3000									
⋮	⋮	⋮	⋮	⋮	⋮	⋮		⋮	⋯	⋮	⋮	⋮	⋮
(0,0,1,0)	3	453578	476343	3300	18	453543	476345	3300	⋯				
(0,0,1,1)									⋯				
(0,0,1,2)									⋯				
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋯	⋮	⋮	⋮	⋮
(0,0,2,0)													
(0,0,2,1)													
(0,0,2,2)	1	414541	473341	3600	2	423543	546391	36	⋯	n	451243	514389	3600
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋯	⋮	⋮	⋮	⋮

Figure 30: Time-Space data structure (TSDS) of the grid data

Since each row in TSDS represents a grid cell, a pointer array is used in the grid cell data to store pointers to the three-dimensional spatial location data of each trajectory sample point belonging to the grid cell.

In using the data of TSDS for conflict detection, when the algorithm will deal with some trajectory, it first calculates the position of the grid cell and the neighborhood grid cell in TSDS for each sampling points the trajectory according to the formula (2.6.8), and then accesses the record content in TSDS by the record position. If the pointer array in the record is empty, it means that no other trajectory sample points are in the grid cell, so the occupation of this grid will not cause a conflict. If the pointer array is not null, the pointer array will be circularly accessed, the three-dimensional spatial position information of other sampling points occupying the grid cell stored by each pointer is accessed sequentially, and the horizontal and vertical distances between these sample points are calculated to determine whether the safety separation standard is violated. The conflict of the trajectory can be obtained by traversing all the sampling points in the trajectory.

2.6.1.2 DETERMINISTIC CONFLICT DETECTION METHOD BASED ON DIFFERENT TYPES OF CONFLICT

The methods of deterministic conflict detection based on the horizontal link conflict, vertex crossing conflict, and vertical conflict are as follows:

(1) Horizontal vertex crossing conflict

As shown in Figure 31, aircraft i and aircraft j at the same flight altitude pass through the same vertex n at flying speed of v_i and v_j respectively, and their vertex passing time is t_n^i and t_n^j respectively. If the time separation between the passing time of the two flights doesn't meet the standard of required time separation, that is $\left|t_n^i - t_n^j\right| < \overline{S}_{i,j}$, it is considered that there is a conflict between the two flights at the vertex. The time separation standard between flight pairs passing through the same vertex is as follows:

$$\overline{S}_{i,j} = \frac{D}{v_i v_j \left|\sin(\theta_{i,j})\right|} \sqrt{v_i^2 + v_j^2 - 2 v_i v_j \cos(\theta_{i,j})} \tag{2.6.9}$$

In the formula(2.6.9), $\theta_{i,j} \in (0,\pi) \cup (\pi, 2\pi)$ is the crossing angle of the path of the two flights at the vertex, and D is the required horizontal safe separation standard.

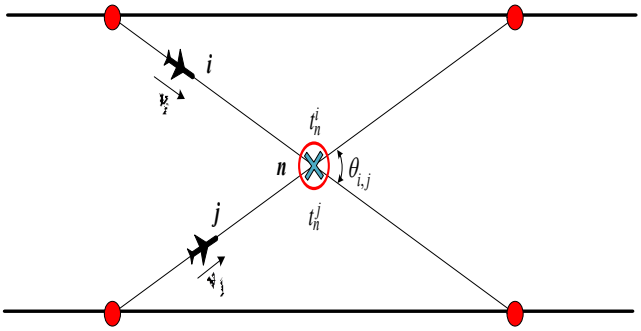


Figure 31: Vertex conflict between two aircraft flying at the crossing route

For situations where the flight speed of an aircraft can be changed, the allowable speed ranges of aircraft i and aircraft j are $v_i \in \left[v_i^-, v_i^+\right]$ and $v_j \in \left[v_j^-, v_j^+\right]$ respectively, where

v_i^- and v_i^+ represent the minimum and maximum speed allowed for aircraft i separately. At the intersection vertex, the time interval between the passing time of the two aircraft is:

$$\overline{S}_{i,j} = \max\left(\frac{D\Omega_{i,j}}{v_i^- \left|\sin(\theta_{i,j})\right|}, \frac{D\Omega_{j,i}}{v_j^- \left|\sin(\theta_{i,j})\right|}\right) \tag{2.6.10}$$

The definition of $\Omega_{i,j}$ is as follows:

$$\Omega_{i,j} = \max\left(\sqrt{\left(\frac{v_i^+}{v_j^-}\right)^2 - \frac{2v_i^+\cos(\theta_{i,j})}{v_j^-} + 1}, \sqrt{\left(\frac{v_i^-}{v_j^+}\right)^2 - \frac{2v_i^-\cos(\theta_{i,j})}{v_j^+} + 1}\right) \tag{2.6.11}$$

(2) Horizontal link conflict

The conflict on the edge is an overtaking conflict. For aircraft flying at the same altitude along the same route segment, an overtaking conflict will occur when the speed of the leading aircraft is less than that of the following aircraft. As shown in Figure 32, aircraft i and aircraft j fly along the same edge, aircraft i enters the edge l earlier than the flight j, that is $t_l^i < t_l^j$, but the aircraft i leaves the edge l later than the aircraft j, that is $\tilde{t}_l^i > \tilde{t}_l^j$. Therefore, it is thought that there is a link conflict between flight i and j when on edge l.

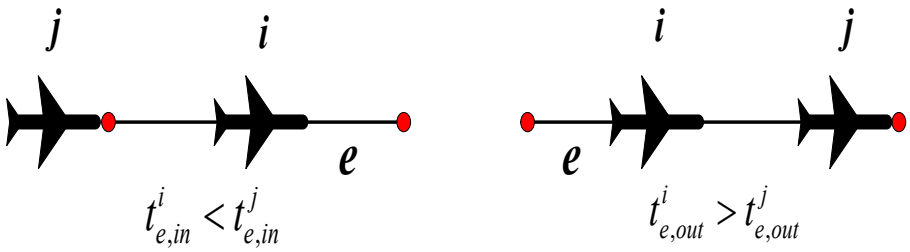


Figure 32: Link conflict between two flights flying on the same edge

Decide whether there is an overtaking conflict between two flights according to the following two conditions:

1) The order of the two flights at the entry point and the exit point of the edge is the same, but the separation required between the flights is not satisfied either at the entry point or the exit point;

2) The order of the two flights at the entry point and the exit point of the edge is different indicating there is an overtaking.

(3) Vertical altitude conflict

The vertical altitude conflict is determined according to where the height separation at the vertex or on the same link is less than the required altitude separation standard, expressed as the following formula:

$$\left|h(t_l^i)-h(t_l^j)\right|\le H_0,\forall l\in L \tag{2.6.12}$$

$$\left|h(t_n^i)-h(t_n^j)\right|\le H_0,\forall n\in V \tag{2.6.13}$$

2.6.2. ROBUST TRAJECTORY CONFLICT DETECTION METHOD CONSIDERING THE UNCERTAINTY OF THE PREDICTED POINT PASSING TIME

2.6.2.1 ROBUST CONFLICT DETECTION METHOD BASED ON GRIDS

Due to the uncertainty of the point passing time of the flight, a spatial-temporal data structure considering the range of the point passing time (Uncertainty Included Time-Space Data Structure ,UI-TSDA) is used for conflict detection. An example of the data structure is shown in Figure 33. A pointer array is used for each grid cell to store pointers to the three-dimensional spatial position data of each trajectory point belonging to the grid cell and the maximum positive and negative time deviation data from the point passing time predicted under the wind ensemble member 0. In each grid element, the maximum positive and negative point passing time deviation value is calculated according to the formula (2.3.9) and formula (2.3.10) in 2.3.2.

	Pointer 1						Pointer 2						⋯
Coordinate	**Flight**	**x**	**y**	**z**	**Maximum Negative deviation**	**Maximum Positive deviation**	**Flight**	**x**	**y**	**z**	**Maximum Negative deviation**	**Maximum Positive deviation**	⋯
(0,0,0,0)													
(0,0,0,1)													
(0,0,0,2)	15	453543	546345	3000	-10	30							
⋮	⋮	⋮	⋮			⋮	⋮	⋮				⋮	⋯
(0,0,1,0)	3	453578	476343	3300			18	453543	476345	3300	-20	50	⋯
(0,0,1,1)													⋯
(0,0,1,2)													⋯
⋮	⋮	⋮	⋮			⋮	⋮	⋮	⋮			⋮	⋯
(0,0,2,0)													
(0,0,2,1)													
(0,0,2,2)	1	414541	473341	3600	-10	30	2	423543	546391	3600	-20	50	⋯
⋮	⋮	⋮	⋮			⋮	⋮	⋮	⋮			⋮	⋯

Figure 33: Time-Space Date Structure considering the point passing time uncertainty

The conflict detection method considering the uncertainty of the predicted uncertainty of the point passing time is based on the deterministic conflict detection method. Since the trajectory point passing time of the flight is no longer a certain time, but a time range, a pointer to the same trajectory point will be stored in multiple space-time grids of different time dimensions within that time range. Then, according to the deterministic space conflict detection method, it is determined whether the corresponding safety separation standard is violated. If there is a violation of the safety separation standard, the probability of the conflict is calculated. The schematic diagram of the calculation method of conflict probability is shown in Figure 34. There is a conflict between flight f's trajectory point p and flight g's trajectory point q. The time ranges of the two flights passing through the conflict trajectory points are respectively $[t_{\min}^p, t_{\max}^p]$ and $[t_{\min}^q, t_{\max}^q]$. The conflict probability is calculated according to the time range passing through the conflicted trajectory points of the two flights, and the formula is shown in the formula (2.6.14) where $t_n^{p,\max}=t_n^{p,0}+\delta t_n^{p,\max}$ and $t_n^{q,\min}=t_n^{q,0}-\delta t_n^{q,\min}$.

$$c(p,q)=\frac{t_n^{p,\max}-t_n^{q,\min}}{t_n^{q,\max}-t_n^{p,\min}} \tag{2.6.14}$$

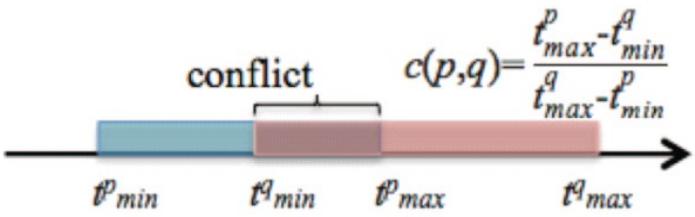


Figure 34: Calculation method of conflict probability between trajectory points considering the points passing time uncertainty

2.6.2.2 ROBUST CONFLICT DETECTION METHOD BASED ON DIFFERENT TYPES OF CONFLICT

Thy symbols involved in the robust trajectory conflict detection method are described as follows:

Estimated arrival time at the node: t_l^f

Earliest possible arrival time at the node: $t_l^{f^+}$

Latest possible arrival time at the node: $t_l^{f^-}$

Set of flight pairs with potential edge overtaking conflict: C_l^0

Set of flight pairs with the potential vertex conflict of the first type : C_v^1

Set of flight pairs with the potential vertex conflict of the second type: C_v^2

Set of flight pairs with the potential vertex conflict of the third type: C_v^3

Set of flight pairs with the potential vertical conflicts on edge l: $C_{h,l}^4$

Set of flight pairs with the potential vertical conflicts on node v: $C_{h,v}^5$

The ratio of ground speed between two aircraft: $\alpha^{(f_1,f_2)}$

Intersection angle of the crossing trajectory: $\theta_v^{(f_1,f_2)}$

Longitudinal distance separation standard between aircraft: s_0

The distance separation required from the trajectory intersection point for the other flight when one flight is passing the point: $S(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})$

Distance required from the trajectory intersection point for two flights with potential vertex conflict: $S'(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})$

(1) Link conflict judgment

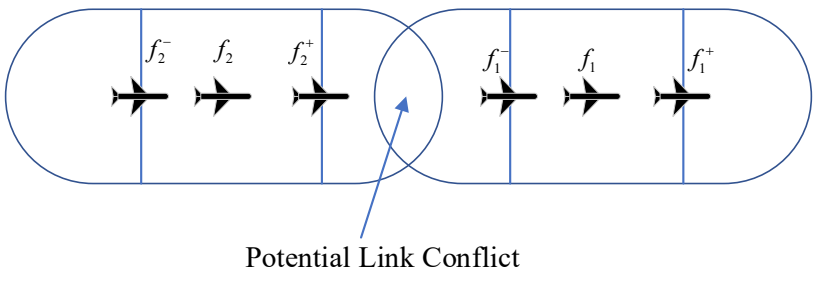


Figure 35: Potential Link conflict

$$\phi_L(l) = \sum_{(f_1,f_2)\in C_l^0} \Pi_{constraint1} - \sum_{(f_1,f_2)\in C_l^0} \Pi_{constraint2} \tag{2.6.15}$$

Potential link conflict is shown in Figure 35. In formula (2.6.15), $\Pi_{constraint1}$ and $\Pi_{constraint2}$ are both indicator functions with the value of 0 or 1. When the constraint condition is satisfied, the value of the indicator function is 1, and when the constraint condition is not satisfied, the value of the indicator function is 0. The corresponding constraint of $\Pi_{constraint1}$ is $t_l^{f_2^+} - (t_l^{f_1^-} + \frac{s_0}{v_g^{f_1}(l)}) < 0$, indicating that there is an intersection between the earliest arrival time of flight f_2 and the corresponding time obtained by the latest arrival time of flight f_1 plus the required time separation $\frac{s_0}{v_g^{f_2}(l)}$. The corresponding constraint of $\Pi_{constraint2}$ is $(t_l^{f_2^+} - \frac{s_0}{v_g^{f_2}(l)}) - t_l^{f_1^-} < 0$, indicating that there is an intersection between the time obtained by the earliest arrival time of the flight f_2 subtracting the required time separation $\frac{s_0}{v_g^{f_2}(l)}$ and the latest arrival time of the flight f_1.

(2) Vertex crossing conflict judgment

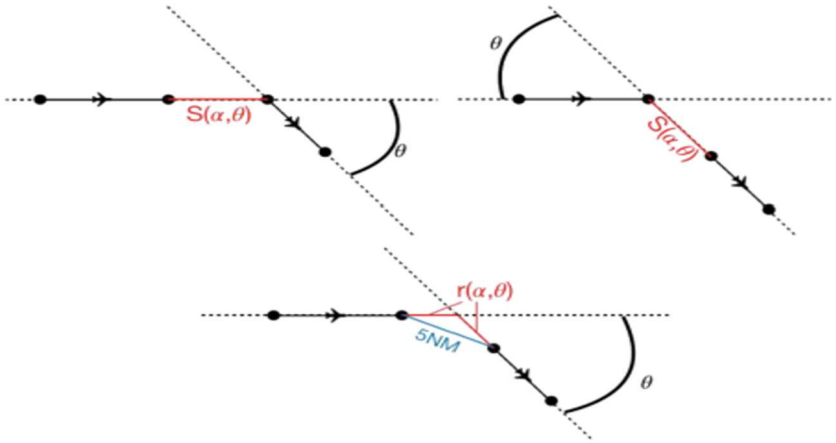


Figure 36: Crossing point conflict

$$S(\alpha^{(f_1,f_2)}, \theta_v^{(f_1,f_2)}) = s_0 * \frac{\sqrt{\alpha^{(f_1,f_2)} \cdot \alpha^{(f_1,f_2)} - 2 \cdot \alpha^{(f_1,f_2)} \cdot \cos(\theta_v^{(f_1,f_2)}) + 1}}{\left|\sin(\theta_v^{(f_1,f_2)})\right|} \tag{2.6.16}$$

$$S'(\alpha^{(f_1,f_2)}, \theta_v^{(f_1,f_2)}) = s_0 * \frac{\sqrt{\alpha^{(f_1,f_2)} \cdot \alpha^{(f_1,f_2)} - 2 \cdot \alpha^{(f_1,f_2)} \cdot \cos(\theta_v^{(f_1,f_2)}) + 1}}{2 \cdot \cos(\frac{\theta_v^{(f_1,f_2)}}{2})} \tag{2.6.17}$$

$$\phi_L(l) = \sum_{(f_1,f_2)\in C_v^1} \Pi_{constraint3} + \sum_{(f_1,f_2)\in C_v^2} \Pi_{constraint4} + \sum_{(f_1,f_2)\in C_v^3} \Pi_{constraint5} \tag{2.6.18}$$

Potential crossing conflict is shown in Figure 36. In formula (2.6.18), $\Pi_{constraint3}$, $\Pi_{constraint4}$, and $\Pi_{constraint5}$ are both indicator functions with values of 0 or 1. The corresponding constraint of $\Pi_{constraint3}$ is $(t_n^{f_2^+} - \frac{S(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_2}(v)}) - t_n^{f_1^-} < 0$, indicating that there is an intersection between the time obtained by the earliest crossing point arrival time of the flight f_2 subtracting the required separation $\frac{S(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_2}(v)}$ and the latest arrival time of the flight f_1. The corresponding constraint of $\Pi_{constraint4}$ is $t_n^{f_2^+} - (t_n^{f_1^-} + \frac{S(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_1}(v)}) < 0$, indicating that there is an intersection between the earliest arrival time of the flight f_2 and the time obtained by the latest arrival time of the flight f_1 plus the required time separation $\frac{S(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_1}(v)}$. The corresponding constraint of $\Pi_{constraint5}$ is $(t_n^{f_2^+} - \frac{S'(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_2}(v)}) - (t_n^{f_1^-} + \frac{S'(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_1}(v)}) < 0$, indicating that there is an intersection between the time obtained by the earliest crossing point arrival time of flight f_2 subtracting the required time separation $\frac{S'(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_2}(v)}$ and the time obtained by the latest crossing point arrival time of flight f_1 plus the required time separation $\frac{S'(\alpha^{(f_1,f_2)},\theta_v^{(f_1,f_2)})}{v_g^{f_1}(v)}$.

(3) Vertical altitude conflict

$$H_{constraint1} = \begin{cases} 1, \left|h(t_n^{f_1^-}) - h(t_n^{f_2^+})\right| \le H_0 \\ 0, \left|h(t_n^{f_1^-}) - h(t_n^{f_2^+})\right| > H_0 \end{cases}, \forall l \in L, (f_1, f_2) \in C_{h,l}^4 \quad (2.6.19)$$

$$H_{constraint2} = \begin{cases} 1, \left|h(t_n^{f_1^-}) - h(t_n^{f_2^+})\right| \le H_0 \\ 0, \left|h(t_n^{f_1^-}) - h(t_n^{f_2^+})\right| > H_0 \end{cases}, \forall n \in V, (f_1, f_2) \in C_{h,v}^5 \quad (2.6.20)$$

In formula (2.6.19) and (2.6.20), $H_{constraint1}$, $H_{constraint2}$ are both indicator functions with values of 0 or 1. The corresponding constraint of $H_{constraint1}$ is $\left|h(t_l^{f_1^-}) - h(t_l^{f_2^+})\right| \le H_0, \forall l \in L, (f_1, f_2) \in C_{h,l}^4$, which indicates there is an altitude conflict at the edge. The corresponding constraint of $H_{constraint2}$ is

$\left|h(t_n^{f_1^-}) - h(t_n^{f_2^+})\right| \le H_0, \forall n \in V, (f_1, f_2) \in C_{h,v}^5$, which indicates there is an altitude conflict on the vertex.

2.7. ROBUST TRAJECTORY DECONFLICTION MODEL AND ALGORITHM

2.7.1. ROBUST TRAJECTORY DECONFLICTION MODEL

The model considers the following assumptions:

(1) Only consider the conflicts with the long-haul flights during the cruise phase;

(2) Aircraft can resolve conflicts by adjusting the departure time and flight level;

(3) The starting point and the endpoint of the aircraft trajectory are the top of climb and top of descent, respectively;

(4) The aircraft flies along the established route at a fixed cruising speed and flies in a straight line between the route points.

Introduce the following decision variables:

- Adjustment of take-off time: ΔG^{f_i}, where $i = 1,...,n_f$, $n_f = |F|$
- The flying altitude after deconfliction:

$$x_{l,h}^f(t) = \begin{cases} 1 & \text{, If flight f reaches edge l at flight level h and at time t} \\ 0 & \text{, Otherwise} \end{cases}$$

To resolve the conflicts, the objective function is to minimize the total number of conflicts:

$$\min \Phi = \sum_{l \in L} \phi_L(l)(1 - H_{constra\,int\,1}) + \sum_{v \in V} \phi_v(v)(1 - H_{constra\,int\,2}) \tag{2.7.1}$$

In formula(2.7.1), the definition of $\phi_L(l)$, $\phi_v(v)$, $H_{constra\,int\,1}$ and $H_{constra\,int\,2}$ can be seen in formula (2.6.15), (2.6.18), (2.6.19), (2.6.20).

In addition to reducing conflicts, we also need to consider reducing the deviation from the planned trajectory or preferred trajectory. Taking into account the number of remaining conflicts and flight-weighted deviation of adjusted deconflicted trajectory, the objective function expression is as follows:

$$F_{\max} = \frac{n - \sum_{i=1}^{n} (c_i^{GD} \frac{\delta_i}{\delta_{\max}} + c_i^{FL} \frac{h_i}{h_{\max}}) * c_i}{1 + \sum_{l \in L} \phi_L(l)(1 - H_{constra\,int\,1}) + \sum_{v \in V} \phi_v(v)(1 - H_{constra\,int\,2})} \tag{2.7.2}$$

In formula (2.7.2), c_i^{GD} is the weight coefficient of flight f_i with take-off time slot adjustment, and c_i^{FL} is the weight coefficient of f_i with flight level adjustment. c_i is the weight coefficient of the flight f_i. The higher the priority of the flight, the greater the value of c_i . $\delta_{\max}$ and $h_{\max}$ are parameters for normalization of take-off slot adjustment and flight altitude adjustment, which are the maximum allowable take-off slot adjustment and altitude adjustment, respectively. $c_i^{GD}+c_i^{FL}=1$ is satisfied between c_i^{GD} and c_i^{FL} . Setting different values of c_i^{GD} and c_i^{FL} can reflect different deconfliction preferences when using take-off slot adjustment and flight level adjustment. The larger the value of the weight parameter is, the less inclined it is to adopt the trajectory adjustment method that the weight is related to. The coefficients can also be used to distinguish the priority of different flights when making the trajectory adjustment. The larger the value of the coefficient for the flight, the less inclined the flight trajectory will be adjusted for deconfliction.

The constraints are as follows:

$$x_{l,h}^f(t) \le \sum_{l' \in O_l^f, h' \in L^f} x_{l',h'}^f(t+\alpha_l^f), \forall f \in F, l \in E^f, h \in L^f, t \in T_l^f \tag{2.7.3}$$

$$\sum_{h \in L^f} x_{l,h}^f(\overline{T}_l^f) \le \sum_{l' \in I_l^f, h' \in L^f} x_{l',h'}^f(\overline{T}_{l'}^f), \forall f \in F, l \in E^f \tag{2.7.4}$$

$$\sum_{h \in L^f} x_{l,h}^f(\overline{T}_l^f) \le 1, \forall f \in F, l \in E^f \tag{2.7.5}$$

$$x_{l,h}^f(t-1) - x_{l,h}^f(t) \le 0, \forall f \in F, l \in E^f, h \in L^f, t \in T_l^f \tag{2.7.6}$$

$$(t - t_{sche}^f)(x_{orig_f,h}^f(t) - x_{orig_f,h}^f(t-1)) \le \delta^f, \forall f \in F, h \in L^f, t \in T_l^f \tag{2.7.7}$$

$$\left|h^f - h_{sche}^f\right|(x_{orig_f,h}^f(t) - x_{orig_f,h}^f(t-1)) \le \rho^f, \forall f \in F, h \in L^f, t \in T_l^f \tag{2.7.8}$$

Formula (2.7.3) means that a flight must reach the subsequent edge within the time α_l^f after it reached the edge l, and O_l^f represents the set of edges after leaving the edge l and α_l^f is the flight time duration for the flight f to fly over the edge l. Formula (2.7.4) means that the edge l must be entered through the pre-sequence edge segment of the edge l, and I_l^f is the set of edges before entering the edge l and $T_l^f = \left[\underline{T}_l^f, \overline{T}_l^f\right]$ is the collection of feasible periods for flight f to pass through the edge l. Formula (2.7.5) ensures that the aircraft maintains a constant flight level when flying on a specific edge. Formula (2.7.6) ensures the time continuity of the trajectory, that is, the decision variable $x_{l,h}^f(t)$ must be 1 for all the subsequent periods $t \ge t^*$ when a flight reached the edge l at the time t^*. Formula (2.7.3)-(2.7.6) ensure the uniqueness of the trajectory of each flight. Formula (2.7.7) indicates that the departure delay of a flight can't exceed the acceptable delay limit δ^f. Formula (2.7.8) restricts the flight level change range, that is, the gap between the adjusted flight level and the initially planned flight level does not exceed ρ^f.

2.7.2. ROBUST TRAJECTORY DECONFLICTION ALGORITHM

Hybrid Meta-heuristic Optimization Algorithm (HMOA) can be used to solve the 4D trajectory deconfliction problem in and acceptable time range. HMOA combines a simulated annealing algorithm with a local search algorithm. The local search algorithm is used to intensify the search of the solution space around a potentially feasible solution and executes with probability, while SA can avoid the algorithm being trapped into the local optimum by accepting degraded solutions with probability. The local search algorithm is considered as the inner loop of the simulated annealing algorithm and will execute when predefined conditions are met. Figure 37 shows the process of HMOA.

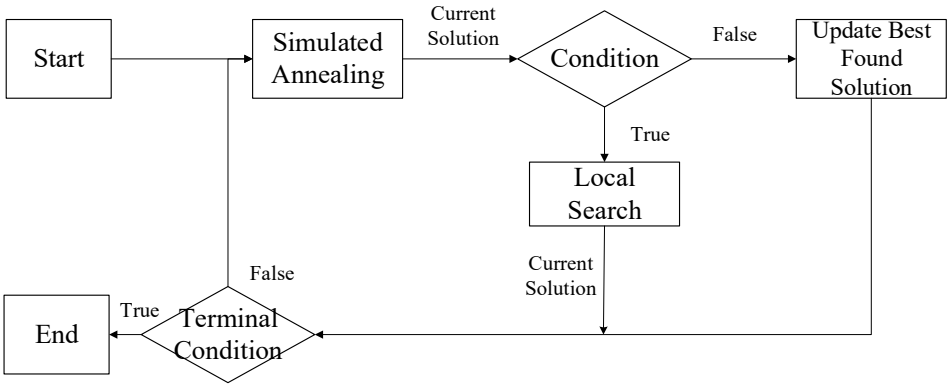


Figure 37: HMOA framework

2.7.2.1 SIMULATED ANNEALING ALGORITHM

The simulated annealing algorithm is inspired by the annealing process in metallurgy. Making use of the energy change characteristic of thermal motion, the algorithm extended on the base of the local search algorithm. The most important feature of the algorithm is the use of the Metropolis algorithm, that is, it accepts the degraded solution as a new solution with a certain probability. For the generated neighborhood solution , if it meets the established acceptance criterion, the solution is accepted as a new solution; otherwise, the degraded solution is accepted according to the following probability：

$$p = e^{-\frac{\Delta E}{T}} \tag{2.7.9}$$

In formula (2.7.9), ΔE refers to the degradation of the objective function which is the difference in the objective function between the new solution and the current solution. If the maximum number of iterations N_I at a given temperature is reached, the temperature is decreased with the formula $T_{i+1} = \alpha \cdot T_i$, and this process is repeated until the pre-defined final temperature T_f is reached.

Other important parameters of the simulated annealing algorithm include initial temperature, cooling formula, length of the Markov chains, and termination condition.

2.7.2.2 NEIGHBORHOOD STRUCTURE AND ACCEPTANCE CRITERIA

The definition of neighborhood structure is an important part of the hybrid meta-heuristic algorithm, which determines the next step search direction.To generate a new neighborhood solution, a local change is carried out on a chosen trajectory i in the current solution. To obtain more effective solutions that satisfy the conflict constraints, the roulette selection method is used to select the trajectory that will be locally changed.

When the objective function only considers minimizing conflicts, according to the results of conflict detection, the number of conflicts Φ_i with the other flights for each flight is calculated, and the probability p_i of the flight i being selected is calculated according to the number of conflicts Φ_i of flight i :

$$p_i = \frac{\Phi_i}{\sum_{i=1}^{N} \Phi_i} \tag{2.7.10}$$

When the objective function takes into account both the minimization of conflicts and the amount of trajectory deviation from the planned trajectory considering flight priority, the probability p_i of flight i being selected is calculated according to the number of conflicts Φ_i and the coefficient c_i of flight i :

$$p_i = \frac{\frac{\Phi_i}{c_i}}{\sum_{i=1}^{N} \frac{\Phi_i}{c_i}} \tag{2.7.11}$$

The cumulative probability P_i is obtained by accumulating the sum:

$$P_i = \sum_{k=1}^{i} p_k \tag{2.7.12}$$

A value of r in [0,1] is generated randomly, and the trajectory i is selected according to the cumulative probability interval in which this random value lies, that is, the random value satisfies the constraint $P_{i-1} \le r < P_i$. If there is no conflict, the trajectory for local change is randomly selected. For the selected trajectory i, two maneuver options could be used for conflict resolution, including ground delays and flight level allocations. To allow airspace users to express their preference on the maneuver options of conflict resolution, a user-defined parameter, P_{GH}, between [0,1] is introduced, which controls the probability of performing ground delay, and the probability of performing flight level allocation on the trajectory i then is ($1-P_{GH}$). The process of generating the neighborhood solution is presented in Figure 38.

After evaluating the generated neighborhood solution, the algorithm needs to determine whether it improves objectives compared with the current solution and whether to accept it as the new current solution. Hence, it is necessary to specify the acceptance criterion. The objective function value of the solution is used as the acceptance criterion, and the algorithm accepts the solution that reduces the objective value.

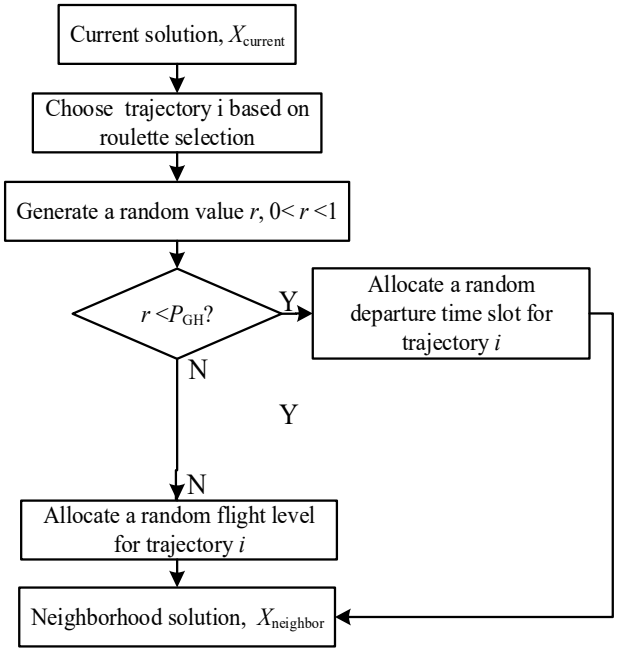


Figure 38: Flow chart of Neighborhood structure

2.7.2.3 LOCAL SEARCH ALGORITHM

In the hybrid meta-heuristic algorithm, the hill-climbing local search algorithm is used to search the adjacent solution space of the current solution to update the current optimal solution. Firstly, a neighborhood solution is generated by applying the local change to the current solution according to the above-mentioned neighborhood structure. Then, the algorithm only accepts the neighborhood solution, which can improve the value of the objective function, as the new current solution. This process is repeated until reaching the maximum number of iterations n_I. The local search algorithm adopts the following two searching strategies:

(1) Intensification of the search on one Particular Trajectory

Given a flight i, this state-exploitation step focuses on improving the current solution by applying a local change from the neighborhood structure only to flight i.

(2) Intensification of the search on the Interacting Trajectories

Given a flight i, this state-exploitation step applies a local change from the neighborhood structure to every flight that is currently interacting with the flight i. For instance, suppose that trajectory i interacts with trajectory p, q, and r, then trajectory p, q, and r and adjusted by the departure time or flight level respectively to generate new neighborhood solutions.

2.7.2.4 THE HYBRID META-HEURISTIC ALGORITHM

HMOA executes the simulated annealing algorithm (SA) and/or the hill-climbing local search through the predetermined probability. The execution probability is related to the controlled temperature.

The probability P_{SA} of executing the simulated annealing algorithm is calculated as:

$$P_{\mathrm{SA}}(T) = P_{\mathrm{SA,min}} + (P_{\mathrm{SA,max}} - P_{\mathrm{SA,min}}) \cdot \frac{T_0 - T}{T_0} \tag{2.7.13}$$

In formula (2.7.13), $P_{\mathrm{SA,max}}$ and $P_{\mathrm{SA,min}}$ are the maximum and minimum probability of executing the simulated annealing algorithm set by the user, T_0 and T are the initial temperature and the current temperature, respectively.

The probability P_{LOC} of executing the local search algorithm is calculated as:

$$P_{\mathrm{LOC}}(T) = P_{\mathrm{LOC,min}} + (P_{\mathrm{LOC,max}} - P_{\mathrm{LOC,min}}) \cdot \frac{T_0 - T}{T_0} \tag{2.7.14}$$

In formula (2.7.14), $P_{\mathrm{LOC,max}}$ and $P_{\mathrm{LOC,min}}$ are the maximum and minimum probability of executing the local search algorithm.

The probability P_{SL} of performing simulated annealing and local search algorithm simultaneously is calculated as:

$$P_{\mathrm{SL}}(T) = 1 - \left(P_{\mathrm{SA}}(T) + P_{\mathrm{LOC}}(T)\right) \tag{2.7.15}$$

The local search algorithm is used to intensify the search of the solution space around a potentially feasible solution, while SA can avoid the algorithm being trapped into the local optimum by accepting degraded solutions with probability. The key factor of the HMOA is to achieve an effective balance between the exploration and exploitation of the algorithm. That is, it is necessary to make a compromise between the exploration of the whole search space and the exploitation of the local optimal.

The flow chart of the proposed HMOA is illustrated in Figure 39.

Step 1: Set the controlled temperature as the initial temperature, that is, $T = T_0$. Generate a feasible initial solution X_{current} and calculate the objective function value Y_{current} of the initial solution.

Step 2: Calculate the probability P_{SA}, P_{LOC}, and P_{SL} of executing each algorithm according to the controlled temperature.

Step 3: A random value of r in [0,1] is generated, and judge the relationship between r and the probability parameter P_{SA}. When the condition $r > P_{\mathrm{SA}}$ satisfies, the simulated annealing algorithm is executed to update the current solution X_{current}, otherwise, continue with Step 4.

Step 4: Judge the relationship between r and the probability parameter P_{LOC}. When the condition $r > P_{\mathrm{LOC}}$ satisfies, the local search algorithm is executed, otherwise, both the simulated annealing algorithm and the local search algorithm are both executed to update the current solution X_{current}.

Step 5: Repeat Steps 2-4 until the number of cycles reaches the Markov chain length N_I at the specified current temperature. Update the controlled temperature according to the cooling formula.

Step 6: Repeat Steps 2-5 and terminate the algorithm when the controlled temperature reaches or below the termination temperature.

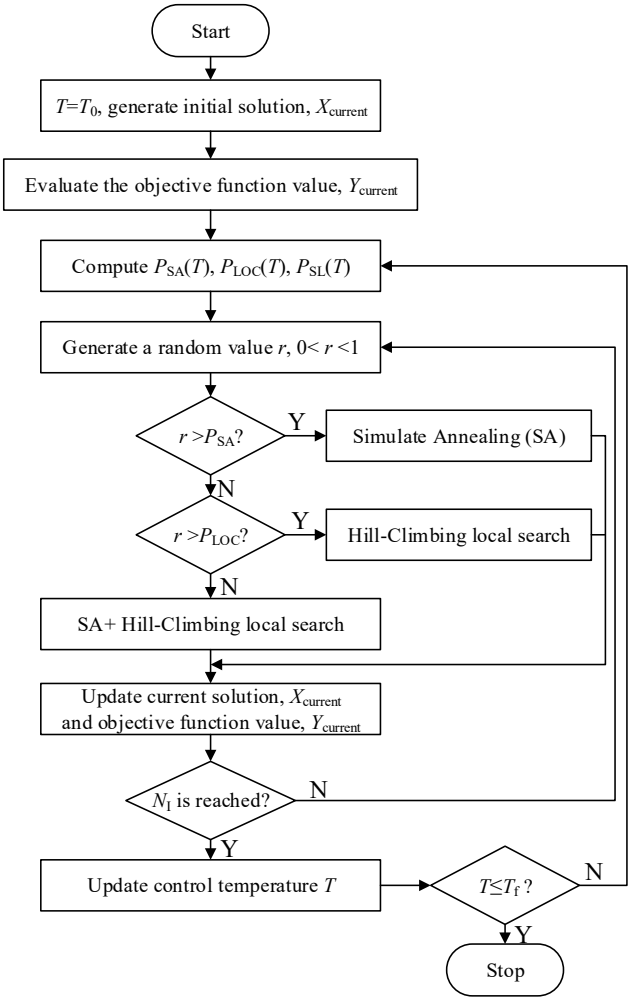


Figure 39: Flow chart of the Hybrid meta-heuristic algorithm

2.7.3. ROBUST TRAJECTORY CONFLICT RESOLUTION VERIFICATION

The conflict resolution algorithm is verified with flights that pass by any sector that is the same as those passed by the flight between the city pairs of Sino-European. According to the optimized horizontal path of flight KLM888 which is executed on 2019.6.8 between VHHH and EHAM with the estimated departure time at 04: 20(UTC). In eastern China, there are 7 sectors passed by the flight, including ZGGGAR05, ZGGGAR03, ZGGGAR23, ZGGGAR11, ZUUUAR23, ZUUUAR22, ZUUUAR18, and in western China, there are 9 sectors passed by the flight, including ZUUUAR08, ZUUUAR17, ZUUUAR12, ZUUUAR03, ZLLLAR04, ZLLLAR12, ZWWWAR01, ZWWWAR04, ZWWWAR02. A total of 636 flights were chosen for potential conflicts with KLM888 in the 16 sectors.

Based on the robust trajectory conflict detection method for different types of conflicts in 2.6.2.2, the conflict resolution calculation experiment is carried out by using the robust conflict resolution algorithm in 2.7.1. The model parameters are set as in Table 19. The HMOA algorithm parameters are set as in Table 20.

Table 19: Parameter setting of the model

Parameter	Value
Length of grid cell	5 NM
Height of grid cell	1000 ft
Discretization time step, Δt	20 s
coefficient of ground delay cost, c_i^{GD}	0.5
coefficient of flight level allocation cost, c_i^{FL}	0.5
Maximum departure time shift, δ^f	60 min
Discretization step of departure time	5 min
Maximum flight level shifts, ρ^f	4

Table 20: Parameter setting of HMOA

Parameter	Value
Maximum iteration number of SA, N_1	100
Maximum iteration number of local search, n_1	5
Initial rate of temperature reduction coefficient, τ_0	0.4
Temperature reduction coefficient, α	0.99
Final temperature, T_f	$T_0/1000$
Probability of flight level shift, P_{FL}	0.9
Departure time adjustment probability, P_{GH}	0.1
Minimum probability of running SA, $P_{\mathrm{SA,min}}$	0.8
Maximum probability of running SA, $P_{\mathrm{SA,max}}$	0.9
Minimum probability of running local search, $P_{\mathrm{LOC,min}}$	0.4
Maximum probability of running local search, $P_{\mathrm{LOC,max}}$	0.6

The model is solved and the results are analyzed in three cases: only conflict minimization is considered in the objective function, both conflict and trajectory adjustment minimization are considered, and both minimizations of conflict and trajectory adjustment weighted by flight priority are considered.

(1) Results for the case when only conflict minimization is considered in the objective

function

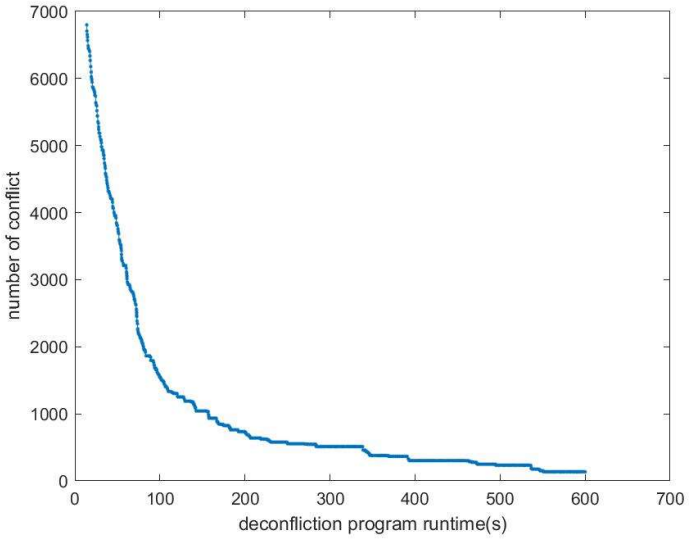


Figure 40: Iterative convergence curve of conflicts number

Table 21: Variation of the number of remaining conflicts with iterative time when the objective function only considers the conflicts number

Iterative time(s)	Remaining conflicts number
20	5945
40	4304
60	3171
120	1300
180	826
240	578
300	512
360	378
420	303
480	248
540	174
600	135

It is shown in Figure 40 and Table 21, the remaining conflicts kept reducing from 6800 to 135 in 10 minutes. About 98% of the conflicts have been resolved.

(2) Results for the case when both conflict and trajectory deviation amount minimization are considered in the objective function

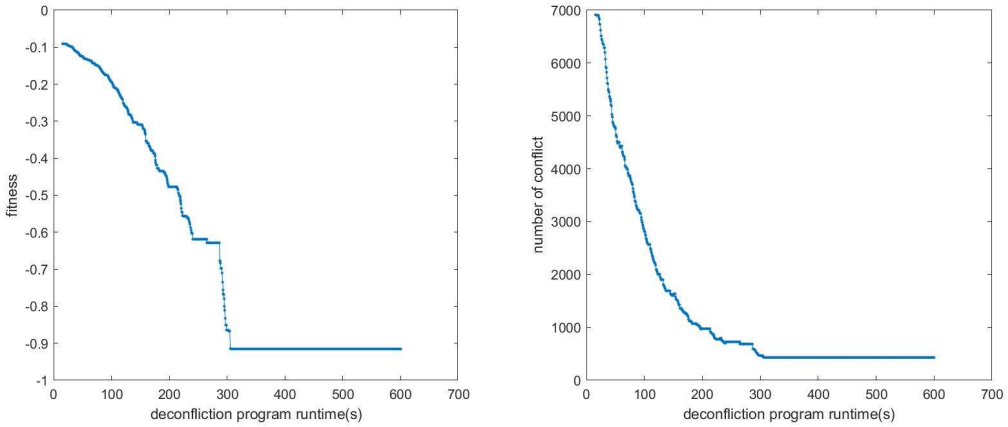


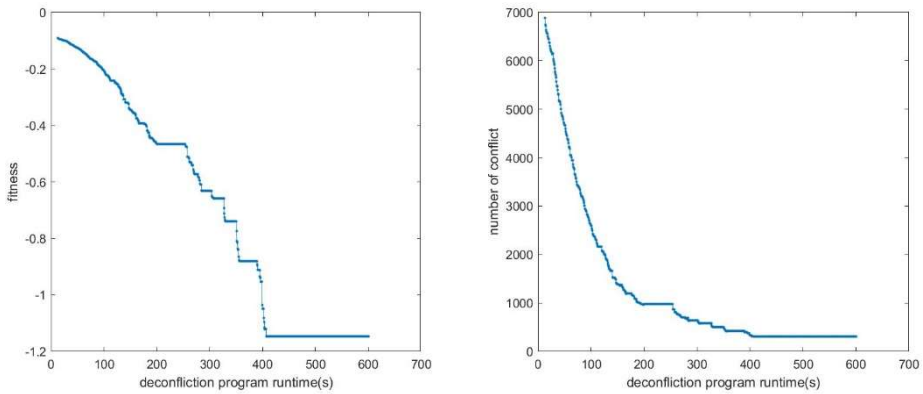
Figure 41: Iterative convergence curve of fitness function and conflicts number

Table 22: Variation of the value of optimization metrics with iterative time when the objective function considers both the conflicts number and the trajectory deviation amount

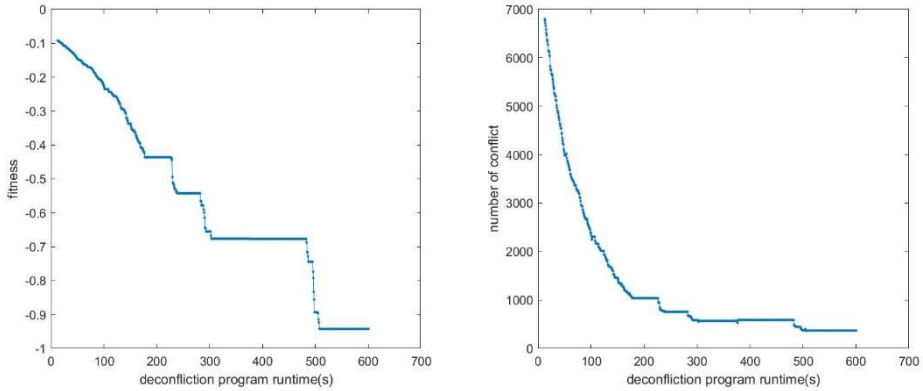
Iterative time(s)	Remaining conflicts number	Total departure time adjustment amount(min)	Total flight level adjustment amount(m)
20	6903	50700	1500
40	5336	1127400	4500
60	4432	1787400	7200
120	2094	4393200	23400
180	1112	7133100	38400
240	727	8655600	55200
300	466	9572100	70800
360	430	9594900	74100
420	430	9594900	74100
480	430	9594900	74100
540	430	9594900	74100
600	430	9594900	74100

It is shown in Figure 41 and Table 22, the remaining conflicts reduced from 6800 to 430 in 6 minutes, decreasing by 94%. After that, with the increase of running time, the number of remaining conflicts and the amount of flight trajectory adjustment no longer change, indicating the algorithm has entered a stage of convergence stagnation.

(3) Results for the case when both conflict and trajectory deviation amounts weighted by flight priority minimization are considered



(a) c_i=20



(b) c_i=100

Figure 42: Iterative convergence curve of fitness function and conflicts number for different flight priority coefficient c_i

Table 23: Average value of optimization metrics for different flight priority coefficient c_i

(Average results of 15 times)

c_i	Remaining conflicts number	Departure time adjustment for KLM888 (min)	Flight level adjustment for KLM888 (m)	Departure time adjustment for other flights except for KLM888 (min)	Flight level adjustment for other flights except for KLM888 (m)
1	427	4.67	1075.18	1283.67	$1.044*10^7$
5	434.33	1.33	1333.67	1347	$1.035*10^7$
20	389.67	4.67	640.84	1504.33	$1.092*10^7$
50	449	0	200.17	1472.67	$1.068*10^7$
100	473.47	0	120	1450.67	$1.058*10^7$
Absolute priority	474.32	0	0	1432.14	$1.052*10^7$

The convergence curve is shown in Figure 42 for the value of c_i equals 20 and 100 separately. The 15 times average results of the optimization metrics under the different values of c_i are shown in Table 23. As can be seen from Table 23, for all the tested values of c_i, the conflict decreasing rate is not lower than 93% in a 10min running time. When value of the flight priority weight coefficient c_i is 1, 5, and 20, the reduction of the number of remaining conflicts will lead to an increase in the total flight trajectory deviation amount under the same algorithm solving time (10 minutes). Then with the further increase of the priority of the flight, the cost for making trajectory deviation of the priority flight is further increased. But at the same time, the priority flight is also a flight with many conflicts with other flights, making trajectory adjustments for the flight is beneficial for reducing the conflict. Under the objective function of comprehensively considering reducing the amount of trajectory adjustment and reducing the conflicts, when the value of c_i increases from 50 to 100, the total conflicts number and the trajectory adjustment amount for other flights will both increase with the increase of c_i. When the value of c_i is 100 , the result is close to the result of having absolute priority

for the Sino-European flight, that is, no trajectory adjustment will be made for the priority flight.

3. CONCLUSIONS

According to the above research work, the main conclusions are as follows:

(1) As the forecast look-ahead time increases, the degree of dispersion of high-altitude wind ensemble forecasting members and the error of wind ensemble forecasts both show an increasing trend. The wind forecast errors calculated based on AMDAR data and 0-Hr analysis data are calculated. It is found that using of historical forecast errors to correct the forecast values of ensemble members of ECMWF EPS could not reduce the forecast errors based on currently available data.

(2) Using the ensemble trajectory prediction result, the maximum positive and maximum negative point passing time deviation relative to the time predicted under 0-wind ensemble forecast member is found influenced by the forecast look-ahead time of the wind and flying time after departure. Significant multiple linear regression equations were built for quantifying the uncertainty of waypoint passing time.

(3) Trajectory optimization adopts the method of optimizing the horizontal route and vertical profile respectively with the output of the optimized horizontal route as the input of the vertical profile optimization. The effect of both flight time and fuel consumption of the optimized trajectory is verified by comparing with the simulation results of the planned trajectory and for all the ensemble members of the wind ensemble forecast data. The average flight time and fuel consumption decreasing percentage for flying by flight level of the verifying flight is 7.4% and 1.8% separately.

(4) The robust trajectory deconfliction model was built and Hybrid Meta-heuristic Optimization Algorithm was designed for solving the model. With the quantified waypoint passing time deviation and the optimized trajectory for the VHHH-EHAM flight as input, 636 flights that have potential conflicts in sectors of Chinese airspace with the trajectory optimized flights are chosen to verify the deconfliction model and algorithm. The conflict decreasing rate is not lower than 93% for different flight priority values of the VHHH-EHAM flight.