



Project no. TREN/07/FP6AE/S07.71574/037180 IFLY

iFly

Safety, Complexity and Responsibility based design and validation of highly automated Air Traffic Management

Specific Targeted Research Projects (STREP)

Thematic Priority 1.3.1.4.g Aeronautics and Space

iFly Deliverable D5.4
Final Report Including validation
Version: Final Draft (1.0)

Authors: E. Siva (UCAM), **J.M. Maciejowski** (UCAM),
G. Chaloulos (ETHZ), **J. Lygeros** (ETHZ), **G. Roussos** (NTUA),
K.Kyriakopoulos (NTUA)

Start date of project: 22 May 2007

Duration: 51 months

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

DOCUMENT CONTROL SHEET

Title of document: Final Report Including validation

Authors of document: E. Siva, J. Maciejowski, G. Chaloulos, J. Lygeros, G. Roussos, K. Kyriakopoulos

Deliverable number: D5.4

Project acronym: iFly

Project title: Safety, Complexity and Responsibility based design and validation of highly automated Air Traffic Management

Project no.: TREN/07/FP6AE/S07.71574/037180 IFLY

Instrument: Specific Targeted Research Projects (STREP)

Thematic Priority: 1.3.1.4.g Aeronautics and Space

DOCUMENT CHANGE LOG

Version #	Issue Date	Sections affected	Relevant information
0.1	24.08.2011	All	First draft
0.2	07.09.2011	1, 4, App. A	Second draft
0.3	03.10.2011	All	Third Draft after Consortium review
0.4	17.10.2011	2, 3, 4	Additional changes after Consortium comments
1.0	26.10.2011	3	Final Draft Minor changes in the presentation of Short-term validation results

Version		Organisation	Signature/Date
Authors	E. Siva	UCAM	
	J.M. Maciejowski	UCAM	
	G. Chaloulos	ETHZ	
	J. Lygeros	ETHZ	
	G. Roussos	NTUA	
	K. Kyriakopoulos	NTUA	
Internal reviewers	H.A.P. Blom	NLR	
	R. Irvine	EEC	
	P. Casek	HNWL	

iFly, Work Package WP5, D5.4 Final Report Including validation

Edited by:
NTUA

October 26, 2011

Abstract

This is the final deliverable of the work package 5 of the project iFly. The aim of WP5.4 is to perform an initial validation of the conflict resolution methods that have been developed in the previous sub work-packages of WP5. In order to assess the performance of the algorithms in demanding multiple aircraft conflict situations, realistic air traffic samples provided by Eurocontrol Experimental Centre have been used.

Contents

1	Introduction	5
1.1	iFly WP5	5
1.2	Objective of this deliverable	6
1.3	Organisation of this deliverable	8
2	Realistic Air Traffic Sample	9
3	Validation of Navigation Functions Short-term algorithm	11
3.1	Short-term CD&R using Navigation Function (NF)	11
3.2	Validation against a realistic traffic sample	14
3.2.1	Medium-scale scenarios	14
3.2.2	Large-scale scenarios	21
3.3	Conclusions	25
4	Validation of Prioritized MPC Mid-term algorithm	28
4.1	Centralized Prioritized Conflict Resolution	28
4.2	Centralized Formulation	29
4.3	Overall hierarchical formulation	31
4.4	Decentralized Hierarchical Formulation	32
4.5	Simulation setup and results	32
4.5.1	Effect of priorities	33
4.5.2	Effect of wind simplifications	35
4.5.3	Separation constraint enforcement	38
4.5.4	Decentralized approach	39
4.5.5	Open-loop vs Feedback approach	39
4.5.6	Summary of simulations	41
4.6	Validation against a realistic traffic sample	42
4.6.1	Multiple Flight Levels	43
4.6.2	Identifying conflicting situations	43
5	Validation of Distributed MPC Mid-term algorithm	48
5.1	Recap of robust distributed MMPC	48
5.2	Results on Simple Scenarios	49
5.3	Application to Realistic Traffic Scenarios	55
5.3.1	Summary	58
6	Concluding remarks	63
A	Prioritized Conflict Resolution	65
A.1	Centralized Formulation	66
A.1.1	Dynamical constraints	67
A.1.2	Velocity and acceleration constraints	67
A.1.3	Cost and priority constraints	68

A.1.4	Separation constraints	68
A.1.5	Constraint relaxation	69
A.1.6	Wind correlation modeling	70
A.1.7	Enforcement of inter-sample constraints	72
A.1.8	Enforcement of constraints only on sample points	74
A.2	Feedback policies	74
A.3	Overall finite horizon optimization problem	76
A.4	Flight Management System (FMS), autopilot and dynamics	77
A.5	Overall hierarchical formulation	77
A.6	Model Parameters	78
B	MMPC Technical Details	80
B.1	Variable Horizon	80
B.2	Modified Robust MMPC Algorithm	80

List of Acronyms

A³	Autonomous Aircraft Advanced
ASAS	Airborne Separation Assurance System
ATC	Air Traffic Control
ATM	Air Traffic Management
BADA	Base of Aircraft Data
CD&R	Conflict Detection and Resolution
ConOps	Concept of Operations
CR	Conflict Resolution
ECAC	European Civil Aviation Conference
FMS	Flight Management System
FP	Flight Phase
MILP	Mixed Integer Linear Program
MPC	Model Predictive Control
NF	Navigation Function
SA	Separation Assurance
SWIM	System Wide Information Management
TAS	True Air Speed

1 Introduction

1.1 iFly WP5

The objective of WP5 is to investigate and push the limits of conflict resolution algorithms for the Autonomous Aircraft Advanced (A³) Concept of Operations (ConOps) by WP1. This covers both the most advanced conflict resolution methods that have been already developed in the literature, as well as novel approaches which have been identified by the HYBRIDGE project as innovative and feasible for application to air traffic management and are being further developed within WP5. The work in WP5 is structured in four sub-WPs:

- WP5.1: Comparative study of conflict resolution methods. Within this sub-WP, a survey of different methods proposed for conflict resolution has already been carried out. Both centralized and decentralized conflict resolution methods have been considered as starting points towards a distributed Conflict Detection and Resolution (CD&R) approach. The emphasis has been on methods that provide proven performance and can be applied in an autonomous fashion. The methods have been analysed and compared in terms of their capabilities, limitations and complementarities from a general autonomous aircraft conflict resolution perspective. The findings of this sub-WP have been documented in [5].
- WP5.2: Analysis of conflict resolution needs of A³ operation developed by WP1 and WP2. Within this sub-WP, the conflict resolution requirements imposed by this concept, as well as the resources that the concept can make available for conflict resolution tasks (in terms of communication, computation, stakeholder roles, etc.) have been identified. Furthermore, conflict resolution methods have been compared versus these requirements and strengths and weaknesses of each method have been identified. The findings of this sub-WP have been documented in [11].
- WP5.3: Further development of conflict resolution methods. In order to match the A³ ConOps requirements further development of the conflict resolution methods is necessary. WP5.3 concentrated on developing those methods. Deliverable D5.3i [24] has already documented the initial indications of the methods chosen to further develop within the WP. The final deliverable D5.3 [6] documented the final results of the work undertaken within this sub-WP.
- WP5.4: Validation of the resulting conflict resolution method against the requirements. The aim of this sub-WP is to compare the resulting conflict resolution methods against the best currently known by the

autonomous aircraft research community and against the requirements identified in WP5.2. The results of the validation are presented in this deliverable.

The first three sub-WPs of WP5 have been focused on the identification and development of conflict avoidance algorithms for application in the A³ ConOps of iFLY. Following the structure of the Separation Assurance (SA) system described in the ConOps [7], the work there has been directed towards Short and Mid-term CD&R levels, as defined using the time-to-conflict criterion¹.

For the Short-term level the Decentralised Navigation Functions (NFs) framework has been chosen and significant development and refinement has been performed to derive a CD&R algorithm suitable for the needs of Air Traffic Management (ATM) that maintains the powerful formal properties of Navigation Functions (NFs). The specific performance constraints of (civilian) aircraft and current ATM practice have been taken into account for the development of the method.

In the Mid-term level Model Predictive Control (MPC) has been selected to cope with the performance optimisation requirements, longer lookahead horizon and wind disturbance. Two directions of development for the Mid-term CD&R have been pursued using the Decentralised MPC framework, one focusing on the integration of prioritisation as specified in the ConOps and another one aiming to robustify the calculated resolution manoeuvres against wind uncertainty.

The details of the algorithm developed have been described in deliverable D5.3 of iFLY [6] and some initial simulation results using small scenarios have been presented.

1.2 Objective of this deliverable

The aim of this document is to present the initial validation results of the conflict resolution methods that have been developed in the previous sub work-packages of WP5. In order to assess the performance of the algorithms in demanding multiple aircraft conflict scenarios reflecting the increased future traffic levels, realistic air traffic samples provided by Eurocontrol Experimental Centre have been used.

According to the Concept of Operations developed in WP1, the SA mechanism is composed by three levels, Long, Mid and Short-term, according to the time-to-conflict. Long-term SA falls into flow management and is outside the scope of WP5, while CD&R is handled in the Mid and Short-term levels. The requirements and specifications for set by the ConOps developed in WP1 have been taken into consideration for each different level of

¹Short-term CD&R handles conflicts up to about 5 minutes away, while Mid-term resolves conflicts tens of minutes ahead (usually around 20 minutes)

CD&R. An overview of the different surveillance scopes used in the three CD&R levels is shown in Figure 1.

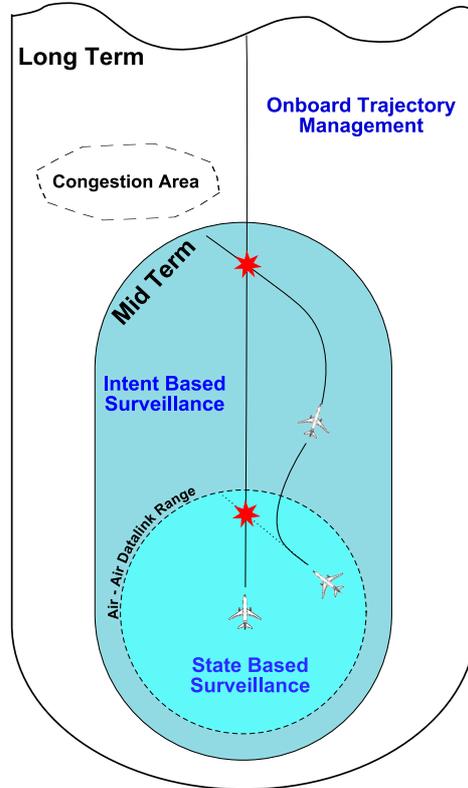


Figure 1: Air Traffic Surveillance in Short, Mid and Long Term CD&R

The refined CD&R algorithms developed within WP5 have been detailed in deliverable D5.3 [6] of iFLY. For Short-term CD&R the proposed algorithm is based on the NF framework [19], offering formal conflict avoidance and convergence properties. Optimisation techniques and especially MPC [17] have been employed for the Mid-term CD&R algorithm, allowing the integration of intent information and performance criteria. Moreover, one possible of integrating the Short and Mid-term CD&R levels into the Airborne Separation Assurance System (ASAS) system using a hierarchical structure has been suggested in the deliverable.

In order to evaluate the capabilities of the final algorithms in realistic scenarios, a large-scale traffic sample has been used in simulations. The validation of the CD&R algorithms has been performed independently, in order to reduce the computational requirements and allow better evaluation of each individual CD&R level.

1.3 Organisation of this deliverable

The rest of this document is organised as follows: the air traffic sample used for the algorithms validation is introduced in Section 2. The validation procedure along with the results for the Short-term CD&R level are presented in Section 3 followed by the validation of the Mid-term algorithms in Sections 4 and 5. Finally, the conclusions of the validation process are summarised in Section 6.

2 Realistic Air Traffic Sample

In order to assess the performance of the CD&R algorithms developed within WP5, simulations against a realistic air traffic sample have been performed. The traffic sample used is one which has been developed for the Episode 3 project for use in initial validation of the SESAR Target Concept [23]. It contains about 3 times as many flights as on the peak day in 2006. The sample contains flights in the European Civil Aviation Conference (ECAC) area over a period of 48 hours. Most of the flights enter the ECAC area in the second 24-hour period, though the sample also contains those flights which begin on the preceding day but which are aloft within the ECAC area at the start of the nominal day of the traffic sample. The traffic sample is effectively an estimation of airline demand without modification of departure times to avoid runway or airspace congestion. The sample contains almost 98000 flights in Europe which are airborne on a given day. Each flight is described by the departure and arrival airport and the planned take-off time, while no information about intermediate waypoints and flight altitude is given.

In order to keep the computational cost of the simulations manageable, a $400nm \times 400nm$ interest area centered around Zurich has been studied, as show in Figure 2. This represents a busy part of central European airspace, as indicated by the 35000 flights that fly in this area over the 24 hour period. Since most of the flights that cross the interest area do not land or take-off inside it, the start and destination positions have been set at the intersection of the straight line between each flight's departure and arrival airport and the boundaries of the interest area. The effect of the earth's curvature has not been taken into account here for simplicity since the area considered is relatively small. Moreover, since the purpose of the simulations performed here is not to calculate accurate trajectories but to evaluate the capabilities of the CD&R algorithms when handling large traffic samples, the precision of these conventions is not significant for the final conclusions about the algorithms' performance. The evolution of total traffic inside the interest area is shown in Figure 3. This estimation has been obtained by assuming that each aircraft flies straight to its destination without any CD&R taking place.

Since the aircraft type is provided for each flight, the performance parameters used in each set of simulations have been extracted from the BADA aircraft performance model [8] according to the flight level chosen for each case. More specific details about how the traffic sample has been used in each set of simulations are given in the corresponding chapters (3, 4).

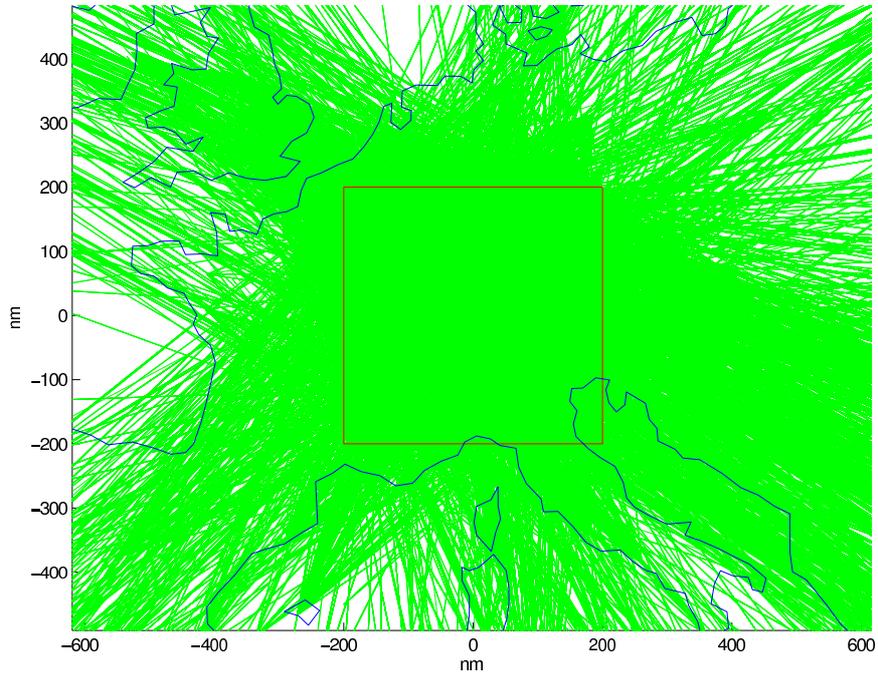


Figure 2: Flights crossing the interest area studied

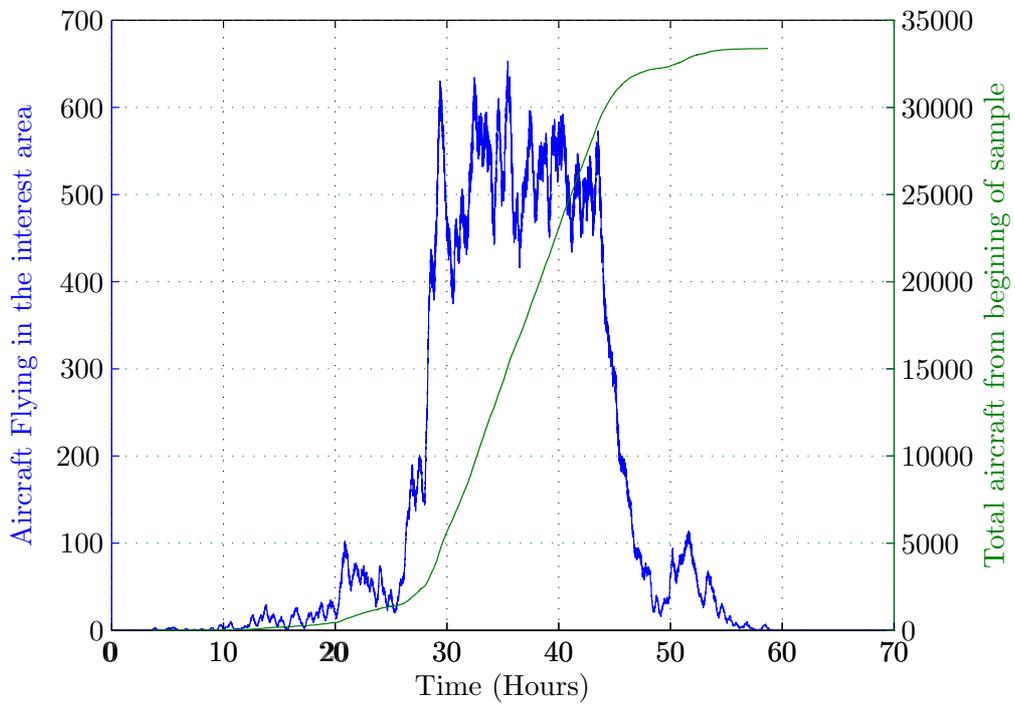


Figure 3: Evolution of sample traffic inside the interest area

3 Validation of Navigation Functions Short-term algorithm

3.1 Short-term CD&R using Navigation Function (NF)

The Short-term CD&R algorithm developed within WP5 is based on the powerful Decentralised Navigation Functions (NFs) methodology. This framework employs an artificial potential field for each aircraft that is repulsive with respect to conflicts with other aircraft and attractive to its goal (destination). Each aircraft is driven along a flow line of its potential field, thus avoiding all possible conflicts and reaching its goal. The key contributions to the NF methodology that have been made within WP5 of iFLY are the consideration of aircraft performance constraints and ATC practice [20] and the integration of an adjustable local awareness scheme that improves the resulting trajectories and reduces the computational load [21]. Moreover, the use of the NF framework enables the Short-term CD&R algorithm to provide formal guarantees for conflict avoidance and convergence to the goal.

The resulting Short-term algorithm is completely decentralised, since each aircraft needs only information about other aircraft inside its awareness zone and calculates its control inputs independently. Thus no form of clustering is used or required and no communication or negotiation is performed. Combined with the feedback nature of the control scheme, this allows a rapid recalculation and update of the controls, taking into account any modelling errors and environmental disturbances (eg. wind).

The details on the decentralised potential construction can be found in [21]. The main principle lies in the use of a non-circular awareness zone vs. circular zones used in previous NF approaches with limited sensing. The choice of the shape of the awareness zone enables significant improvements in the resulting trajectories and gives rise to implicit prioritisation resembling a *rules of the road* scheme. In order to adapt the algorithm to aircraft CD&R, the shape chosen for the awareness zone of each aircraft consists of a semicircle in the rear semi-plane and a semi-ellipse in the forward semiplane allowing longer range, as shown in Figure 4a. The notion behind this choice is that each aircraft should exploit its full Short-term information range in the forward direction to allow timely avoidance of potential conflicts, but limit the effect of other aircraft on its sides and rear since these are not as threatening and can induce undesired deviations.

Such an awareness zone shifts the attention of each aircraft mostly towards other aircraft flying in front of it rather than behind it and creates implicit prioritisation via asymmetrical sensing between two neighboring aircraft, as shown in 4b. Thus, the effect of other aircraft flying behind the own aircraft is significantly reduced, both in terms of the resulting trajectory and the computational effort. It is important to note, that this adaptation of

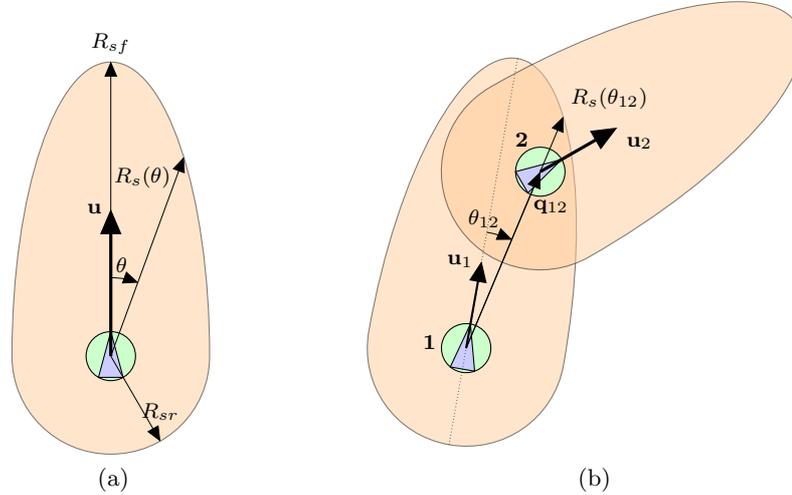


Figure 4: (a): The non-circular awareness zone used for each aircraft
(b): Implicit prioritisation of aircraft 2 wrt aircraft 1: Aircraft 1 is outside aircraft's 2 awareness zone, while aircraft 2 is inside aircraft's 1 awareness. Thus, only aircraft 1 considers aircraft 2 (but not vice versa) and only aircraft 1 will manoeuvre initially, unless it enters aircraft's 2 awareness zone, resulting in a cooperative resolution.

the NFs algorithm has not affected the formal conflict avoidance and convergence guarantees that are the main advantage of this class of methods.

A simple scenario involving a single aircraft and a static obstacle (i.e. an area-to-avoid) on its path has been used to demonstrate the benefits of the improved awareness scheme. Three different shapes have been used for the awareness zone, a small circle, the mixed circular-elliptical scheme presented above with the radius of the small circle in the rear semiplane and a longer range forward, and a large circle with a radius equal to the maximum range of the previous case. The results are shown in Figures 5a, 5b and 5c, as well as in Table 1. It is clear that the non-circular awareness zone shape yields the best results, by allowing the aircraft to fly a smoother and shorter trajectory while reducing the computation load of the algorithm.

Sensing Scheme	(a)	(b)	(c)
Computation Time	25sec	20sec	26sec
Path Length	7.56	5.13	6.59
Total steering angle	6.11	3.66	5.00

Table 1: Simulation results: Obstacle avoidance using various sensing schemes

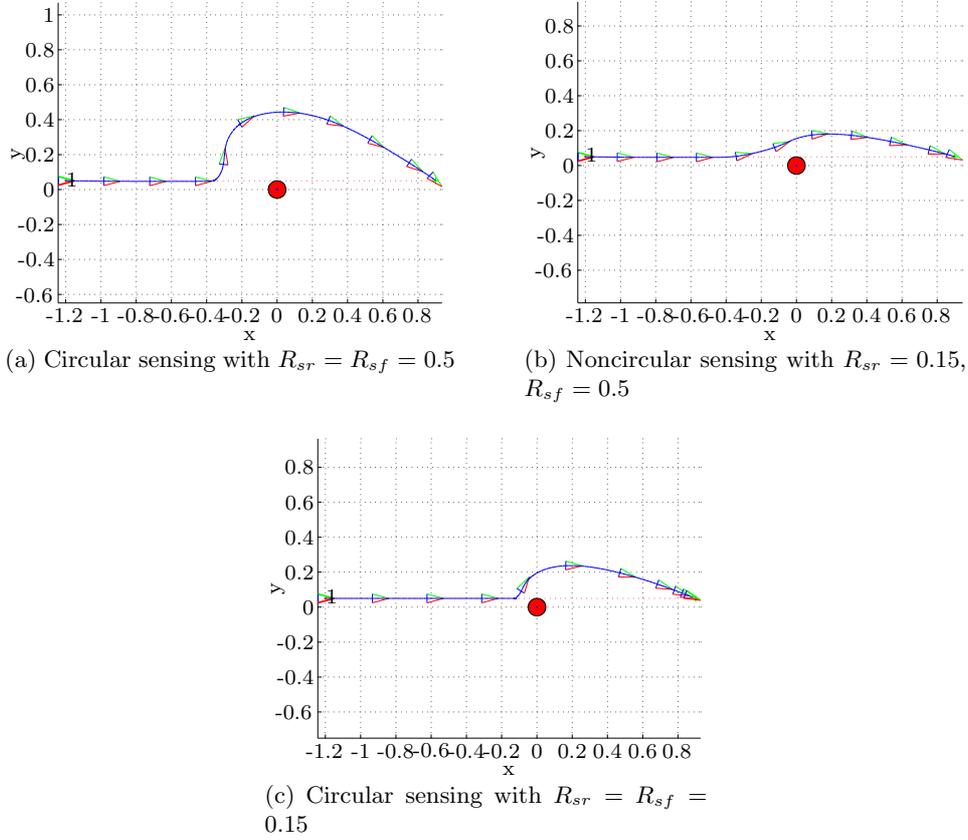


Figure 5: Simulation results: Obstacle avoidance using various sensing schemes

The main principles of the control scheme employed in the validation here are:

- The preservation of the formal conflict avoidance guarantees of the NF methodology
- The reduction of unnecessary manoeuvring and deviations
- The use of a constant speed whenever possible
- Compatibility with Air Traffic Control (ATC) practice

The control scheme used for the navigation of each aircraft is detailed in [22] and in [20] for the case of 3D flight, though in this report planar flight has been assumed for all simulation scenarios.

3.2 Validation against a realistic traffic sample

3.2.1 Medium-scale scenarios

For the evaluation of the NF-based Short-term CD&R approach developed in WP5 a number of computer simulations has been performed, using the air traffic sample presented in section [section][2][2]. Since the scope of WP5 is limited to the en-route part of the flight, the initial and final positions of all flights have been set at the same flight level corresponding to the cruising altitude. For this initial validation it was preferred to be able to simulate each flight level independently, thus aircraft have been allowed to use only horizontal manoeuvres to resolve conflicts. Results indicate that for the traffic levels used vertical manoeuvres are not necessary, though it is reasonable to assume that they would improve the performance in terms of efficiency. Each aircraft in the simulation is governed only by the Short-term CD&R algorithm which has to resolve all conflicts rather than handle only those that are not solved by the Mid-term level. All computations have been performed in a single desktop computer using the implementation of the NF-based CD&R algorithm developed in MATLAB.

The first simulation setting is the simplest one, comprising the first 1000 flights that enter the interest area. This scenario spreads over a period of around 26 hours, although the traffic density varies over time as will be shown in the results. The Short-term CD&R algorithm handled the scenario successfully and no conflicts occurred. The results of this simulation give an overview of the performance of the algorithm: in Figure 6a the number of aircraft flying inside the interest area at each time instant is shown, while the next next Figure 6b presents the number of aircraft manoeuvring over time. As can be seen in the figures, during the first 10 hours only a few aircraft enter the interest area resulting in very few resolutions. Later on however traffic increases significantly, up to about 140 aircraft flying simultaneously, resulting in more resolutions, up to about 110 in progress simultaneously. Finally, in Figure 6c the distribution of the total number of resolutions performed by each aircraft while flying inside the interest area is shown. Most of the aircraft perform less than 10-15, while only a handful are required to perform more than 25 resolutions. It is expected that the integration of the NF algorithm with a Mid-term solution will limit the number of conflicts requiring Short-term action, allowing the Short-term level to handle only those that require prompt action in order to guarantee resolution. Additional information about the scenario are shown in Table 2. Specifically, the total number of flight hours gives an overview of the total flight time simulated for all aircraft, and suggests that on average each aircraft flew in the interest area for about 30 minutes, of which 72.9% of the time, or about 22 minutes, the Short-Term CD&R algorithm was performing a resolution. This figure is increased here since there is no Mid-term algorithm

to avoid some conflicts before the Short-term level is engaged. The average number of resolutions each aircraft has to perform inside the interest area is the mean of Figure 6c. The increase in total flown distance is calculated over the straight-line paths between each flight’s initial and final position and is shown to be quite small even without any optimisation eprformed by the Mid-term CD&R level. Finally, the execution time implies that using a single desktop computer for all 1000 aircraft is enough to run the NF-based algorithm more that 300 time faster than real-time. Significant gains in this aspect can be achieved by distributing the computation over all the aircraft and more efficient implementation of the CD&R algorithm.

Total Aircraft	1000
Total Flight-hours	495.3
Average % time in resolution	72.9%
Average number of resolutions	4.65
Total distance increase	1.15%
Total execution time (sec)	5460
Real time speed	326x

Table 2: Simulation Results: First 1000 flights

In order to accommodate more aircraft, simulations using multiple flight levels have been performed for the first 4000 flights in the traffic sample. The aircraft have been to divided to 4 different levels, using two different protocols. Each flight level is simulated independently since no vertical manoeuvres are considered that would cause aircraft from adjacent levels to interact. Thus, the choice of these two flight level allocation schemes used here is merely to provide traffic data for individual levels that can be used independetly with the Short-term algorithm. Although a comparison between the two schemes is discussed in the conclusions, flight level assignment is beyond the scope of this report.

For the first set of simulations the flight level assignment is performed according to the direction of the flight route, as defined by the straight line path between the initial and final position. The first flight level includes aircraft with a route direction between North and East (heading angle $0^\circ - 90^\circ$), the second routes flying between East and South ($90^\circ - 180^\circ$), the third between South and West ($180^\circ - 270^\circ$) and the fourth between West and North ($270^\circ - 360^\circ$). As expected, the distribution of aircraft between the different flight levels is not uniform, since some directions are busier than others and this relation changes throughout the day. The second set of simulations has been performed by assigning each flight one of 4 flight levels (mentioned as FL A to FL D here) in a round-robin fashion, i.e. the 1st flight to enter the interest area is assigned to FL A, the 2nd to FL B etc in a circular procedure. In this way the aircraft are distributed fairly

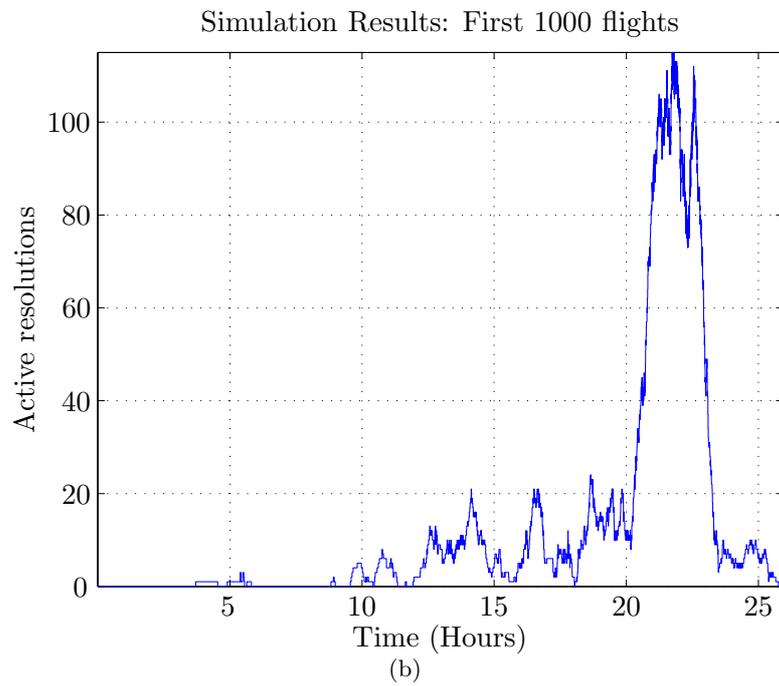
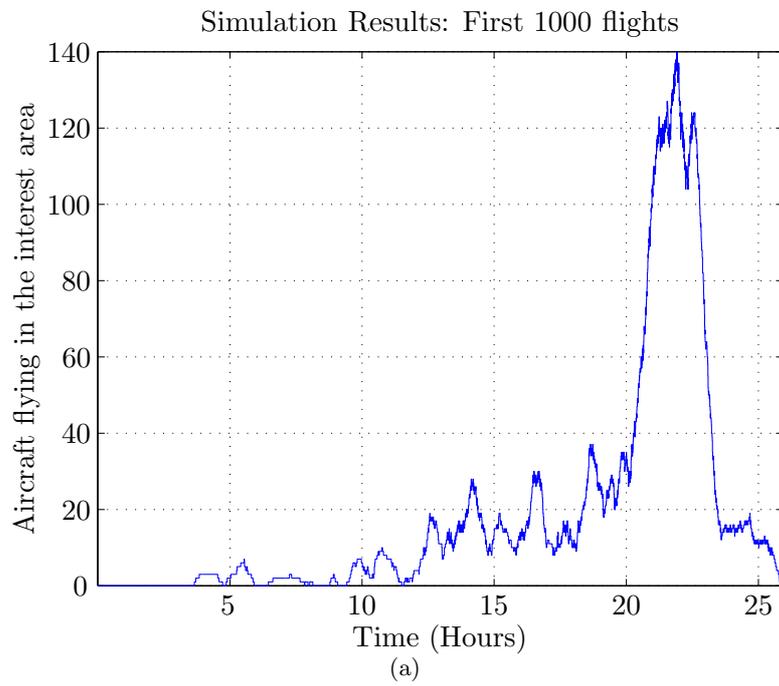


Figure 6: Simulation Results: First 1000 flights

between flight levels throughout time. The difference between the two FL assignments schemes can be seen in Figures 7a and , where the number of

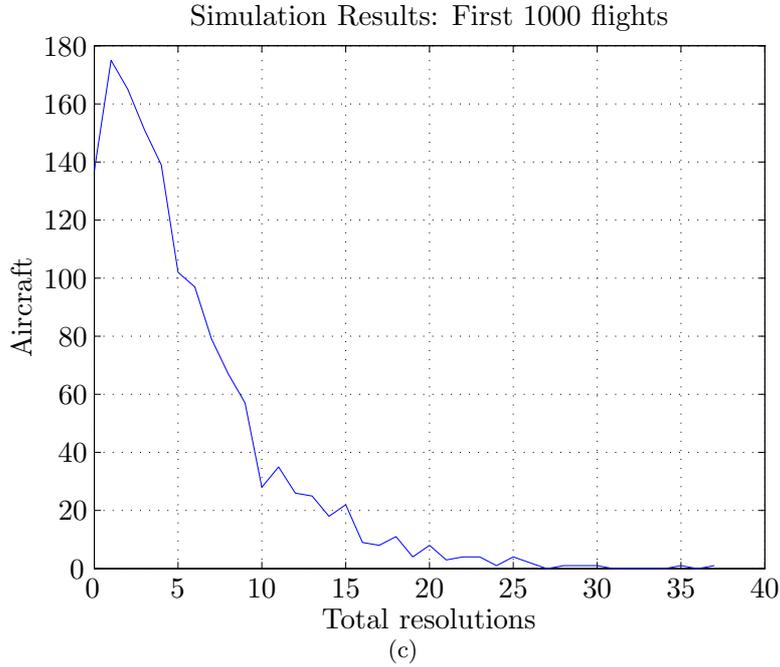


Figure 6: Simulation Results: First 1000 flights(cont.)

aircraft flying in each flight level is shown over time, without taking into account any CD&R.

The results for the first set of simulations, where FL is assigned according to the direction of flight, are presented in Figures 8a-8c and Table 3, where significant differences can be observed between among the 4 flight levels. As can be seen in the table, the total number of aircraft alone is not enough to characterise the resolution effort, as measured by the average number of resolutions per aircraft and the execution speed relative to real-time. Specifically, the N-E and S-W flight levels have a similar number of aircraft going through. However, as can be seen in Figure 8a, traffic in the S-W flight level is highly concentrated between 25 and 30 hours, while the traffic in the N-E level is spread out in a larger time period, significantly reducing the algorithm effort.

The results of the simulations using the round robin flight level scheme are shown in Figures 9a-9c and Table 4. Apart from the number of aircraft being the same, the number of active resolutions is also very similar between all 4 flight levels, since the distribution of flight directions is essentially arbitrary. Thus, the evolution of all the metrics presented in the figures and the statistics table are very similar for all flight levels.

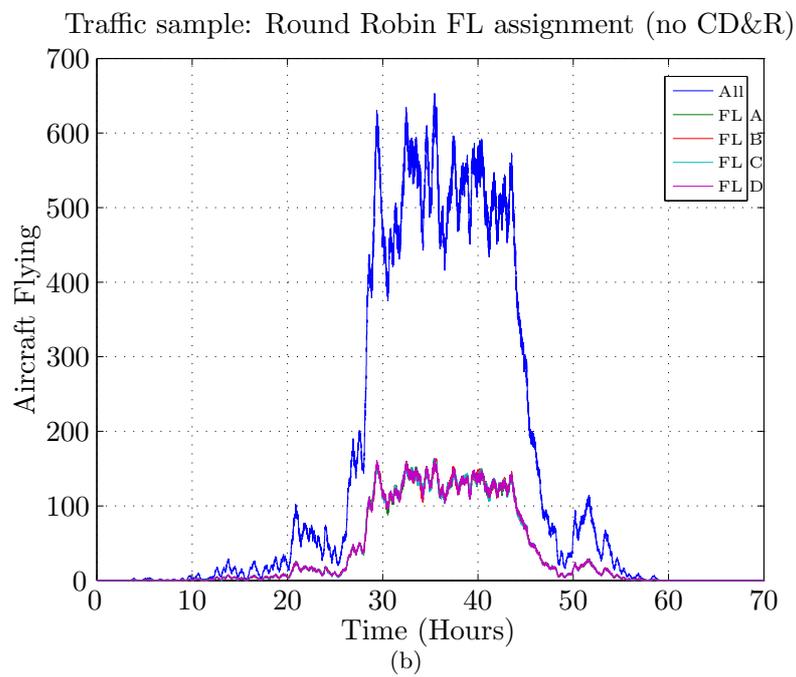
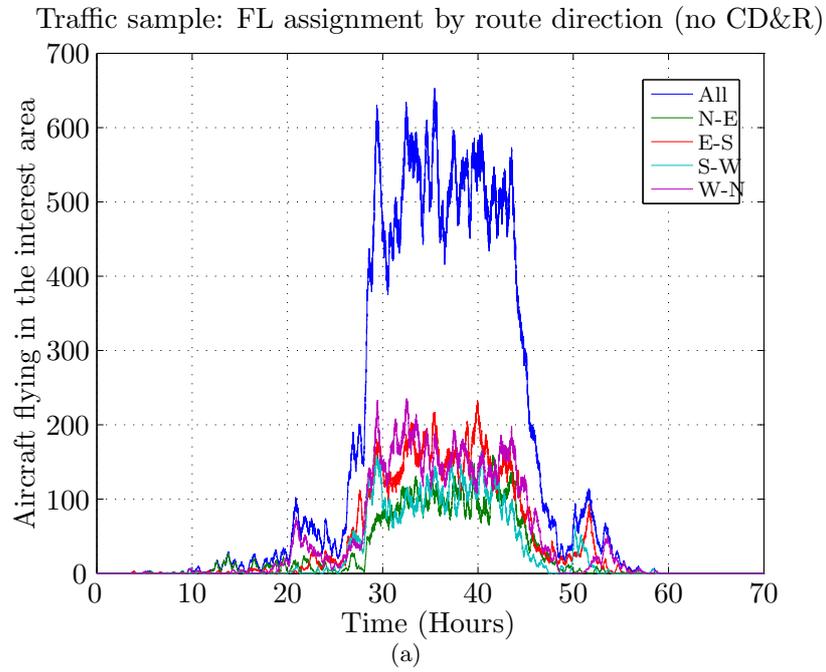
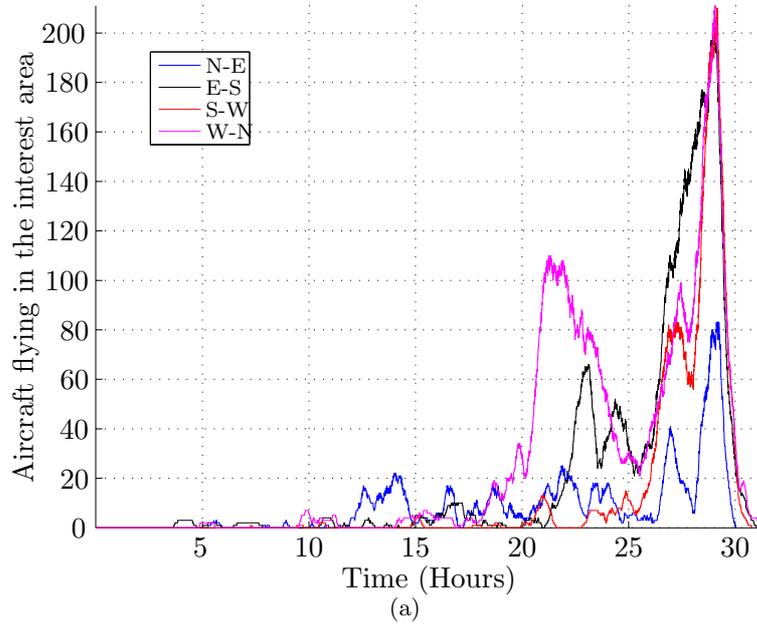


Figure 7: Estimation off traffic over time using different FL assignment protocols

Simulation Results: FL assignment by route direction



Simulation Results: FL assignment by route direction

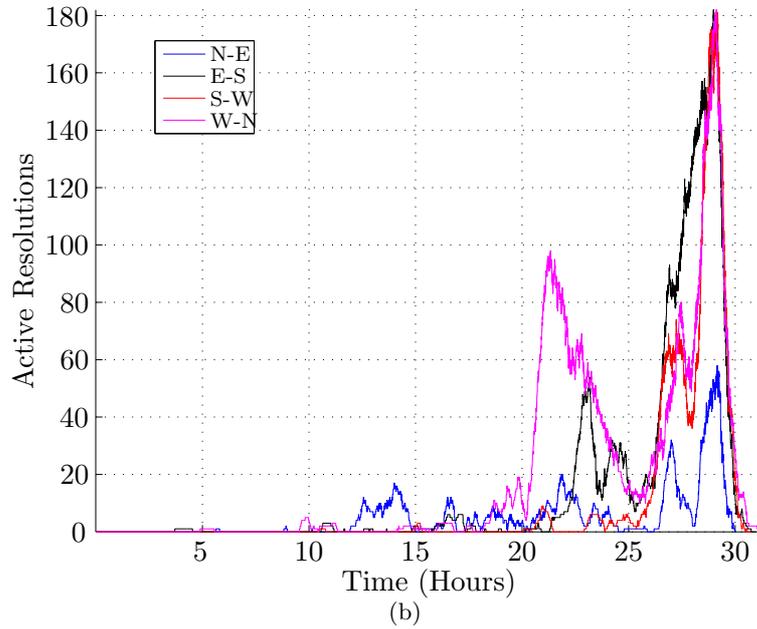


Figure 8: Simulation Results: FL assignment by route direction

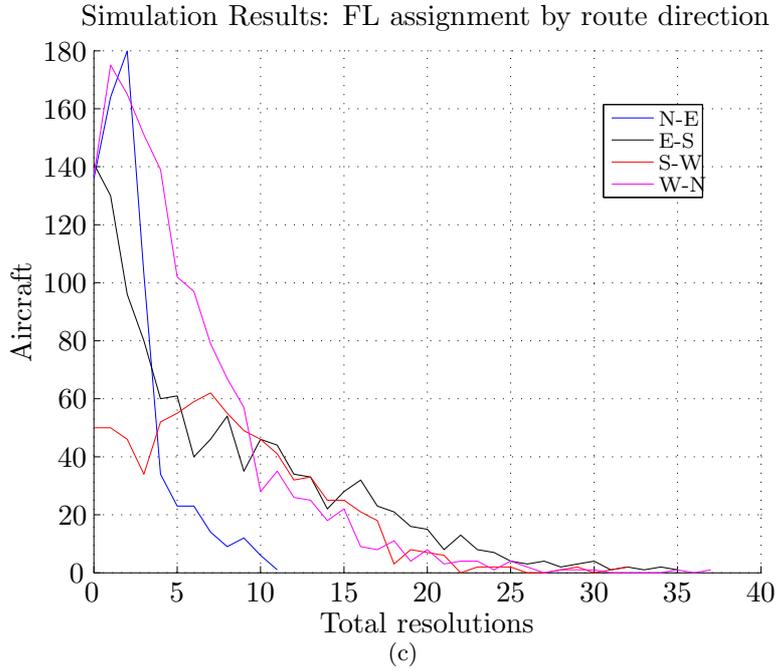


Figure 8: Simulation Results: FL assignment by route direction(cont.)

Route Direction	N-E	E-S	S-W	W-N
Total Aircraft	705	1120	789	1386
Total Flight-hours	258,49	675,28	419,57	840,29
Avg time in resolution	62,23%	79,58%	86,20%	79,12%
Avg number of resolutions	2,22	7,49	7,96	5,32
Total distance increase	0,393%	1,39%	2,559%	1,49%
Total execution time (sec)	1361	10976	6708	12320
Real time speed	683,7×	221,5×	225,2×	245,5×

Table 3: Simulation Results: FL assignment by route direction

Round-Robin Queue	FL A	FL B	FL C	FL D
Total Aircraft	1000	1000	1000	1000
Total Flight-hours	523,63	521,37	522,75	525,32
Avg time in resolution	69,41%	67,42%	70,37%	68,71%
Avg number of resolutions	7,25	6,96	7,58	7,42
Total distance increase	2,39%	2,37%	2,44%	3,19%
Total execution time (sec)	5886	5620	843	5985
Real time speed	320,3×	334,0×	322,1×	316,0×

Table 4: Simulation Results: Round Robin FL assignment

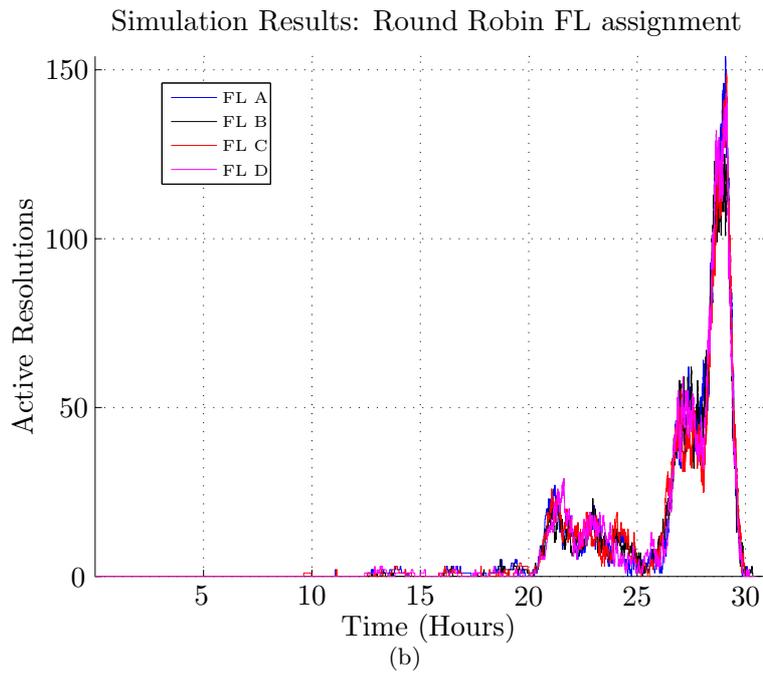
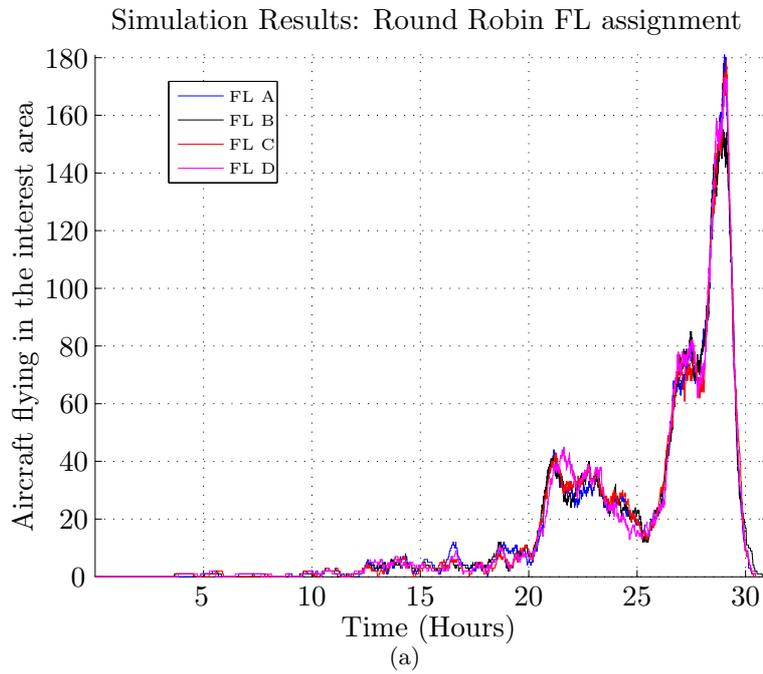


Figure 9: Simulation Results: Round Robin FL assignment

3.2.2 Large-scale scenarios

The simulation results presented in the previous chapter give a clear indication that NF-based short-term aircraft CD&R can be successfully used, at

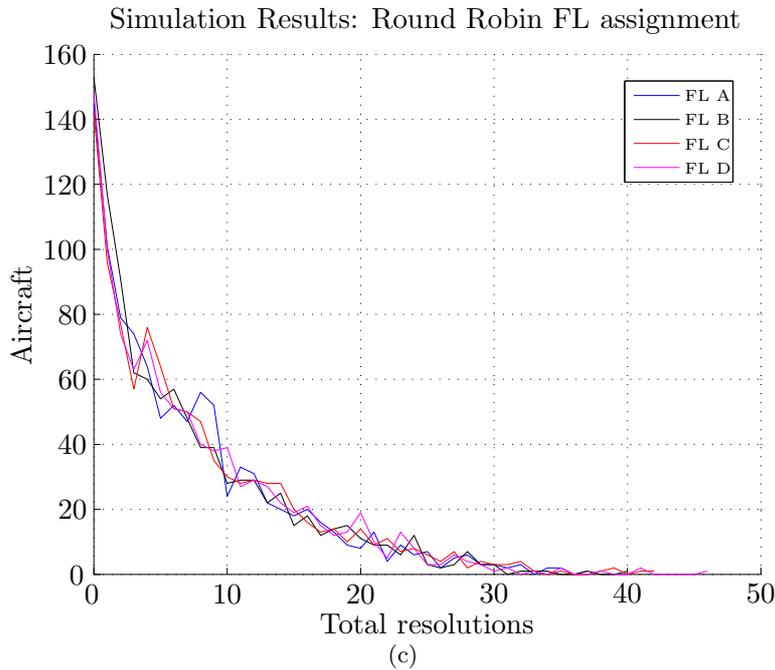


Figure 9: Simulation Results: Round Robin FL assignment(cont.)

least in the first 24-hours of the traffic sample used. As has been shown in Figure 3, the first 4000 which have been simulated in section 3.2.1 cover a little less than the first 30 hours of the traffic sample. including the beginning of the second day where total traffic is significantly higher. In order to give a clearer view of the NF algorithm performance under the heavy traffic conditions of the second 24-hour period, additional simulation scenarios have been used, chosen so that they span a significant part of the high-traffic period.

For these larger sets of simulations 16000 flights have been used, while the first 700 flights have been omitted to focus on the high-traffic period. Referring again to Figure 3, it can be seen there that this amount of flights span the first half (about 8-10 hours) of the high-traffic period. Thus the results presented below should give a reliable view of the NF algorithm performance under the heavy traffic levels found in the traffic sample. As in the previous section, flights have been divided to 4 different flight levels using the two previously presented protocols; the first set of simulations below uses the route direction to assign the flight level, while the second one uses the round robin approach.

The results using the first FL assignment rule are shown in Figures 10a-10c. As in the smaller scenario above, it is obvious that this rule produces a different traffic load for each flight level, according to the route direction of incoming traffic. It can be seen in Figure 10a that a total of almost 250

aircraft are flying for several hours in the two busiest flight levels. This heavier traffic was successfully handled by the proposed NF Short-term CD&R algorithm and no conflicts were recorded. Statistical information about this set of simulations can be seen in Table 5. Compared to the data in Table 3, the percentage of time in resolution and the average number of Short-term encounters are somewhat increased, due to the higher traffic density of the large-scale sample used here.

Route Direction	N-E	E-S	S-W	W-N
Total Aircraft	2679	4857	3212	5252
Total Flight-hours	1101,59	2340,54	1388,19	2536,84
Avg time in resolution	78,77%	86,91%	83,37%	84,92%
Avg number of resolutions	4,68	7,49	5,74	7,00
Total distance increase	0,86%	1,45%	1.26%	1,39%

Table 5: Large-Scale Simulation Results: FL assignment by route direction

Finally, the large-scale scenario of 16000 flights has been used with the round robin FL assignment rule, resulting in a uniform distribution of 4000 flights per level. The results of this set of simulations are shown in Figures 11a-11c. The total number of aircraft in each flight level remains above 100

Large-Scale Simulation Results: FL assignment by route direction

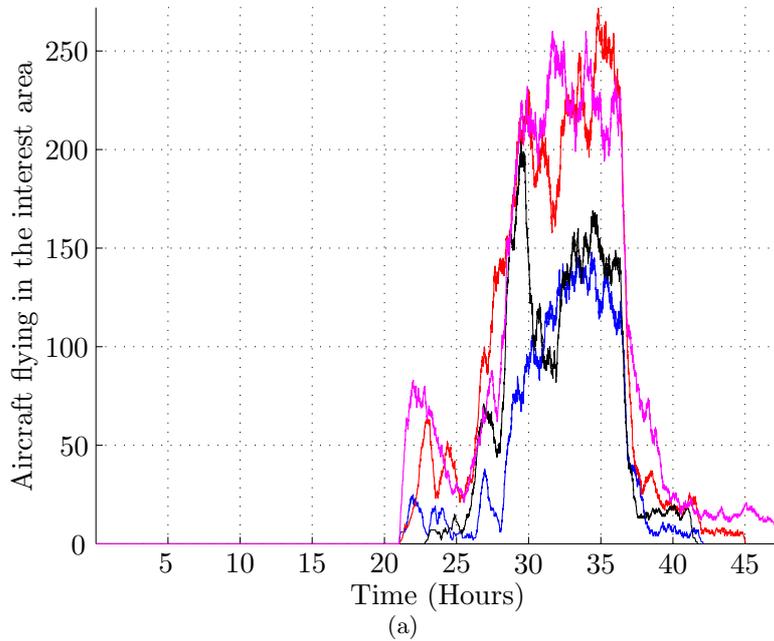
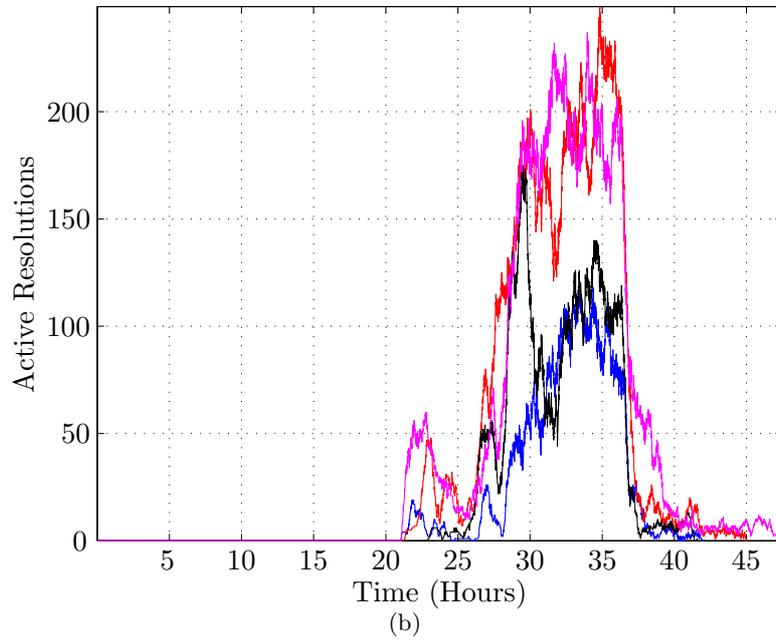


Figure 10: Large-Scale Simulation Results: FL assignment by route direction

Large-Scale Simulation Results: FL assignment by route direction



Large-Scale Simulation Results: FL assignment by route direction

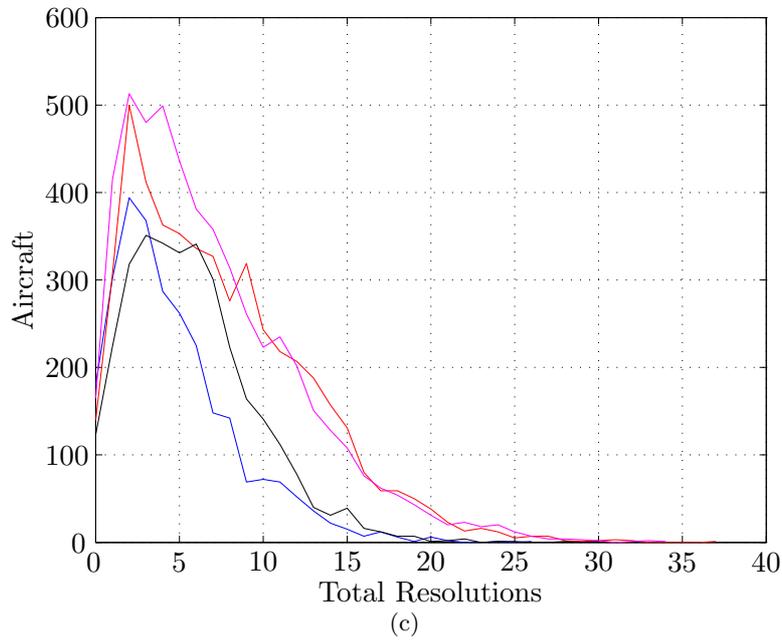


Figure 10: Large-Scale Simulation Results: FL assignment by route direction (cont.)

Round-Robin Queue	FL A	FL B	FL C	FL D
Total Aircraft	4000	4000	4000	4000
Total Flight-hours	1381,07	1362,55	1366,75	1365,72
Avg time in resolution	79,48 %	79,10%	79,04%	78,98%
Avg number of resolutions	6,47	6,45	6,57	6,51
Total distance increase	2.78%	2.88%	3.51%	2.89%

Table 6: Large-Scale Simulation Results: Round Robin FL assignment

for almost 10 hours, approaching 150 momentarily. The number of active resolution remains also high. Table 6 shows that the average time in each Short-term encounter is increased compared to the medium scale round-robin simulations (see Table 4) but the load remains equally distributed between levels. As in the previous tests, the NF algorithm resolved all conflicts successfully and no loss of separation was observed.

3.3 Conclusions

Overall, the results of the simulations suggest that NFs can successfully be applied to realistic aircraft scenarios. The decentralised nature of the algorithm means that its performance is not affected by the total number of aircraft.

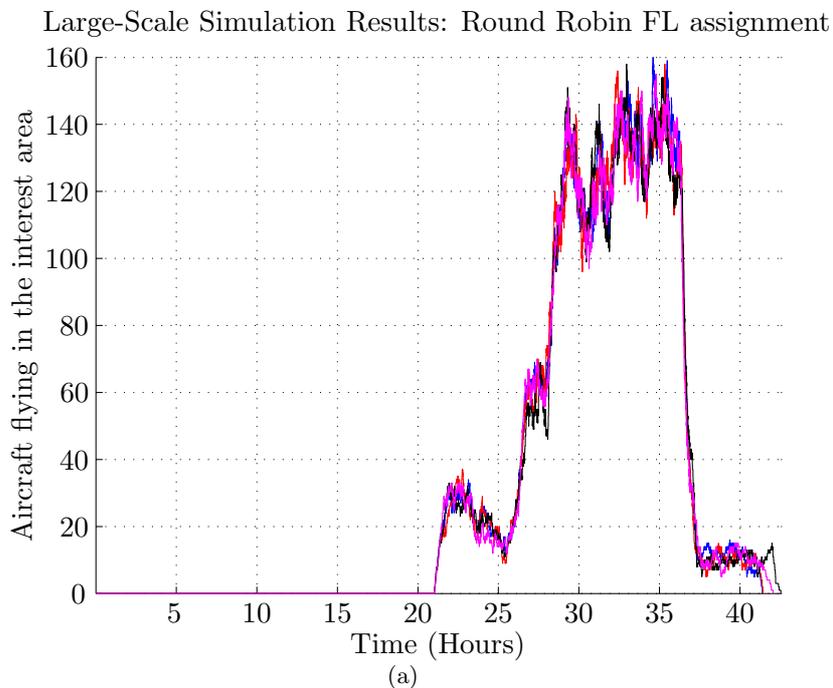
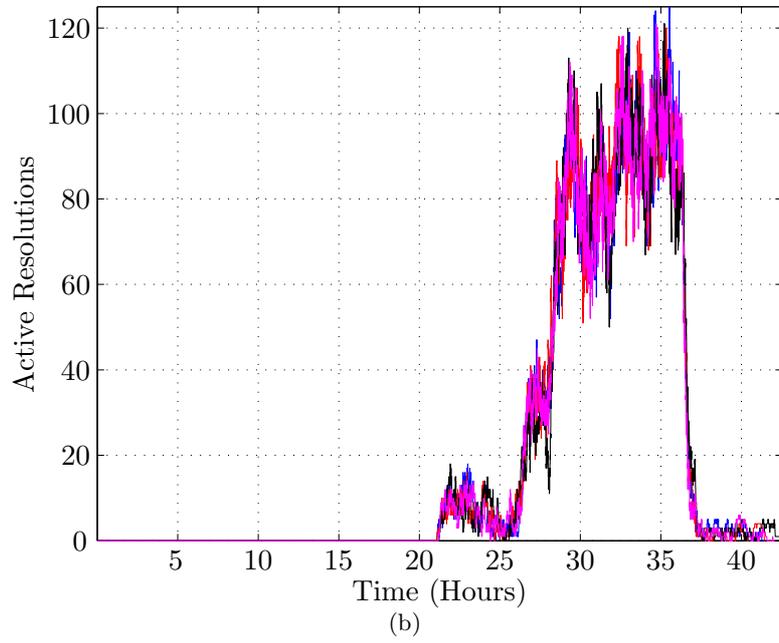


Figure 11: Large-Scale Simulation Results: Round Robin FL assignment

Large-Scale Simulation Results: Round Robin FL assignment



Large-Scale Simulation Results: Round Robin FL assignment

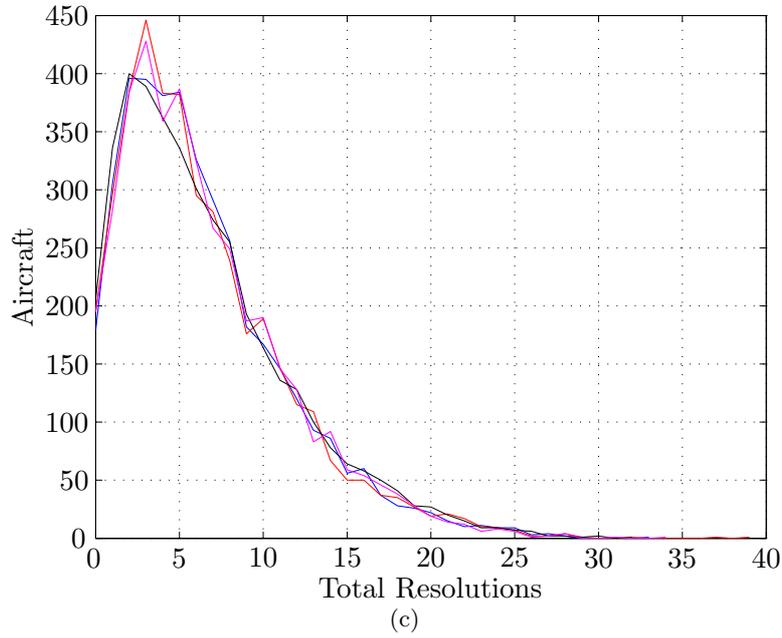


Figure 11: Large-Scale Simulation Results: Round Robin FL assignment (cont.)

The results of the two different protocols for dividing the aircraft into flight levels indicate that the path direction creates on average less conflicts to be resolved, thus allowing the aircraft to make less detours and fly shorter distances.

Further development in the area of Short-term CD&R using NFs may be directed towards the simplification of the resulting trajectories, using some motion primitives (lines, arcs, etc). Moreover, the incorporation of additional criteria for selecting the intruding neighbors that contribute to each aircraft's potential may be considered. This can be used to exploit conflict prediction rules based on all the currently available information.

4 Validation of Prioritized MPC Mid-term algorithm

In this Section, a Conflict Resolution (CR) method using prioritized MPC, which incorporates the use of priority rules which are proposed within the A3 concept [7], is outlined and validated against test scenarios, as well as a realistic traffic sample. The algorithm produces 2D resolutions and currently ignores vertical motion of aircraft. In the realistic traffic sample case, heuristics are developed in order to assign flights to different flight levels as well as to determine the group of aircraft that the resolution algorithm needs to be applied.

This method adopts a strict priority scheme, as the one proposed by iFly D1.3 [7], according to which higher priority aircraft may only maneuver in cases that a maneuver by **all** lower priority aircraft is not adequate to resolve the conflicting situation. In this section, the method is outlined and the results are presented; details on the method can be found in the Appendix A. First, the centralized version of the scheme is presented and then in a decentralized version of the scheme is developed.

4.1 Centralized Prioritized Conflict Resolution

The dynamics for a level flight cruise are, in general nonlinear. The CR problem can be described as the optimal control problem that determines the optimal (corresponding to some desired cost function) inputs for all aircraft such that they respect the following sets of constraints:

- *Velocity and acceleration constraints.* Aerodynamic reasons impose some physical constraints on the minimum and maximum True Air Speed (TAS) an aircraft can fly at each altitude. Furthermore, passenger comfort, as well as other human factors reasons impose constraints on the acceleration and the turning rate.
- *Conflict avoidance constraints.* All aircraft should remain separated at all times, by at least a minimum distance Δ , which is typically set to $\Delta = 5\text{nm}$ for cruising altitudes.
- *Priority constraints.* The priority concept used enforces the constraint that an aircraft will maneuver if (and only if) all aircraft with lower priority cannot satisfy the problem constraints without a maneuver by this aircraft.

The problem described cannot be modeled as a tractable optimization problem, due to the complexity of the nonlinear constraints. In order to be able to resolve it, a series of approximations that allow us to formulate the problem as a Mixed Integer Linear Program (MILP) is developed. Even

though the resulting optimization problem is still NP-hard, instances of realistic size in the ATM context are readily solvable by commercially available computational tools.

4.2 Centralized Formulation

The problem is modeled as a two-level hierarchical algorithm. At the highest level, a centralized MPC problem with simplified dynamics is solved, taking into account all constraints and generating an optimal set of inputs for each aircraft over a certain prediction horizon N . Once the optimal input sequences have been generated for all aircraft, they are pushed down to the lower level in the hierarchy, namely the Flight Management System (FMS). The FMS generates the appropriate inputs to be applied through the autopilot on the actual aircraft dynamics. The optimization problem is then resolved periodically and applied in a receding horizon fashion. This setup is illustrated in Figure 12.

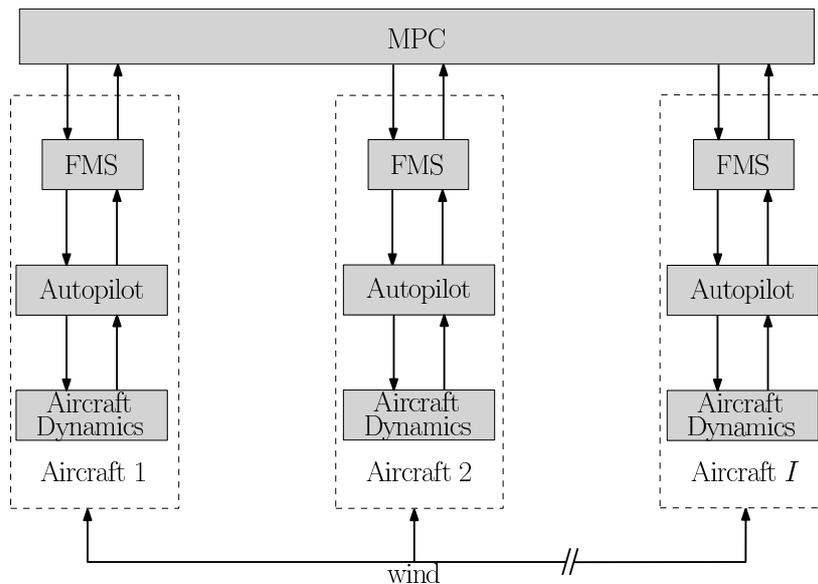


Figure 12: Hierarchical Multi-Level System

The MILP formulation of the problem has been constructed using the following simplifications:

Dynamical, velocity and acceleration constraints. The nonlinear dynamics are abstracted to a linear discrete-time model, based on single integrator dynamics. For velocity and acceleration constraints, instead of using 2-norm inequalities, simplified constraints using 1- and ∞ -norms are used.

Cost and priority constraints. Guided by the setup in [12], we define binary variables, one for each aircraft. Then, the cost is designed such that deviations from the nominal trajectories of the aircraft are penalized. Furthermore, a higher cost incurs for allowing a specific aircraft to maneuver, according to its priority. Using an appropriate weighting, we ensure that the priorities part of the cost dominates the deviation from nominal part. The binary variables ensure that the satisfaction of higher priority constraints always results in a lower cost than any possible combination of the lower priority constraints. In the air-traffic problem, this would mean that a higher priority aircraft will deviate from its nominal flight plan only if all other aircraft with lower priority cannot resolve the conflict.

Separation constraints. It is important that the formulation is robust against wind, so that conflicts do not occur even in the case of strong winds. Nevertheless, wind is generally unbounded, making a robust formulation against all possible noise realizations impossible. Instead, a confidence interval of the noise will be chosen against which the problem will be made robust. A “safe” choice would be to choose the 99.7% confidence interval, which corresponds to making the formulation robust against 99.7% of the possible wind realizations. The constraints are then conservatively approximated by ∞ -norms instead of 2-norms. Different sets of constraints are constructed, depending whether the wind correlation structure is taken into account or not. Furthermore, as the constraints are enforced in sampled intervals, rather than in continuous time (to maintain tractability), they are either enforced at a finer grid between the update samples or enforced with additional conservative constraints on the samples, making sure that a separation does not occur between the samples.

Constraint relaxation. Because of the model mismatch between the actual nonlinear model and the simplified linear one, as well as because of the fact that the tails of the wind error distribution are ignored, it is likely that aircraft find themselves in a situation where a feasible solution no longer exists. It is nevertheless important that even in such a situation, the algorithm provides aircraft with a solution, in order to avoid an actual collision. In such a configuration, aircraft should try to return as soon as possible to a feasible configuration, rather than continuing to fly in a potentially unsafe configuration optimizing over usual criteria. This is dealt, through the introduction of some additional binary variables, in a similar fashion to the priority binaries, but incurring a higher cost and allowing their actuation only to incur when otherwise the situation would remain infeasible. Then, the optimization problem becomes the maximization of the minimum guaranteed separation among all aircraft and fuel consumption is not still the most important parameter. More specifically, if maintaining a safety

distance is not possible for the whole optimization horizon of the MPC algorithm, the algorithm attempts to minimize the distance by which the safety separation is violated, while also keeping it as far in the future as possible. The rationale behind this is that since the algorithm is designed to find a robust resolution against all possible wind scenarios, the furthest in the future a safe separation cannot be guaranteed, the more likely it is that it will not be violated in practice, as worst-case scenarios for the designed uncertainty are unlikely to happen at all steps.

Feedback policies. Finally, two different open-loop strategies against a feedback one are evaluated. The basic difference between a feedback policy against an open-loop one is that in such a formulation, the fact that aircraft will re-optimize after the first parts of the inputs are applied is taken into account when calculating the inputs for the aircraft. In this fashion, the constraints do not get tighter as rapidly as in the open-loop case (see A). A heuristic, trying to maintain the lower computation time required by open-loop policies, assuming that aircraft will get to correct any deviations because of the wind, but not explicitly coding it in the optimization, but allowing the constraints to tighten only one step and then saturating them, is also presented.

4.3 Overall hierarchical formulation

The scheme proposed for the CR problem is summarized in Algorithm 1. The problem is solved every h minutes and when the optimal solution is calculated, only the first step is applied. Then, the first step of the optimal solution is translated through the FMS into thrust and bank angle commands that the autopilot implements on the aircraft dynamics for h minutes. The procedure is then repeated in a receding horizon fashion, until all aircraft reach their destination.

Algorithm 1 Prioritized Hierarchical Algorithm

Require: Initial aircraft positions

- 1: **while** Not all aircraft have arrived at their destinations **do**
 - 2: Measure aircraft positions
 - 3: Solve the MPC problem
 - 4: Translate the calculated inputs through the FMS and apply them through the autopilot for the first h minutes
 - 5: Measure new aircraft positions
 - 6: **end while**
-

4.4 Decentralized Hierarchical Formulation

While in the previous case, it is assumed that a CR algorithm can be solved on ground and up-linked, for instance through System Wide Information Management (SWIM), to the aircraft, there might be no computational capability on ground to perform a CR. In this, the priorities introduce a natural direction for decentralizing the problem. Assuming that the algorithm as well as the communications between aircraft are fast enough to ignore their effects in such a setting, a sequential approach is proposed. Given the priorities, each aircraft tries to resolve a simpler version of the centralized problem, having the ability to also determine control actions for aircraft with lower priority. In the case that the problem is infeasible (also in the centralized case), only the highest priority aircraft is allowed to calculate and announce a resolution that allows the violation of the constraints (since it is the only aircraft that can determine whether a resolution maneuver by all aircraft would be enough to avoid a constraint violation). Then, the lowest priority aircraft that finds a feasible solution, announces it and it is implemented by all aircraft.

The scheme proposed for the CR problem is summarized in Algorithm 2.

Algorithm 2 Decentralized Prioritized Hierarchical Algorithm

Require: Initial aircraft positions

- 1: **while** Not all aircraft have arrived at their destinations **do**
 - 2: Measure aircraft positions
 - 3: Set i = lowest priority aircraft
 - 4: **while** The decentralized MPC problem for aircraft i does not produce a feasible solution **do**
 - 5: Set i = the aircraft with the lowest priority among the remaining aircraft
 - 6: **end while**
 - 7: Translate the calculated inputs through the FMS and apply them through the autopilot for the first h minutes
 - 8: Measure new aircraft positions
 - 9: **end while**
-

4.5 Simulation setup and results

To evaluate the performance of the proposed algorithm for various choices of parameters, such as priorities and assumptions about the wind, a benchmark 4-aircraft encounter is used. The encounter under consideration is shown in Figure 13. All aircraft start at the same flight level and on a circle with a diameter of 250km. They all fly towards the diametrically opposite point on the circle. Nominally, all aircraft would collide at the same time at the

Case	Priorities	Constraint relaxation	Wind	Inter-sample constraints	Feedback	Decentralized
1	Yes	Yes	Correlated	Yes	No	No
2	No	Yes	Correlated	Yes	No	No
3	Yes	Yes	No wind	Yes	No	No
4	Yes	Yes	Uncorrelated	Yes	No	No
5	Yes	Yes	Correlated	No	No	No
6	Yes	Yes	Correlated	Yes	No	Yes
7	Yes	Yes	Uncorrelated	Yes	Yes	No
8	Yes	Yes	Uncorrelated	Yes	Saturation	No
9	Yes	Yes	Correlated	Yes	Saturation	No

Table 7: Alternatives tested in simulation

center of the circle. All possible priority configurations have been studied and in Figure 13 one of those possibilities is illustrated. The sampling period of the algorithm T_h is set to 3 minutes and the prediction horizon is set to $N = 5$. For the cases that separation constraints are enforced at inter-sample points, $L = 6$ is used, i.e. the constraints are enforced every 30 seconds. In all simulations, the MILP solver CPLEX [10] through the interface package YALMIP (see [15]) for MATLAB has been used. All algorithm alternatives are evaluated through 1000 Monte Carlo runs, representing 1000 different wind and priority scenarios. The algorithms are compared in terms of their effectiveness in resolving conflicts (by measuring the minimum separation between the simulated aircraft trajectory realizations), in terms of fuel burnt and in terms of computation times.

The different scenarios tested are presented in Table 7.

4.5.1 Effect of priorities

First the effect of the priorities in the resolution maneuvers is examined. Two cases are simulated:

Case 1 *Resolution with priorities:* In this case, as shown in Table 7, apart from imposing priorities on aircraft, constraints are allowed to be relaxed in case no feasible solution can be found, wind is assumed to be correlated, constraints are enforced on sub-intervals of the sampling period and the

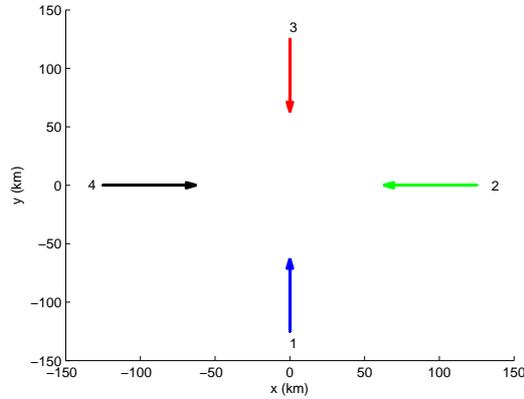


Figure 13: Conflict scenario with 4 aircraft

problem is solved in a centralized fashion without using feedback control policies.

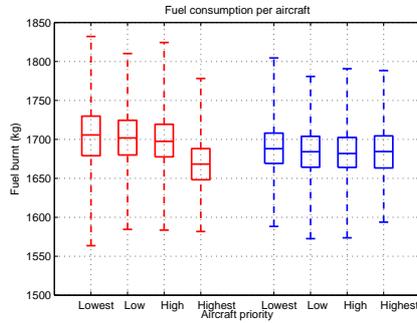
Case 2 *Resolution without priorities*: This case is exactly the same as case 1, with the exception that priorities are not used.

Statistics of the simulations are provided in Table 8. Data about the average fuel burnt per aircraft, the percentage of Monte Carlo runs in which conflicts occurred, the percentage of pairs of aircraft that were involved in conflict, the mean and maximum cpu time the solver needed to produce a resolution (for each time step, i.e. every three minutes) and the amount of cases that the separation constraints had to be relaxed are indicated. As shown, each aircraft burns on average 8kg more fuel when priorities are present. This is somewhat expected, as some aircraft are forced to fly larger maneuvers in order to avoid higher priority aircraft. Figure 15 presents a box-and-whisker diagram for the fuel burnt per aircraft in the two cases. As shown, the highest priority aircraft has a clear fuel advantage in the case that priorities are taken into consideration, while the other three aircraft contribute equally to the resolution in terms of fuel burnt. This happens because of the fact that in the cost considered (13), the penalty on the deviation of aircraft from their optimal inputs does not depend on the aircraft priority, i.e. if the three lower priority aircraft all have to maneuver (which is the case in our setup), then they contribute equally to the conflict resolution.

An important fact to note though is the higher amount of conflicts occurring when priorities are not present. As the constraints on the other hand were never in fact relaxed during the simulations, this is explained by the model mismatch between the simplified linear model (4) and the actual nonlinear dynamics (1); in the presence of priorities the inputs per aircraft

are more consistent throughout each simulation, leading thus to smaller errors between the two models. As shown in Figure 14a, even in cases that the separation is lost, it never drops below 4nm, while when priorities are not present (Figure 14b), the aircraft remain separated by more than 3nm in all cases.

The algorithm appears to be faster in the case that priorities are present, both on average as well as in worst case. This has again to do both with the fact that the solutions are consistent throughout the simulations, as the previous solutions are fed to the solver as an initial starting point. Furthermore, the number of extra binary variables needed when priorities are present is not big compared to the number of binaries required for the separation constraints.



Case	Mean fuel burnt (kg)	Conflicting scenarios	Conflicting pairs	Mean cpu time (sec)	Maximum cpu time (sec)	Constraint relaxations
Case 1	1693	0.8%	0.1%	0.4	3	0
Case 2	1685	9.4%	1.6%	0.8	10.8	0

Figure 15: Fuel consumption per aircraft (red:with priorities, blue:without priorities) Table 8: Comparison of prioritized and non-prioritized resolutions

4.5.2 Effect of wind simplifications

To examine the different possible wind simplifications, the following three cases have been simulated:

Case 1 *Resolution with correlated wind model*: In this case, as shown in Table 7, apart from the correlated wind assumption, priorities are imposed on aircraft, constraints are allowed to be relaxed in case no feasible solution can be found, constraints are enforced on sub-intervals of the sampling period and the problem is solved in a centralized fashion without using feedback control policies.

Case 3 *Resolution without wind*: This case is exactly the same as case 1, with the exception that wind used in the MPC algorithm is assumed to be zero, but the system is simulated with the actual wind model dynamics.

Case 4 *Resolution with uncorrelated wind*: This case is exactly the same as case 1, with the exception that the wind disturbances the MPC algorithm

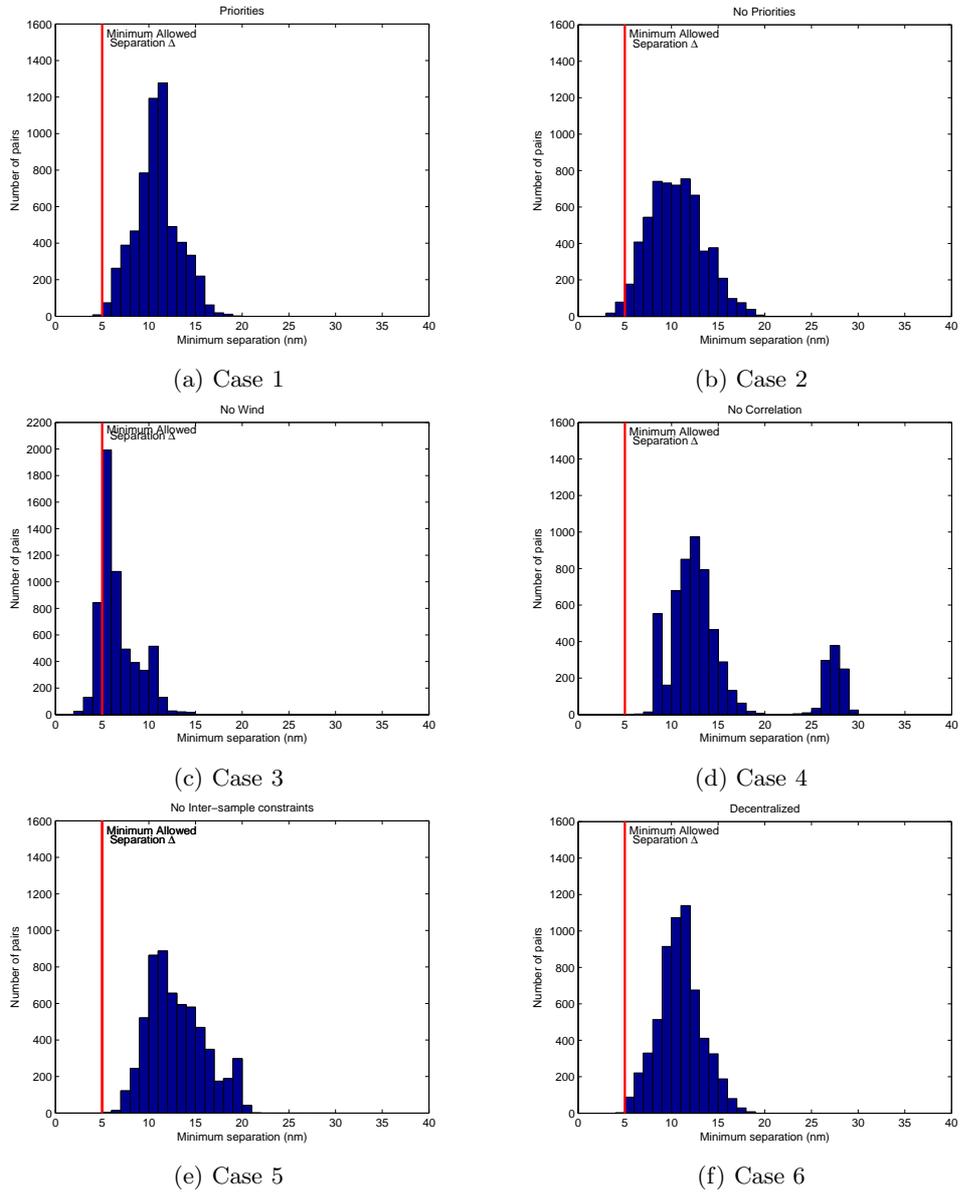
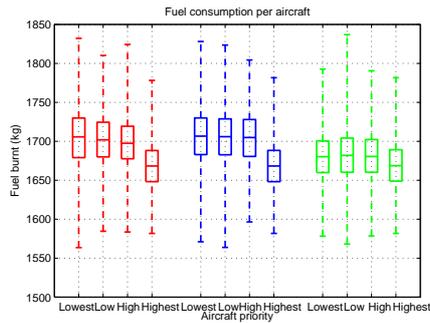


Figure 14: Minimum observed separations over all pairs of aircraft

are assumed to be uncorrelated, but the system is simulated with the actual wind model dynamics.

Figures 14c, 14d show the histograms of the minimum separations observed over all pairs. As the histograms suggest, ignoring the wind and solving a deterministic problem instead cannot adequately resolve the conflicts in several cases, leading to a substantial conflict probability. On the other hand, ignoring the correlation structure of the wind produces more conservative resolutions. This can be observed both in the histogram, where the minimum achieved separations between aircraft are now bigger, as well as in the fuel consumption, shown in Figure 16, incurring higher fuel consumptions on average.

Table 9 shows some more specific statistics regarding the simulations in these three cases. As shown, ignoring the correlation structure of the wind leads to an average 3kg more fuel burnt per aircraft. This indication is quite important, as this improvement in fuel consumption comes at moderate computational cost, as shown by the mean and worst case computation times. Finally, attempting to solve a deterministic problem, assuming that the wind speed is small enough to be ignored leads to many unresolved conflicts. The fact that the number of actual conflict occurring is much higher than the times that the constraints had to be relaxed in the MPC problem indicates that because of the assumption that wind is not present, it is not possible for the algorithm to detect conflicting situations and most of them actually occur between the sampling times of 3 minutes of the MPC. Thus, they are not perceived by the algorithm, since when it recalculates a resolution, the aircraft have already exited the conflicting area.



Case	Mean fuel burnt (kg)	Conflicting scenarios	Conflicting pairs	Mean cpu time (sec)	Maximum cpu time (sec)	Constraint relaxations
Case 1	1693	0.8%	0.1%	0.4	3	0
Case 3	1679	74%	17%	0.5	2	6.5%
Case 4	1696	0	0	0.5	7	0

Figure 16: Fuel consumption per aircraft (red:correlated, blue:uncorrelated, green:no wind) Table 9: Comparison of three different wind assumptions

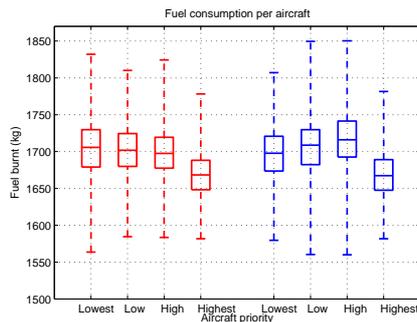
4.5.3 Separation constraint enforcement

An alternative that might be able to reduce the computational cost is, instead of checking the constraints on points in-between the MPC samples, to enforce some linear constraints on the binary variables and on how they are allowed to change values between consecutive time steps in the horizon. To quantify how much conservatism this approach adds to the formulation, a comparison will be done between the two following cases:

Case 1 *Resolution with additional constraints in-between MPC samples:* In this case, as shown in Table 7, apart from allowing the constraints to be relaxed, imposing priorities on aircraft, taking wind correlation into account, and solving the problem is solved in a centralized fashion without using feedback control policies, constraints are enforced on sub-intervals of the sampling period.

Case 5 *Resolution without additional constraints in-between MPC samples:* This case is exactly the same as case 1, with the exception that instead of imposing constraints in-between MPC samples, constraints are imposed on the binary variables used for the separation constraints.

Comparing the Figures 14a, 14e, the conservativeness introduced by this approach is obvious, as aircraft in order to avoid the conflicts maneuver much more, and are farther separated. This is also clear from Figure 17, where the fuel burnt is presented and the fact that more fuel needs to be consumed in order to resolve the situation in the case that the separation binaries are further constrained to avoid introducing more constraints between the MPC samples. On the other hand, as shown in Table 10, avoiding to introduce more binaries behaves better computationally, solving the problem much faster in this case.



Case	Mean fuel burnt (kg)	Conflicting scenarios	Conflicting pairs	Mean cpu time (sec)	Maximum cpu time (sec)	Constraint relaxations
Case 1	1693	0.8%	0.1%	0.4	3	0
Case 5	1696	0	0	0.3	1	20.4%

Figure 17: Fuel consumption per aircraft (red:constraints in-between samples, blue:No constraints in-between samples)

Table 10: Comparison of the two different constraint separation enforcements

4.5.4 Decentralized approach

In an ASAS environment, a CR algorithm should have the ability to operate in a decentralized fashion, onboard the aircraft. To evaluate its performance, this concept is compared against the centralized, ground-based CR with an equivalent formulation. In this decentralized approach proposed, as discussed, all aircraft are optimizing in parallel, with a different set of constraints each (see Section 4.4). Then, the aircraft with the lowest priority that finds a feasible resolution, announces it and all aircraft implement it. The two following cases are compared in order to quantify how much worse such a decentralized setting performs:

Case 1 *Centralized algorithm*: In this case, all the optimization is done on the ground. As before, the constraints are allowed to be relaxed, priorities are imposed, wind correlation is taken into account and no feedback is applied, while to maintain separation, the constraints are enforced on sub-intervals of the sampling period.

Case 6 *Decentralized MPC algorithm*: This case is exactly the same as case 1 in terms of configuration, with the exception that instead of solutions being calculated on the ground, they are calculated by the aircraft themselves.

As the configuration in this setting does not ignore any of the constraints of the centralized algorithm, it is expected that no additional conflicts will arise. Indeed, as Figures 14a, 14f suggest, aircraft behave in a very similar manner in the two cases. As Table 11 suggests, the extra fuel burnt by aircraft when resolving conflicts in a decentralized setting is only 1kg per aircraft. The computation times are a bit better for the decentralized case, but it should be noted that the computation times reported are the ones for the aircraft that actually takes the decision (i.e. the total computation is higher). Finally, Figure 18 shows the distribution of the fuel consumption among the involved aircraft, where the similar behavior of those two formulations is also observed.

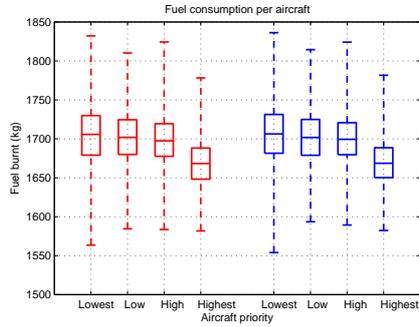
4.5.5 Open-loop vs Feedback approach

The last case examined is the several alternatives discussed to calculate the optimal input sequence. Three different alternatives have been simulated²:

Case 1 *Open-loop control*: This is the usual case, where the MPC is solved in an open-loop fashion with the typical configuration as before.

Case 7 *Feedback control policies*: This case differs from the setup in case 1 as an explicit, affine on the noise, policy is required for the inputs calculated. Furthermore, the correlation structure of the wind is ignored.

²The case of feedback control policies with correlation is ignored, as the results already suggest that the more conservative feedback control policies without correlation already violate the separation very often.



Case	Mean fuel burnt (kg)	Conflicting scenarios	Conflicting pairs	Mean cpu time (sec)	Maximum cpu time (sec)	Constraint relaxations
Case 1	1693	0.8%	0.1%	0.4	3	0
Case 6	1694	0.3%	0.05%	0.3	2.6	0

Figure 18: Fuel consumption per aircraft (red: Centralized, blue: Decentralized)

Table 11: Comparison of centralized and decentralized formulations

Case 8 *Open-loop control with constraint growth saturation:* This case is exactly the same as case 4 in terms of configuration, with the only exception constraints do not grow along the horizon, but are considered to saturate after a one-step growth.

Case 9 *Open-loop control with constraint growth saturation and correlation of the wind:* Here, the correlation structure of the wind is also taken into consideration. Otherwise, the setup is the same as in 8.

Once again, Figures 19b, 19c and 19d show the distribution of the minimum observed separations over all pairs of aircraft. Figure 19a is re-included for comparison purposes. In combination with Table 12, that provides some more statistical information about the simulations, several interesting facts can be extracted. Concerning the case that saturation is introduced, both in the correlated, as well as in the uncorrelated model, it appears that the distributions of the minimum observed separations among aircraft are shifted towards the minimum prescribed separation Δ . This allows for less conservative resolutions to be calculated, reducing the fuel consumption. In the case that feedback control is applied, a different shape appears in the distribution of the minimum observed separations (see Figure 19b). In this case, it appears that the algorithm attempts to push all aircraft to avoid the conflict tangentially, taking advantage of the expected knowledge of the noise. Nevertheless, because of the mismatch between the models, this is not always possible, and in several cases conflicts occur. It is still important to notice that in no case do the aircraft come closer than 3nm, indicating a mismatch of up to 2nm between the linear and the nonlinear models. In the presence of such mismatches in general, one should expect that an algorithm which can stress the model and find optimal solutions on the separation constraints, may violate the constraints in the actual nonlinear system.

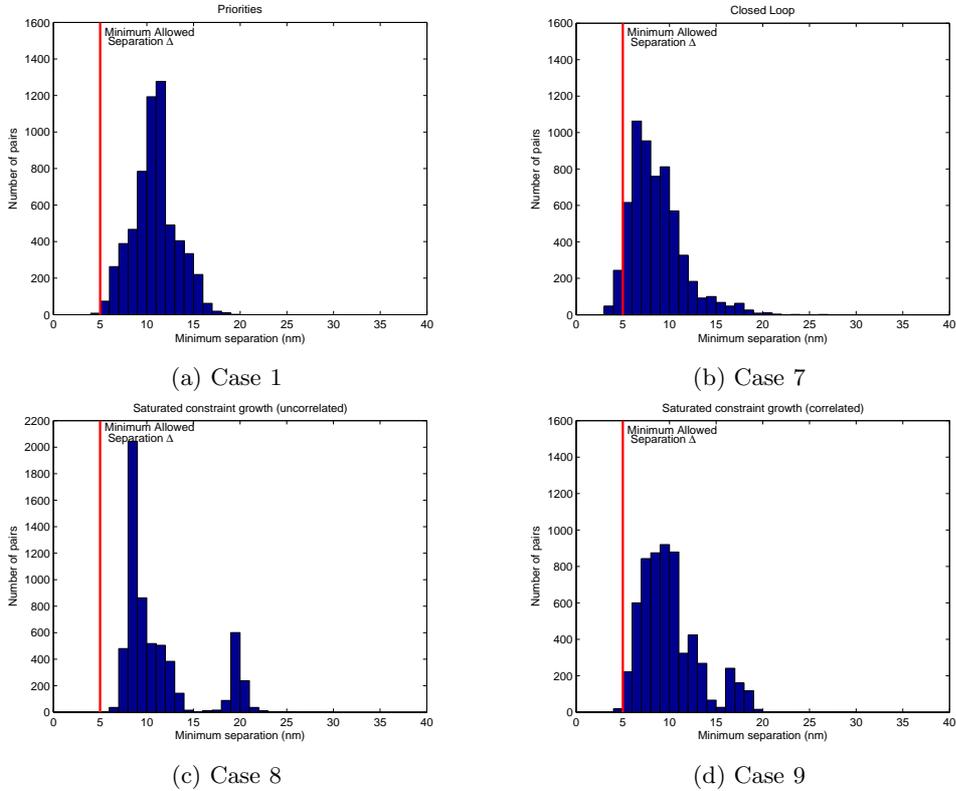


Figure 19: Minimum observed separations over all pairs of aircraft

The fuel consumption is in all cases better than the open loop control, as shown in Figure 20. An interesting fact is that saturated open-loop burns less fuel in some cases than the feedback control case, but this is probably due to the often occurrence of constraint relaxations that differentiate the cost. The computation times are similar for the three open-loop cases, while they are about two orders of magnitude higher in the case that the algorithm solves for closed-loop policies, as shown in Table 12.

4.5.6 Summary of simulations

Several algorithm configurations have been examined. From the corresponding results, it is clear that the cases that the constraints do not grow along the horizon are the best in terms of fuel consumption, while maintaining safe separation. When this heuristic is not used, the use of the correlation structure of the wind may be used to improve the algorithm performance, without introducing further complexity to the computations.

An important fact is the similar performance of the centralized and the decentralized algorithm, as well as the better behavior of the algorithm when priorities are introduced. In general, it seems that an open-loop MPC

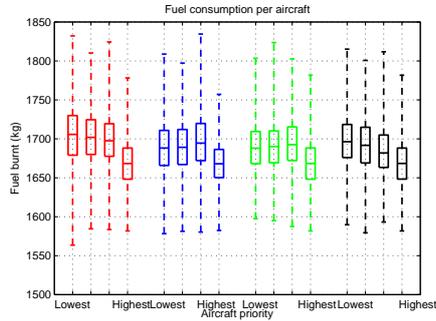


Figure 20: Fuel consumption per aircraft (red: Open-Loop, blue: Closed-Loop, green: Saturated Open-Loop (uncorrelated), black: Saturated Open-Loop (correlated))

Case	Mean fuel burnt (kg)	Conflicting scenarios	Conflicting pairs	Mean cpu time (sec)	Maximum cpu time (sec)	Constraint relaxations
Case 1	1693	0.8%	0.1%	0.4	3	0
Case 7	1686	19.8%	3.9%	38	750	0.3%
Case 8	1685	0	0	0.3	3.2	0.3%
Case 9	1686	0.18%	0.04%	0.4	6.3	5.4%

Table 12: Comparison between open-loop, saturated and closed-loop policies

algorithm, with a saturated growth of its constraints is the best fit for the CR purposes studied here. Nevertheless, this should not be considered a general result about the optimization itself, as the dynamics used in the optimization are not the actual nonlinear system dynamics, while the system performance is measured on the nonlinear ones.

4.6 Validation against a realistic traffic sample

In order to assess the performance of the CR algorithm developed, the traffic sample described in Section 2 has been used. For those flight plans, in order to determine the points that they enter and leave the controlled airspace, it is assumed that the take off and landing phases are carried out according to nominal operations, without any uncertainties. Then, the points that each aircraft would have to enter at each flight level, or start descending, along with the corresponding times are calculated. Those are the specific points and times in the future that each aircraft enters or leaves the area of interest. For flights that those points are outside the interest area, their entry and exit points to the area are set at the intersection of the straight line between their entry and exit point at each flight level and the boundaries of the interest area.

The parameters for each aircraft are taken from the Base of Aircraft Data (BADA) database [8]. To cope with the big number of aircraft of the traffic sample in a reasonable computation time, the wind effects will be ignored for this simulation. Furthermore, a heuristic on distributing aircraft on different flight levels will be used, as well as a rudimentary conflict detection in order to divide each scenario in smaller ones, that can be better handled by the MPC algorithm presented.

4.6.1 Multiple Flight Levels

In an actual ATM scenario, not all aircraft fly at the same flight level, but several levels might be used for this purpose. Despite the fact that airlines ideally want their aircraft to fly as high as possible (for better fuel consumption), this is not always possible, because of the limits of the number of aircraft that can be accommodated per flight level.

A natural way to assign each aircraft at a flight level is to divide the traffic according to their flight directions. This choice, apart from intuitive, is also appropriate for the MPC algorithm, as, since aircraft are flying on similar directions and airspeeds, if when an aircraft enters the flight level the problem is feasible within the prediction horizon, it will remain feasible throughout the simulation. The traffic sample is thus divided in 4 parts, according to four perpendicular directions, chosen such that aircraft are approximately equally distributed among them. Then, each aircraft is assigned to a different nominal flight level according to its direction (and its maximum operating flight level).

Thus, to determine at which flight level an aircraft should enter, the aircraft notifies a centralized MPC algorithm, which is assumed to be supervising all different flight levels, at which flight level and at which point it is planning to enter. This supervisory MPC checks, using the algorithm previously described in Section A.1, if on the requested flight level it is feasible to admit the aircraft or it would lead to a potential loss of separation. In the latter case, the aircraft is not admitted at this flight level, and the process is repeated for lower flight levels, until the aircraft is admitted at the highest feasible flight level, with aircraft flying along the same direction.

4.6.2 Identifying conflicting situations

In such a big traffic sample, it is important to identify (at each time step) all aircraft that are involved in a conflict within the prediction horizon and create several groups of conflicting aircraft to facilitate the resolution process. This is done by iterating the following steps:

- Identify all aircraft that are about to enter our area of interest within the MPC prediction horizon.
- Build an undirected graph with all aircraft as vertices and edges connecting the pair of aircraft that are involved in conflicts, assuming that all aircraft keep their current flight plans.
- Assume that all aircraft which are involved in conflicts maneuver and identify all other aircraft that some maneuver of the former ones might lead them to conflict, adding the corresponding edges and vertices in the graph. Repeat the process until no new edge or vertex is introduced

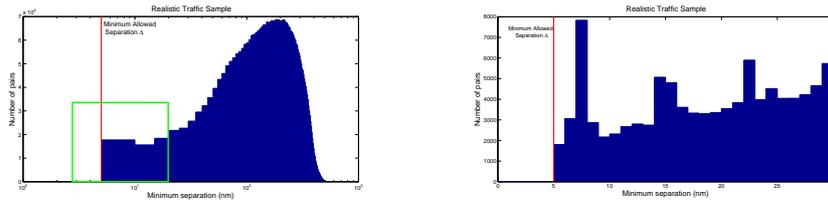
to the graph. From now on, this will be referred to as the *Conflict Graph*.

- Determine all the connected components³ of the Conflict Graph. For this purpose, Tarjan’s algorithm is used [25], which is already implemented in MATLAB.
- Resolve each connected component as a separate group of conflicting aircraft.

Using this approach, the size of the scenarios that have to be resolved at each time step remains moderate. Furthermore, for computational reasons, we have limited the allowed size of a connected component to 90. Thus, if a cluster of aircraft that need to be resolved reaches this number, no other aircraft are admitted if they would result in a bigger group to be resolved. Then, those aircraft are redirected to lower flight levels. This limitation arises mainly because of the available computational power at the time of the simulations and not with the algorithm itself.

For this traffic sample, the centralized version of the algorithm developed will be used, while the actual nonlinear dynamics are ignored.

In total 34238 aircraft from the traffic sample can fly higher than FL260, which is the lowest level simulated. Because of the limit imposed on the algorithm, 7104 aircraft were not admitted at any flight level. The minimum observed separations between all pairs are shown in Figure 21a, while in Figure 21b the highlighted area in Figure 21a is shown in detail. As shown, no conflicts arise and aircraft are safely separated under the assumptions that no wind arises and no mismatch between the actual nonlinear model and the linear one exists.



(a) Minimum observed separation for the traffic sample

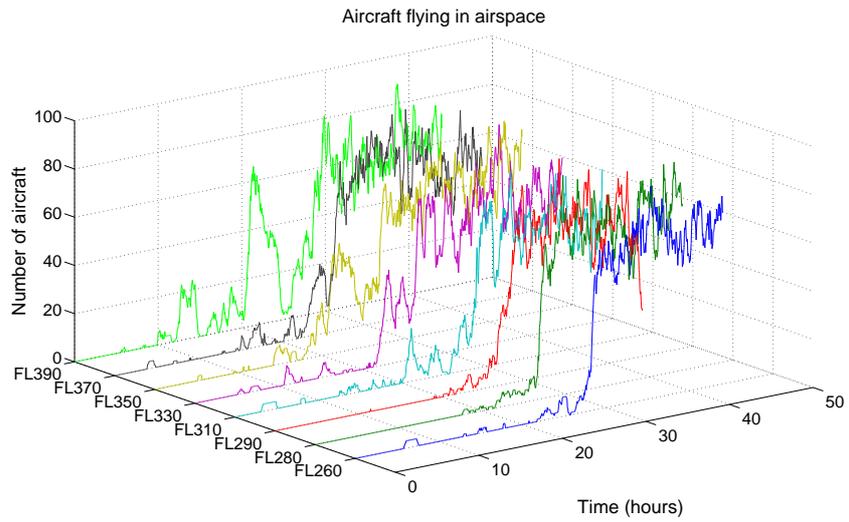
(b) Minimum observed separation for cases close to 5nm

Figure 21: Statistics for the traffic sample

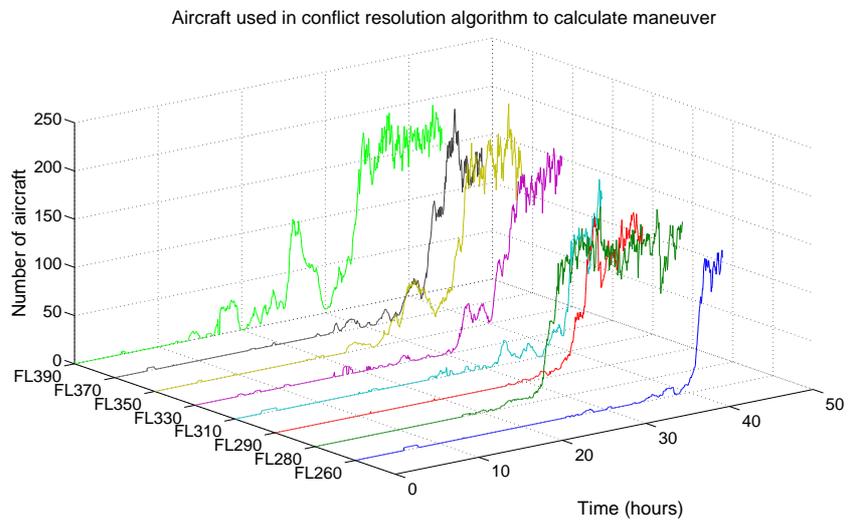
Furthermore, Figure 22a shows the number of flights in the interest area at all flight level at each time, while Figure 22b shows the number of flights that need to be taken into account when performing the CR algorithm (i.e. both the flights that are currently flying in the airspace, as well as those that

³A connected component of an undirected graph is a maximal group of nodes that are mutually reachable that are connected to each other by edges.

are requesting to enter within the airspace within the prediction horizon of the MPC algorithm). Those flights are divided in several groups, as shown in Figure 22c, the biggest of which comprises of a number of aircraft shown in Figure 22d. Finally, the amount of aircraft maneuvering at each time step is shown in Figure 22e.



(a)



(b)

Figure 22

Statistical results from the simulation of the traffic sample are provided in Table 13. As shown, the total extra flown distance because of the conflict resolution is less than 2.2% in all flight levels in order for the situation to

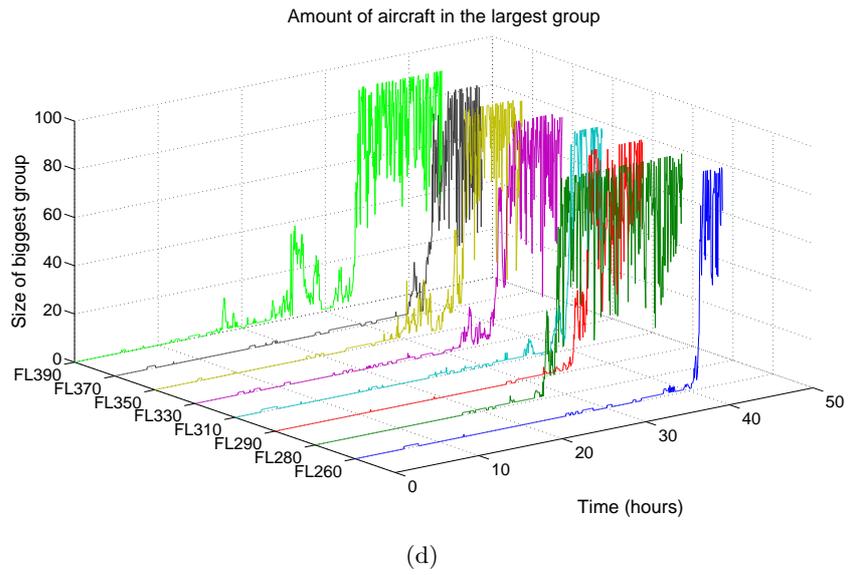
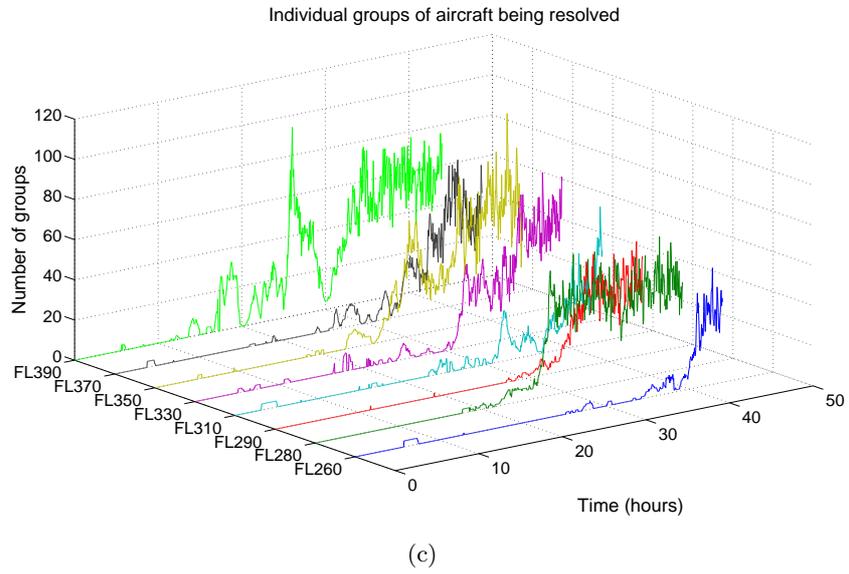
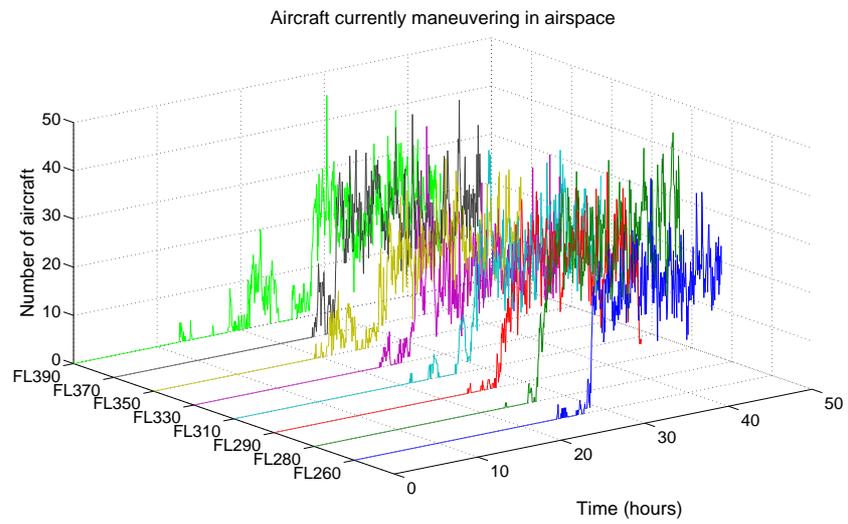


Figure 22

	FL260	FL280	FL290	FL310	FL330	FL350	FL370	FL390
Aircraft	2710	2731	3060	2860	3557	3761	4065	4390
Total extra distance flown	1.97%	1.83%	2.15%	2.06%	1.66%	1.48%	2%	1.64%
Runtime (min)	474	448	501	523	558	503	543	486

Table 13: Traffic sample statistical results



(e)

Figure 22

remain conflict free. One should note that the fact that aircraft might be flying in non-optimal flight levels is here ignored, as aircraft are assigned in a heuristic way to flight levels.

5 Validation of Distributed MPC Mid-term algorithm

In this section, we summarise findings obtained on application of robust multiplexed Model Predictive Control to both simple scenarios and a subset of the realistic air traffic samples considered throughout this report. Unlike the mid term conflict resolution algorithms presented in the previous section, the algorithms considered here are robust to the effects of wind uncertainty and provide theoretical guarantees of conflict avoidance and arrival at pre-determined target regions. The purpose of this section is to investigate the potential of this alternative mid term conflict resolution algorithm on more realistic scenarios and assess its limitations. We first provide a brief recap of distributed robust multiplexed MPC in Section 5.1. Results on simple scenarios are presented in Section 5.2. Finally, in Section 5.3, we present some of the results obtained on application of the modified robust MMPC algorithm to the realistic air traffic data, outlining the necessary additional assumptions and changes to the algorithms.

5.1 Recap of robust distributed MMPC

As detailed in [6], in its most basic form, the underlying protocol in distributed multiplexed MPC is that aircraft plan their future trajectories in a predefined cyclic sequence, taking into account the plans of other aircraft. Each aircraft involved in an encounter plans its own future trajectory, then transmits its future plan to the other aircraft. The next aircraft in the sequence does the same. In between updates, each aircraft executes disturbance feedback according to its planned policy until it is its next turn to replan. The effect of disturbance feedback is to reduce the impact of the future unknown disturbances. A centralised solution for aircraft initially entering the scenario, assumed to be obtained from SWIM, is used to initialise the distributed scheme.

Variants of the basic single aircraft update MMPC algorithm include parallel optimisation variable update order MMPC (P-MMPC), introduced in [6]. Here, aircraft optimise simultaneously in parallel, making the assumption that other aircraft do not modify their plans from the previous timestep. We examine the performance of this algorithm by comparison with the single update MMPC formulation and a baseline disturbance feedback controller on simulated scenarios.

To accommodate the evolving nature of the air traffic scenarios, that is, aircraft continually entering and leaving the scenario, it is necessary to make some changes to the basic MMPC algorithms presented previously in [6]. We demonstrate this with results obtained on simulated scenarios in which unanticipated aircraft enter a region containing existing aircraft executing robust MMPC.

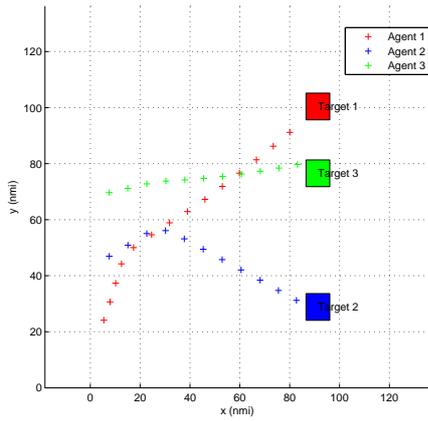
5.2 Results on Simple Scenarios

We define the interest area used in our scenarios below: The area of interest used in our scenarios was a rectangular region measuring 200x200 nautical miles. We first present a performance evaluation of the robust MMPC schemes on closed scenarios, where all aircraft involved in a potential conflict are identified in advance, and new aircraft do not enter the predetermined region of interest. The performances of the single update order MMPC and parallel optimisation variable update order MMPC (P-MMPC) algorithms are compared against that of a baseline single shot disturbance feedback MMPC controller. In execution of the baseline control, aircraft do not optimise for their control moves; rather, they continue to apply their fixed disturbance feedback policy obtained from the initial optimisation from SWIM.

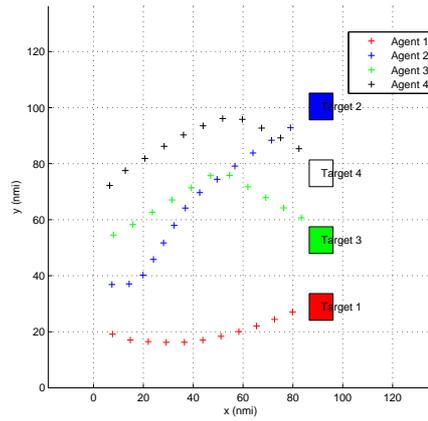
The schemes are implemented within two scenario types involving 3 and 4 aircraft. We consider wind velocity distributions drawn from two disturbance distributions; a bounded uniform distribution with $2\sigma_1$ -bounds, where $\sigma_1 = 0.48\text{m/s}$, and a Gaussian distribution with standard deviation σ_1 . For each MMPC scheme, results are obtained for 50 simulations for each of the disturbance types, and each configuration of aircraft.

In our formulation, aircraft are required to reach their respect target regions in finite time whilst maintaining safe separation. These target regions are located on the boundary of the interest area and determined in advance. The specific aircraft–target configurations considered are depicted in Figure 23, where closed loop trajectories obtained from application of single update robust MMPC are plotted.

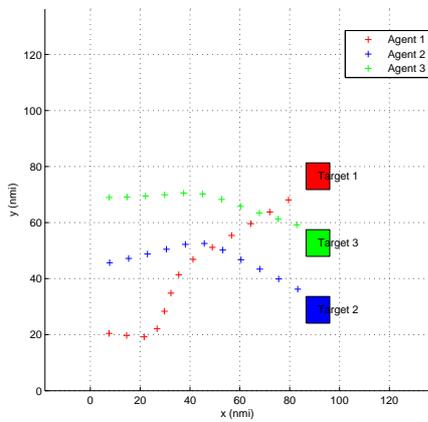
Tables 14 summarises the ensemble statistics obtained for each of the three schemes for an example fixed configurations of three aircraft. The means of the closed loop costs for each aircraft over the ensemble of 50 simulations for each disturbance setting are displayed for each scheme. The infeasibility column displays the proportion of simulations in the ensemble which were rendered infeasible, either for cases where the solver failed to find a feasible solution or when constraints were violated. Tables 15 and 16 present statistics on the proportion of stages in a simulation in which all aircraft execute parallel update in P-MMPC for fixed configurations of 3 and 4 aircraft. Recall that in P-MMPC, all aircraft optimise in parallel and if the coupling constraints i.e. the minimum separation constraints for the entire prediction horizon are satisfied, all aircraft update their policies in parallel. The results in Tables 15 and 16 show that for most of the time, aircraft update in parallel, and in some simulations, aircraft execute parallel update at every time step. This shows that when the schemes have been initialised with an optimal solution, the separation constraints along the prediction horizon at subsequent time steps are mostly not active. This can be attributed to the low levels of the stochastic component of the disturbances compared with the nominal speeds of the aircraft; for most of



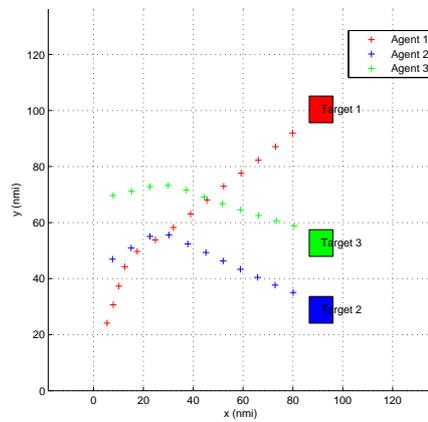
(a) 3 aircraft



(b) 4 aircraft



(c) 3 aircraft



(d) 3 aircraft

Figure 23: Sample trajectories obtained with single update robust MMPC. Crosses are plotted at 1 minute intervals, and the target regions are depicted as squares.

the time aircraft are following the initial optimal solution. This is supported by the observation that the closed loop costs obtained with the two MMPC schemes are identical to those associated with the single-shot disturbance rejection controller.

Table 14 summarises the results obtained for an example configuration. The closed loop costs associated with each aircraft are shown. The infeasibility column displays the proportion of simulations which were rendered infeasible, either for cases where the solver failed to find a feasible solution or when constraints were violated. The benefit of parallel updating is apparent at the higher disturbance levels, where the proportion of infeasibilities is almost half that of the MMPC1 scheme. The benefit is more apparent for the 4 agent case, which is to be expected as there are more constraints to be violated, and additionally, in MMPC1, the proportion of time where the agent applies disturbance feedback is higher; It can be seen that the closed loop costs obtained are identical to the single-shot disturbance rejection controller. This is unsurprising, as the stochastic component of the disturbances acting on the system is very low relative to the nominal speeds; consequently the initial solution fixes the subsequent optimisations, as the subsequent solutions merely track the initial solution. The benefits of parallel update are not apparent for the typical disturbance levels present. The closed loop cost is not a true indicator of performance as the measure only corresponds to those simulations that were feasible.

In practice of course, the regions of interest are not fixed, as the aircraft alert zones are time-varying. It is therefore important to address the issue of how to deal with interaction of neighbouring scenarios. We now examine by simulation the robustness of the MMPC methods developed to uncertainty in the form of unanticipated aircraft entering the conflict zone. When two groups of aircraft meet, the aircraft originally contained in the interest area are referred to as the ‘original’ group, with higher priority, whilst those joining the interest area are known as the ‘newcomers’. The trajectories of the newcomers are constrained by the plans of the original group. Although we do not detail here a method for identifying the original and new groups, heuristics for doing so can be imagined.

Figure 24 shows the results obtained with a scenario in which two new aircraft enter a region of interest. The original three aircraft involved in the initial encounter are initialised on the left side of the boundary of the region of interest and are required to reach their target regions located on the right hand side. They update their plans in the cyclic sequence $\{1, 2, 3\}$ using prediction horizons of length 30 mins and predicted time to collision roughly 20 minutes, unaware that a fourth and fifth aircraft flying East-West at 454 knots will enter the region of interest after 15 minutes. The new aircraft are not aware of the original two aircraft in the region of interest until the point at which they enter the region. Figure 24a shows the trajectories of the original aircraft up to, but not including time $k = 15$, displayed as

Disturbance	MMPC type											
	MMPC 1				P- MMPC				Single Shot			
	Closed Loop Cost			%	Closed Loop Cost			%	Closed Loop Cost			%
	Agent 1	Agent 2	Agent 3	Infeasible	Agent 1	Agent 2	Agent 3	Infeasible	Agent 1	Agent 2	Agent 3	Infeasible
Uniform $3\text{-}\sigma_1$	15.0	12.0	12.0	0	15.0	12.0	12.0	0	15.0	12.0	12.0	0
Gaussian σ_1	15.0	12.0	12.0	0	15.0	12.0	12.0	2.0	15.0	12.0	12.0	2.2

Table 14: Comparison of closed loop costs of 3 agents , configuration (23a)

Disturbance Type	Statistics summarising % of time steps with simultaneous agent update		
	Mean	Maximum	Minimum
Uniform $3\text{-}\sigma_1$	0.99	1	0.92
Gaussian σ_1	0.98	1	0.85

Table 15: Parallel Optimisation Variable Update Order (P-MMPC) 3 agents

Disturbance Type	Statistics summarising % of time steps with simultaneous agent update		
	Mean	Maximum	Minimum
Uniform $3\text{-}\sigma_1$	0.98	1	0.77
Gaussian σ_1	0.97	1	0.85

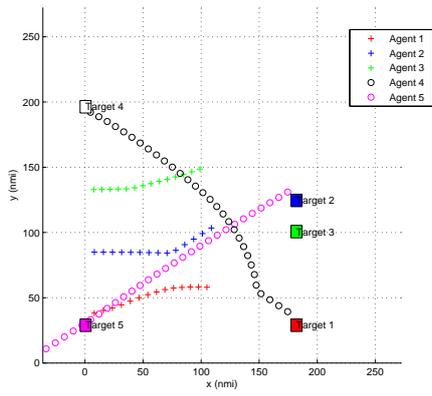
Table 16: Parallel Optimisation Variable Update Order (P-MMPC) 4 agents

crosses plotted at one minute intervals. The open loop predicted trajectories of the new aircraft when they appear at the boundary of the region are displayed as circles. Whilst these solutions satisfy the local and coupled collision avoidance constraints of the new aircraft, loss of separation with the original aircraft would occur without corrective action. The aircraft contained originally in the region of interest are termed 'high priority', as their predicted trajectories constrain those of the incoming 'low priority' aircraft.

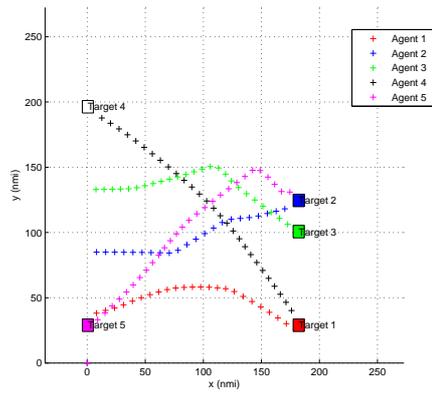
On entry to the region of interest, aircraft 4 optimises for a sequence of moves, its optimisation constrained by disturbance feedback perturbed predicted plans of all the other aircraft in the scenario. Aircraft 5 then does the same at the next timestep, and it can be seen from Figure 24b, which shows the closed loop trajectories of all aircraft, that it has to deviate from its originally planned trajectory to accommodate the plans of the original aircraft. Figure 24c shows the increase in the open loop cost of aircraft 5 incurred on solving the more constrained problem. Once a round of reupdating of the new aircraft has occurred, and feasible solutions are obtained, their open loop costs decrease monotonically and feasibility is retained when bounded disturbances are applied. Figure 24d displays the ground speeds of both sets of aircraft, and it can be seen the constraints on the speed are respected at all time.

The entry of the new aircraft constrains the actions of the original aircraft in subsequent optimisations, and increases the cycle length from 3 minutes to 5 minutes, and thus disturbance feedback is applied 80% of the time, compared to 67%.

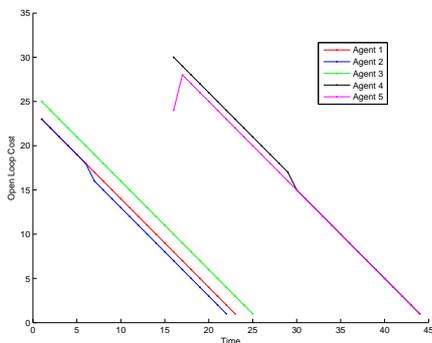
Having demonstrated the flexibility of the robust MMPC algorithm for handling scenarios in which unexpected aircraft enter on a simple example, we now turn our attention to handling more realistic settings with realistic



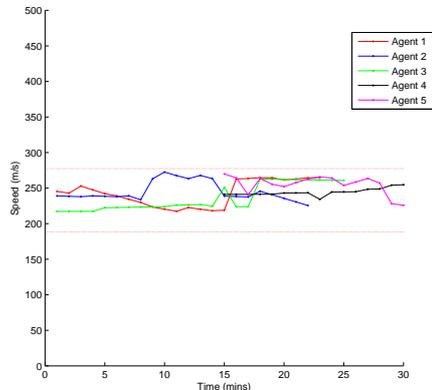
(a)



(b)



(c)



(d)

Figure 24: Interacting Scenarios: 2 aircraft entering a region of interest containing 3 aircraft after 10 minutes.

traffic data.

5.3 Application to Realistic Traffic Scenarios

Details of the realistic traffic scenario data have already been given in Section 2. For the evaluation of the robust MMPC algorithms in this section, we consider a subset of the data previously considered, specifically flights entering a region of area $200\text{nm}\times 200\text{nm}$ centred on Zurich over the 24 hour period. As with the flight data used in the previous sections, the entry points of flights which do not originate within the interest area are determined from extrapolating straight line trajectories from the departure airports. The centres of the exit target regions are similarly obtained.

With a view to keeping the sizes of the optimisation problems manageable, we consider only flights at level 330. Whilst in principle the entire 48 hours of data can be simulated in sequence, we split the data into 1 hour slots. The data is divided further into groups according to the flight heading angle quadrant; between North and East (N-E), with a heading angle of $0^\circ - 90^\circ$, E-S ($90 - 180^\circ$), S-W ($180 - 270^\circ$) and W-N ($270^\circ - 360^\circ$). This gives rise to a total of 192 scenarios. Some further preprocessing of the data is required to achieve feasible initial conditions since multiple aircraft were found to be starting from the same destination at the same time. In such cases, the flights were artificially separated by 80s, corresponding to a separation of 12nm.

Employing a fully distributed solution in which aircraft solve selfishly for their individual trajectories as they arrive without accounting for aircraft yet to appear may lead to infeasibilities, particularly in dense traffic scenarios. However, obtaining a global centralised solution for all aircraft every time new aircraft appear could potentially incur significant computational cost. We therefore propose dividing up the data to obtain smaller, computationally tractable problems and employ a formulation which is partially centralised and partially distributed. Specifically, the idea is to obtain joint solutions for cooperating groups of aircraft at predetermined time intervals, with fully distributed MMPC being executed in between joint updates. The cooperating set of aircraft includes those entering the scenario at the time of optimisation, and those predicted to enter within some sensing time horizon. The joint solution of the cooperating group of aircraft is constrained by the planned trajectories of those already present in the region of interest.

In our algorithm formulation, the prediction horizon used by each aircraft is required to be long enough for the target destination to be reached within the planning interval. The length is estimated by scaling the time corresponding to a constant cruising speed straight line trajectory by some factor greater than 1. When obtaining a joint solution for a cooperating group, the prediction horizons of the aircraft must be combined to accommodate possible interactions of aircraft with those yet to arrive. Consequently,

the inclusion of a new aircraft in the cooperating set potentially increases the prediction horizon used when solving for the group. Thus to prevent the size of the optimisation problem becoming prohibitively large, we restrict the look ahead for sensing new aircraft to be 5 minutes, whilst the planning horizons employed are in the order of tens of minutes. A cooperative joint solution is required to initialise the scheme. Subsequent cooperative joint optimisations are solved on arrival of aircraft for whom trajectories have not been obtained at the previous joint optimisation time. As the aircraft sensing interval is of length 5 minutes, these cooperative joint solutions are obtained at intervals of at least 5 minutes. The fully distributed MMPC optimisations take place every minute. The cycle time for the distributed optimisations increases by one on the addition of every new aircraft to the scenario. A full description of the algorithm is detailed in Algorithm 4 in Appendix B.

We require the following assumptions for our problem formulation.:

Assumption 5.1 *Aircraft are aware of the presence of all other aircraft which are predicted to enter their region of interest considered within some look-ahead interval, chosen to be 5 minutes in this work. Additionally, they have access to their current states.*

It is possible for aircraft in a cooperating group to still be present in the scenario on arrival of aircraft from the next cooperating group. Consequently, we require the use of Assumption 5.2 to ensure that the robust feasibility and finite time completion guarantees hold.

Assumption 5.2 *On entry of unanticipated incoming aircraft to the region of interest, a feasible solution for all aircraft is available without the need for reoptimisation of the trajectories of the original aircraft, who can adopt candidate feasible plans based on their predicted plans made prior to the entry of the new aircraft.*

Assumption 5.3 *The times of entry of aircraft to the region of interest match the predictions exactly.*

Whilst this assumption may appear to be overly restrictive, should this assumption not be met, a number of possibilities for conflict resolution exist; in the event that aircraft appear at times differing from those predicted, the arriving aircraft must solve for their trajectories constraining their respective optimisation problems by the plans of aircraft already present and those of aircraft predicted to appear. If a feasible solution cannot be found this way, a joint solution must be obtained for the entire group.

We present some of the results obtained next. We summarise results obtained from scenarios with a minimum of 3 concurrent aircraft in Table 17. The open loop column contains results obtained in the absence of any

Minimum Sep (nm)	Open Loop	Closed Loop
Mean	1.70	23.5
Maximum	32.7	46.4
Minimum	0.75	16.7
Completion Rate (%)		
Mean	27.2	100
Maximum	80.0	100
Minimum	12.5	100

Table 17: Summary of statistics of all 40 feasible simulations with at least 3 concurrent aircraft in scenario

control action, with disturbances present. The aircraft are initialised with headings and speeds obtained from the data set. The closed loop column displays results obtained on application of single aircraft update robust MMPC, with an identical disturbance realisation acting on the system. Statistics on the minimum separation obtained per scenario and the proportion of aircraft reaching their target regions are displayed. It can be seen that in the absence of control action, aircraft fail to reach their target regions within the times taken by the aircraft executing MMPC. Moreover, the minimum separation constraint of 5 nm is violated. As the aircraft trajectories comprise discrete sequences of positions, an offset corresponding to the largest distance that can be travelled in a discretisation time step is added; the nominal minimum separation distance was increased from 5 to 12.5nm. The minimum separations obtained with MMPC are in excess of this, indicating the conservatism of the constraint tightening employed.

We next present results from individual scenarios including those identified as being the most challenging scenarios for which it was possible to obtain results, and scenarios with flight routes in each of the 4 directions. The direction of flight route information and time slot corresponding to each scenario is presented in Table 18.

Table 19 summarises results obtained on each of the six scenarios selected. The performance metrics are obtained for each of the scenarios with closed loop control (robust MMPC) and compared with those obtained with open loop control (no control action) and as previously, identical disturbance realisations act on the system. To give an indication of the sizes of the problems, we show the maximum prediction horizons employed and the maximum number of aircraft in a cooperating set. The maximum number of aircraft for which it was possible to obtain a joint solution was found to be around 5, with a corresponding maximal prediction horizon of around 50 minutes (with a sampling time of 1 minute). The numbers of cooperating sets in each scenario, and hence the number of joint optimisations performed

Scenario	Flight Heading Direction	Time Slot
(i)	N-E	1267-1327 mins
(ii)	N-E	1322-1382 mins
(iii)	N-E	1444-1504 mins
(iv)	W-N	845-905 mins
(v)	S-W	1264-1324 mins
(vi)	E-S	1094-1154 mins

Table 18: Scenario Description: The second column gives the route direction, and the third column shows the time slots corresponding to each scenario, in minutes after midnight.

are given in the final row.

Figure 25 shows the evolution of the number of aircraft present in the region of interest over time for the selected scenarios (i)–(vi). It can be seen that scenarios with up to 11 concurrent aircraft can be handled.

The distributions of minimum separation distances in each of the scenarios are shown in Figure 26. In each of the six cases, the minimum separation distance is in excess of the 5 nm requirement.

Sample trajectories obtained in the selected scenarios for selected time windows are plotted in Figure 27.

5.3.1 Summary

We have demonstrated the capability of distributed robust MMPC in performing conflict resolution in mid term air traffic scenarios. Results have been obtained on both simple scenarios. We have adapted the algorithms for interacting scenarios by periodically obtaining joint solutions for cooperating sets of aircraft and employing a heuristic for determining the aircraft included in the optimisation problems. Solutions have been obtained for scenarios with prediction horizons of up to around 50 minutes, where the cooperating groups in the joint optimisations have up to about 5 aircraft. Whilst results have not been obtained with the predicted 2035 traffic levels, the algorithm has successfully been executed with densities of up to 11 concurrent aircraft in the scenario. Despite the fewer formal guarantees associated with the original distributed robust MMPC algorithm, it is observed in simulation that the minimum separation constraints are respected in the optimisations that are feasible. The minimum separations obtained are in excess of the safe separation bound of 5nm, even with the inclusion of an offset to allow for discretisation effects, indicating conservatism of the constraint tightening employed in the robust MMPC.

	Scenario (i)		Scenario (ii)		Scenario (iii)		Scenario (iv)		Scenario (v)		Scenario (vi)	
	O/L	C/L	O/L	C/L	O/L	C/L	O/L	C/L	O/L	C/L	O/L	C/L
Minimum Separation (nm)	0.75	17.0	0.75	19.6	0.75	17.6	0.75	32.1	0.75	16.7	0.75	17.7
Completion Rate (%)	27.2	100	33.3	100	25.9	100	33.3	100	41.7	100	80	100
Total number of aircraft	11		12		27		6		24		15	
Maximum no. of concurrent aircraft	–	5	–	5	–	11	–	3	–	10	–	4
Maximum prediction horizon (mins)	–	50	–	50	–	52	–	46	–	50	–	54
Maximum size of cooperating set	–	4	–	2	–	5	–	2	–	5	–	4
Number of cooperating sets	–	6	–	7	–	8	–	5	–	6	–	6

Table 19: Summary of results on individual scenarios (i)–(vi).

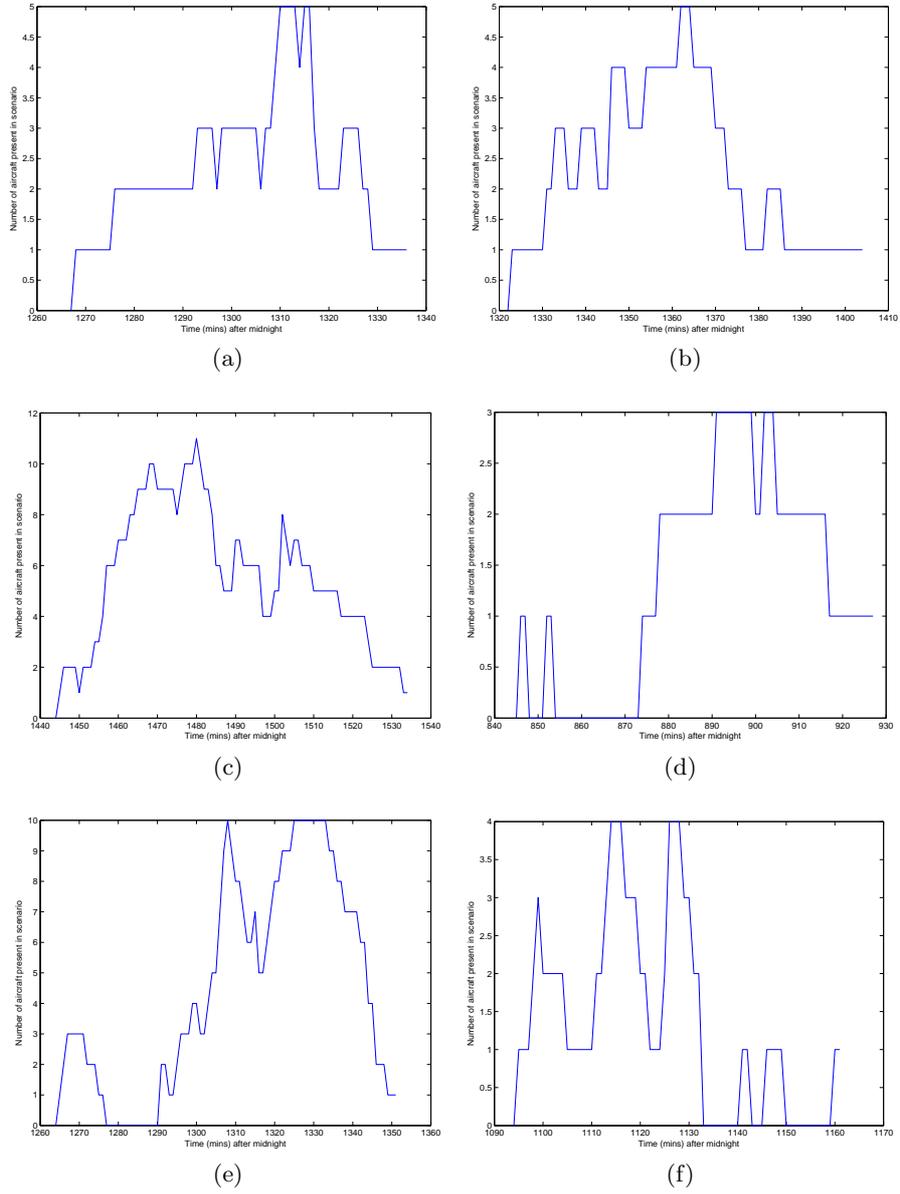


Figure 25: Variation in number of aircraft present in scenario over time for scenarios (i)-(vi).

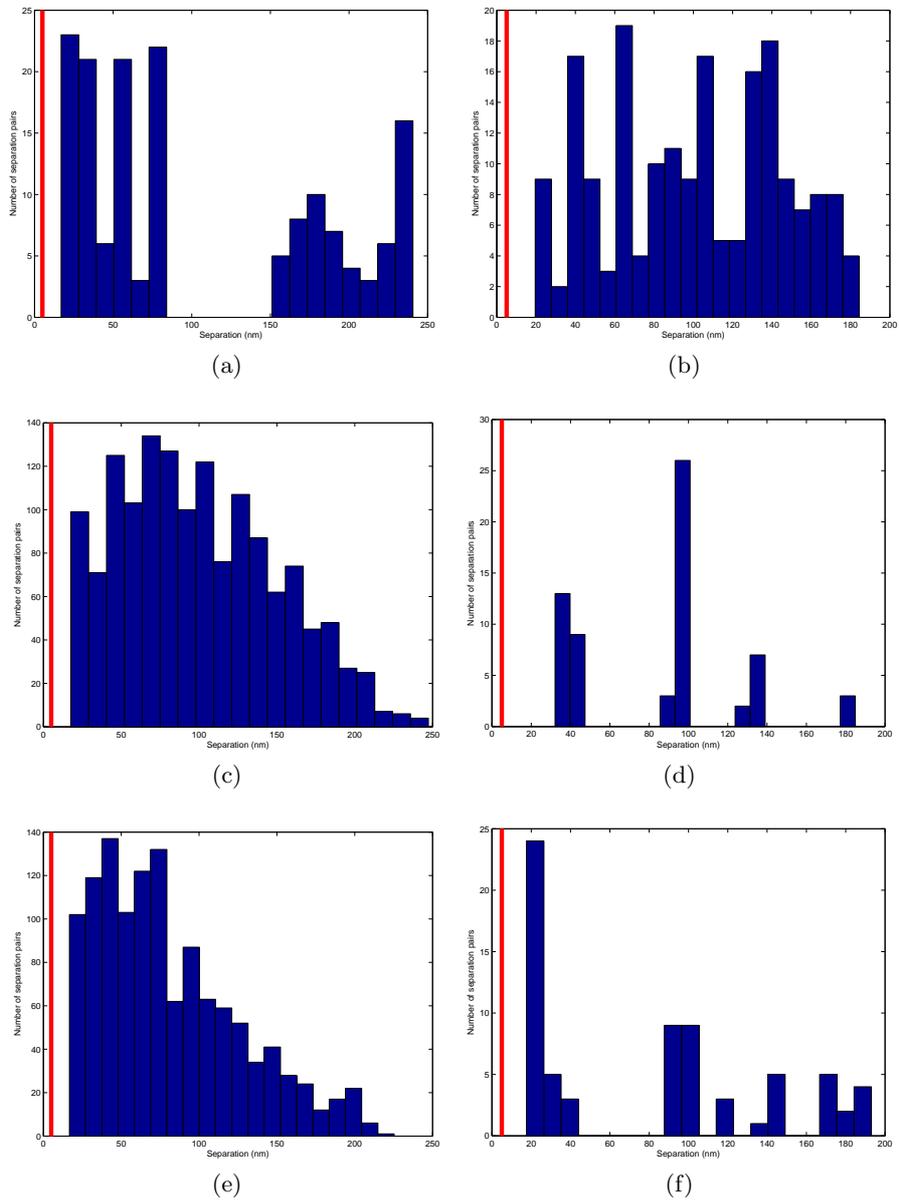


Figure 26: Distribution of minimum separation obtained for scenarios (i)-(vi). The minimum separation bound is shown in red.

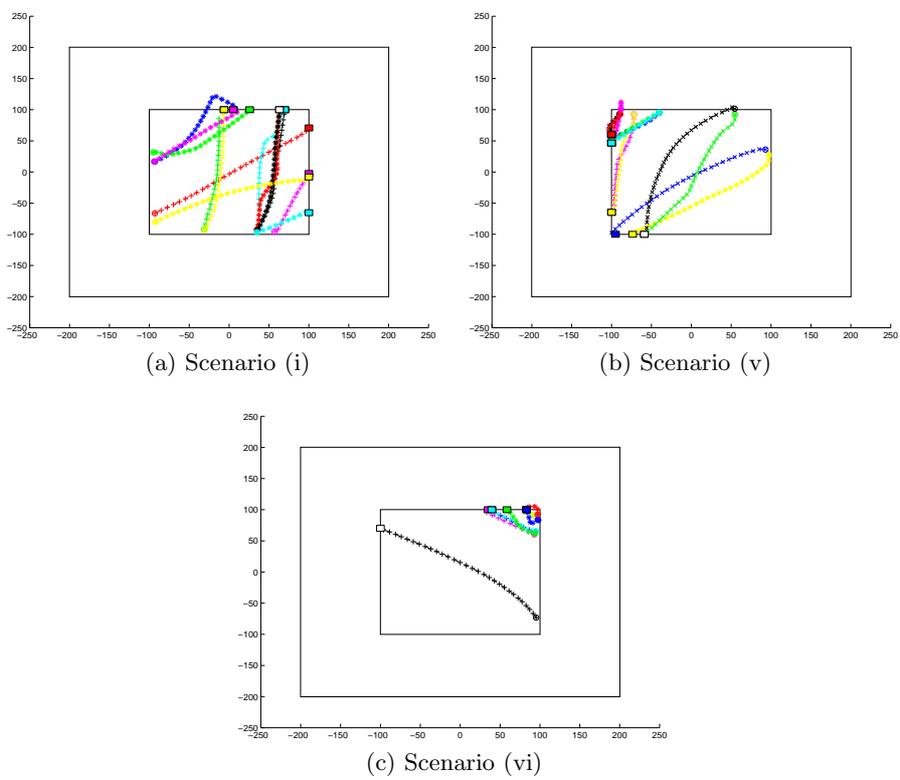


Figure 27: Aircraft trajectories from selected time windows from scenarios (i),(v) and (vi) .

6 Concluding remarks

After a careful review of available CD&R methods carried out in deliverable D5.1 [5], the most relevant methods for the A³ concept were identified in D5.2 [11]. Those methods were chosen based on their performance and their potential for use in the iFly project. They were further developed within WP5 and one algorithm for Short-term CD&R as well as two candidate algorithms for Mid-term CD&R have been validated in this deliverable. The finalised algorithms have been presented in detail in [6], along with some initial simulation results on test scenarios. In this report the main validation results for the proposed CD&R solutions are presented. In order to assess the performance of the algorithms, air traffic data from a realistic traffic sample have been used.

For the Short-term level, Decentralised Navigation Functions (NFs) have been used to handle CD&R in a fast real-time feedback manner. The simulation results show that the algorithm resolved all conflicts while being computationally efficient for real-time application and requiring little increase in the overall flown distance, despite the fact that no optimization was performed.

For the Mid-term level two different approaches based on the MPC methodology have been developed and used in the simulations. The first one, prioritized MPC has been developed based on the priority requirements of the ConOps, defined in D1.3 [7], while the second one aims to provide formal guarantees for the completion time of the aircraft flight plans, utilizing disturbance feedback. They have both been tested in simple test scenarios, in order to tune their parameters and then, have been shown to provide conflict-free trajectories when tested with realistic traffic data. In the case of prioritized MPC, a more demanding traffic has been simulated, ignoring the effects of uncertainty to match with the computational capabilities available at the time of the simulations. In the second alternative, a smaller part of the traffic sample was tested, taking uncertainties into consideration and providing formal guarantees against uncertainty for the traffic sample. Both algorithms have demonstrated a potential to operate under high traffic densities. The results clearly demonstrate a big potential for their use in autonomous aircraft situations. Unsolved problems that previously developed methods in literature faced, like identifying the conflicting situations efficiently and dynamically clustering the airspace to reduce computational complexity and handle high traffic densities have been resolved in this work package. Furthermore, a systematic way to address the issue of priorities that previously did not exist in literature has been developed.

For each algorithm suitable metrics have been used to evaluate the results, according to the characteristics and goals that are inherent to each approach. Although not all algorithms use the same models, making a direct comparison of the methods, the overall results presented in this concluding

report of WP5 indicate that there is promising potential in the concept of decentralised CD&R, both for the Short-term and Mid-term levels. The fact that the two levels operate independently suggests that in an integrated CD&R system the overall benefits can be further improved, exploiting the performance attained by the MPC supported by the formal guarantees of NFs for those few conflicts that escape the Mid-term algorithm. An initial study to examine and exploit the use of algorithms running in different time scales at the same time (Mid and Short term) has been carried out in D5.3 [6].

The ability of the algorithms developed in WP5 to cope with the realistic traffic data suggests that decentralisation of SA in future ATM designs is a viable option to the centralised approach used today and should be exploited further. Issues that have been raised because of the computational limits of available equipment at the time of the simulations is unlikely to exist at the time of the application of the algorithms using future computational power.

Directions for further development of NF-based aircraft CD&R have been suggested in Section 3.3. Research on the prioritized MPC algorithm in the future can focus on several different aspects. First, a 3D algorithm in order to extend its capability to complete 3D maneuvers should be investigated to reduce fuel consumption. Furthermore, the issue of optimally assigning aircraft to different flight levels has not been studied in detail. Finally, exploration towards obtaining better decentralization schemes, taking into account communication delays should be carried out in the future.

A Prioritized Conflict Resolution

Consider I aircraft flying within an area of interest. The continuous-time dynamics for level flight cruise, using are:

$$\dot{Q}_i = \begin{bmatrix} \dot{X}_i \\ \dot{Y}_i \\ \dot{V}_i \\ \dot{\psi}_i \\ \dot{m}_i \end{bmatrix} = \begin{bmatrix} V_i \cos(\psi_i) + W_{i1} \\ V_i \sin(\psi_i) + W_{i2} \\ -\frac{C_{D_i} S_i \rho V_i^2}{2 m_i} + \frac{1}{m_i} T_i \\ \frac{C_{L_i} S_i \rho V_i}{2 m_i} \sin(\phi_i) \\ -\eta_i T_i \end{bmatrix} \quad (1)$$

for $i \in \mathcal{I} \triangleq \{1, \dots, I\}$. In this equation, X_i, Y_i denote the horizontal position of the aircraft, V_i the airspeed, ψ_i the heading angle, m_i the mass, W_{i1}, W_{i2} the wind velocity, C_{D_i}, C_{L_i} are aerodynamic lift and drag coefficients, S_i denotes the surface of the wings, ρ is the air density, η_i is a parameter that represents the rate at which fuel is consumed, T_i is the engine thrust and ϕ_i the bank angle.

The Conflict Resolution (CR) problem can then be described as the optimal control problem that determines the optimal (corresponding to some desired cost function) inputs T_i, ϕ_i for all aircraft $i \in \mathcal{I}$ such that they respect the following two sets of constraints:

- *Velocity and acceleration constraints.* Aerodynamic reasons impose some physical constraints on the minimum and maximum TAS an aircraft can fly at each altitude. Furthermore, passenger comfort, as well as other human factors reasons impose constraints on the acceleration and the turning rate. Those constraints can be expressed in the following form:

$$\begin{aligned} v_{\max} &\geq V_i(t) \geq v_{\min} \\ \dot{V}_i(t) &\leq \delta V \\ \dot{\psi}_i(t) &\leq \delta \psi \end{aligned} \quad (2)$$

- *Conflict avoidance constraints.* All aircraft should remain separated at all times, by at least a minimum distance Δ , which is typically set to $\Delta = 5\text{nm}$ for cruising altitudes. Those constraints can be encoded as:

$$\left\| \begin{bmatrix} X_i \\ Y_i \end{bmatrix} (t) - \begin{bmatrix} X_j \\ Y_j \end{bmatrix} (t) \right\|_2 \geq \Delta \quad (3)$$

for all times $t \geq 0$ and for all aircraft pairs $i \neq j$, where $(i, j) \in \mathcal{I}$.

- *Priority constraints.* Given a bijective priority function $s : \mathcal{I} \rightarrow \mathcal{I}$, each aircraft $i \in \mathcal{I}$ is assigned a priority $s(i)$. Since the function $s(\cdot)$ is bijective, the priority ordering is unique. The priority concept used enforces the constraint that aircraft i will maneuver if (and only if)

all aircraft $j : s(j) < s(i)$ cannot satisfy constraints (3), (2) without aircraft i maneuvering.

The problem described cannot be modeled as a tractable optimization problem, irrespective of the cost function chosen due to the complexity of the nonlinear dynamic constraint (1). In order to be able to resolve it, a series of approximations that allow us to formulate the problem as a MILP is developed. Even though the resulting optimization problem is still NP-hard, instances of realistic size in the ATM context are readily solvable by commercially available computational tools.

A.1 Centralized Formulation

The problem is modeled as a two-level hierarchical algorithm. At the highest level, a centralized MPC problem with simplified dynamics is solved, taking into account all constraints and generating an optimal set of inputs for each aircraft over a certain prediction horizon N . Once the optimal input sequences have been generated for all aircraft, they are pushed down to the lower level in the hierarchy, namely the Flight Management System (FMS). The FMS generates the appropriate inputs to be applied through the autopilot on the actual aircraft dynamics. The optimization problem is then resolved periodically and applied in a receding horizon fashion. This setup is illustrated in Figure 28.

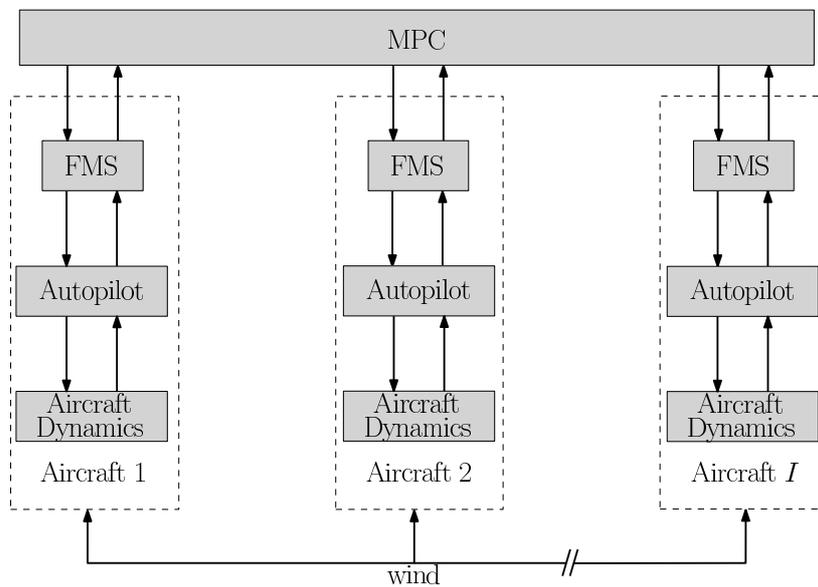


Figure 28: Hierarchical Multi-Level System

Subsequently, a step by step construction of the problem will be detailed.

A.1.1 Dynamical constraints

The dynamics (1) are abstracted to a linear discrete-time model, based on single integrator dynamics, as follows:

$$p_i(t+1) = p_i(t) + hu_i(t) + hw_i(t), \quad (4)$$

where $p_i(t) \triangleq [x_i(t) \ y_i(t)]^T$ denotes the aircraft position, $w_i(t) = [w_i^x(t) \ w_i^y(t)]^T$ denotes the wind uncertainty, $u_i(t) = [u_i^x(t) \ u_i^y(t)]^T$ is the velocity input, h is the sampling period (lower case letters are used to distinguish between the nonlinear and the simplified dynamics) and $t \in \{0, \dots, N-1\}$. Furthermore, the nominal dynamics, computed using an ideal straight flight at nominal speed from the current point to the destination p_i^d in the absence of wind are denoted by

$$\bar{p}_i(t+1) = \bar{p}_i(t) + hu_i^{\text{nom}}(t), \quad (5)$$

where $u_i^{\text{nom}}(t) = \begin{bmatrix} V_i^{\text{nom}} \cos \psi_i^{\text{nom}} \\ V_i^{\text{nom}} \sin \psi_i^{\text{nom}} \end{bmatrix}$.

A.1.2 Velocity and acceleration constraints

Another important factor to consider is the dynamical constraints that aircraft have on their airspeed, their acceleration, as well as their turning rate. Using the simplified dynamics (4), the constraints (2) are approximated by:

$$\begin{aligned} [\cos \psi_i^{\text{nom}} \ \sin \psi_i^{\text{nom}}] u_i(t) &\geq u_{\min} \\ \|u_i(t)\|_{\infty} &\leq u_{\max} \\ \|u_i(t) - u_i(t-1)\|_{\infty} &\leq \delta u \end{aligned} \quad (6)$$

for all $t \in \{0, \dots, N-1\}$, where the convention $u_i(-1) = [V_i^{\text{cur}} \cos \psi_i^{\text{cur}} \ V_i^{\text{cur}} \sin \psi_i^{\text{cur}}]^T$ is used, with V_i^{cur} and ψ_i^{cur} denoting the current speed and heading of the aircraft. The lower constraint on the speed is thus, conservatively approximated by a linear constraint, while the upper one is relaxed into an outer ∞ -norm constraint. Similarly, the 2-norm acceleration as well as the turning rate constraint, are both substituted by a single ∞ -norm one. The resulting constraints are linear, simplifying the optimization process. Alternatively, the lower constraint could be implemented with a less restrictive 1-norm, as in [4], where also 4 binary variables together with a linear equality would have to be introduced. The norm constraints can be further refined using polytopic norms at the expense of the increase of the number of resulting constraints, and thus, more computational effort.

A.1.3 Cost and priority constraints

As discussed, not all aircraft are considered to have equal priorities. Inspired by the concept of operations proposed by the iFly project [7], higher priority aircraft will not deviate from their flight plan, unless all lower priority aircraft cannot resolve the conflict without them deviating. Recall that each aircraft $i \in \mathcal{I}$ is assigned a priority $s(i)$, that takes values in $\{1, \dots, I\}$; the aircraft with priority $s(i) = I$ has the highest priority. Guided by the setup in [12], define I binary variables $\delta_1, \dots, \delta_I$, one for each aircraft. Given the nominal input sequence $u_i^{\text{nom}}(t)$ for $t \in \{0, \dots, N-1\}$, define the following set of *deviation* constraints corresponding to the i -th aircraft:

$$\begin{aligned} \|u_i(t) - u_i^{\text{nom}}(t)\|_\infty &\leq \epsilon_i(t) \\ 0 &\leq \epsilon_i(t) \leq M_\epsilon \delta_i, \end{aligned} \quad (7)$$

for all $t \in \{0, \dots, N-1\}$ where M_ϵ is a finite constant. The constraint (7) penalizes any deviation from the trajectory generated by (5) due to the designed control inputs away from the nominal control input sequence. Given the optimization horizon N , the cost is defined as:

$$\mathcal{J} = \underbrace{\sum_{i=1}^I \left\| \begin{bmatrix} \epsilon_i(0) & \epsilon_i(1) & \dots & \epsilon_i(N-1) \end{bmatrix} \right\|_1}_{\text{deviation from nominal}} + \beta \underbrace{\sum_{i=1}^I 2^{s(i)-1} \delta_i}_{\text{priorities}}, \quad (8)$$

where β is a positive scalar given by $\beta = (I-1)NM_\epsilon + 1$. This choice of β ensures that the priorities part of the cost dominates the deviation from nominal part. Thus, if a deviation occurs, the binary variable δ_i is set to one and results in a higher cost. Moreover, given the specific structure of weighting, the various binary variables ensure that the satisfaction of higher priority constraints always results in a lower cost than any possible combination of the lower priority constraints. In the air-traffic problem, this would mean that a higher priority aircraft will deviate from its nominal flight plan only if all other aircraft with lower priority cannot resolve the conflict.

A.1.4 Separation constraints

It is important that the formulation is robust against wind, so that conflicts do not occur even in the case of strong winds. In general, wind is normally distributed and thus, has unbounded support, making a robust formulation against all possible noise realizations impossible. Instead, a confidence interval of the noise will be chosen against which the problem will be made robust. A “safe” choice would be to choose the 99.7% confidence interval, which corresponds to making the formulation robust against all winds that lie within the $[-3\sigma, 3\sigma]$ interval, where σ is the standard deviation of the

wind used in the model. This can be done using existing tools and approaches for robust programming, like [16] that is implemented in YALMIP [14].

In order to maintain separation between aircraft, constraints of the following form have to be enforced:

$$\|p_i(t) - p_j(t)\|_2 \geq \Delta, \quad (9)$$

for all $t \in \{0, \dots, N\}$, $w_i^x(t), w_i^y(t), w_j^x(t), w_j^y(t) \in [-3\sigma, 3\sigma]$ and $i, j \in \mathcal{I}$ with $i \neq j$. The constraint described by (9) is not convex and, in order to be able to handle it computationally, it is tightened, using the norm inequality $\|\cdot\|_2 \geq \|\cdot\|_\infty$ to:

$$\|p_i(t) - p_j(t)\|_\infty \geq \Delta. \quad (10)$$

Then, using the so-called big-M technique (similar to the formulation in [18]) the equation (10) can be written as:

$$\begin{aligned} x_i(t) - x_j(t) &\geq \Delta - M d_{i,j}^1(t) \\ x_i(t) - x_j(t) &\leq -\Delta + M d_{i,j}^2(t) \\ y_i(t) - y_j(t) &\geq \Delta - M d_{i,j}^3(t) \\ y_i(t) - y_j(t) &\leq -\Delta + M d_{i,j}^4(t) \end{aligned} \quad (11)$$

$$\sum_{\nu=1}^4 d_{i,j}^\nu(t) \leq 3, \quad d_{i,j}^\nu(t) \in \{0, 1\}$$

for all $t \in \{0, \dots, N\}$, $w_i^x(t), w_i^y(t), w_j^x(t), w_j^y(t) \in [-3\sigma, 3\sigma]$ and $i, j \in \mathcal{I}$ with $i \neq j$, where M is a sufficiently large number. The last constraint ensures that at least one of the inequality constraints is active, and consequently that the two aircraft are separated by the required distance along at least one of the axes.

Unlike the original constraints (3), constraints (12) are not enforced at all times, but only at sampled instances. In order to maintain the separation even between those instances, it has to be ensured that if some of the binaries $d_{i,j}^\nu(t)$ and $d_{i,j}^\nu(t+1)$ are different, no conflict has taken place in the meantime. This can either be ensured by enforcing the constraints more often (at the expense of introducing more variables) or by not allowing $d_{i,j}^\nu(t+1)$ to be different from $d_{i,j}^\nu(t)$ in a way that the maintenance of separation in the meantime is compromised (at the expense of conservativeness).

A.1.5 Constraint relaxation

Because of the model mismatch between the actual nonlinear model and the simplified linear one, as well as because of the fact that the tails of the wind error distribution are ignored, it is likely that aircraft find themselves in a situation where a feasible solution no longer exists. It is nevertheless

important that even in such a situation, the algorithm provides aircraft with a solution, in order to avoid an actual collision. In such a configuration, aircraft should try to return as soon as possible to a feasible configuration, rather than continuing to fly in a potentially unsafe configuration optimizing over usual criteria.

This can in fact be dealt with systematically, though the introduction of $N + 1$ more binary variables, $\delta_r(t), t \in \{0, \dots, N\}$, in a similar fashion to the priority binaries, but incurring a higher cost and allowing their actuation only to incur when otherwise the situation would remain infeasible. In this case, the separation constraints (12) become:

$$\begin{aligned}
x_i(t) - x_j(t) &\geq \Delta - r_{ij}(t) - Md_{i,j}^1(t) \\
x_i(t) - x_j(t) &\leq -\Delta + r_{ij}(t) + Md_{i,j}^2(t) \\
y_i(t) - y_j(t) &\geq \Delta - r_{ij}(t) - Md_{i,j}^3(t) \\
y_i(t) - y_j(t) &\leq -\Delta + r_{ij}(t) + Md_{i,j}^4(t) \\
\sum_{\nu=1}^4 d_{i,j}^\nu(t) &\leq 3, \quad d_{i,j}^\nu(t) \in \{0, 1\} \\
0 \leq r_{ij}(t) &\leq M_r \delta_r(t), \quad \delta_r(t) \in \{0, 1\}
\end{aligned} \tag{12}$$

for all $t \in \{0, \dots, N\}$, $w_i^x(t), w_i^y(t), w_j^x(t), w_j^y(t) \in [-3\sigma, 3\sigma]$ and $i, j \in \mathcal{I}$ with $i \neq j$, where M_r is the maximum allowed constraint relaxation.

Using this approach, the cost (8) is then modified to:

$$\begin{aligned}
\mathcal{J} = &\underbrace{\sum_{i=1}^I \|\left[\epsilon_i(0) \quad \epsilon_i(1) \quad \dots \quad \epsilon_i(N-1) \right]\|_1}_{\text{deviation from nominal}} + \underbrace{\beta \sum_{i=1}^I 2^{s(i)-1} \delta_i}_{\text{priorities}} + \\
&\underbrace{\beta_r \sum_{t=0}^N \sum_{i=1}^I \sum_{j=i+1}^I r_{ij}(t) + \beta_{\delta_r} \sum_{t=0}^N 2^{I+N-t} \delta_r(t)}_{\text{constraint relaxation}},
\end{aligned} \tag{13}$$

where β_r is a positive scalar that weighs the amount by which the constraints are relaxed and β_{δ_r} is chosen such that the relaxation of constraints part of the cost dominates the deviation from nominal and the priorities part. Furthermore, if the constraints are to be relaxed, this happens again in a prioritized fashion, first allowing constraints at the end of the horizon to be relaxed (as it is highly likely that what is considered as worst-case by the algorithm will not arise) and the ones at the beginning are the last to relax.

A.1.6 Wind correlation modeling

Wind uncertainty modeled in the simulator is correlated in time and space. It would be important to examine whether this structure is important in

this formulation. As aircraft that are likely to involve in a conflict fly close to each other, it is expected that using the spatial correlation structure of the wind, the algorithm may produce less conservative resolutions, allowing better use of the available airspace.

The spatial correlation of the wind field on the horizontal plane can be described by the following equation:

$$\rho_{xy}(\|p_i(t) - p_j(t)\|_2) = -0.006 + 1.006e^{-\frac{\|p_i(t) - p_j(t)\|_2}{337000}}, \quad (14)$$

where $\|p_i(t) - p_j(t)\|_2$ is the horizontal separation in meters between the two aircraft i, j . Utilizing (14), the differences in the wind speeds experienced by the two aircraft along the two axes in (21) can be expressed as normally distributed variables:

$$\begin{aligned} (w_i^x(t) - w_j^x(t)) &\sim N(0, \sigma_{i,j}(t)) \\ (w_i^y(t) - w_j^y(t)) &\sim N(0, \sigma_{i,j}(t)), \end{aligned} \quad (15)$$

where

$$\sigma_{i,j}(t) = \sigma \sqrt{2 - 2\rho_{xy}(\|p_i(t) - p_j(t)\|_2)}. \quad (16)$$

Since $\sigma_{i,j}(t)$ depends in a non-convex fashion on the inputs via the dynamics (4), using this exact constraint is not possible. Instead, we use

$$\bar{\sigma}_{i,j}(t) = \sigma \sqrt{2 - 2\rho_{xy}(\|\bar{p}_i(t) - \bar{p}_j(t)\|_2 + ht\delta v)}, \quad (17)$$

where $\|\bar{p}_i(t) - \bar{p}_j(t)\|_2$ is the distance that the aircraft would have, had followed their nominal flight plans, and δv is a constant related to the maximum change of the airspeed magnitude at each step. In a similar fashion as before, assuming that the differences in the wind speeds lie in the 99.7% confidence interval, and using (17), the constraints (10) can be rewritten as:

$$\left\| p_i(0) - p_j(0) + h \sum_{\tau=0}^{t-1} (u_i(\tau) - u_j(\tau) + w_i(\tau) - w_j(\tau)) \right\|_{\infty} \geq \Delta \quad (18)$$

which, using the triangle inequality becomes:

$$\left\| p_i(0) - p_j(0) + h \sum_{\tau=0}^{t-1} (u_i(\tau) - u_j(\tau)) \right\|_{\infty} \geq \Delta + h \left\| \sum_{\tau=0}^{t-1} (w_i(\tau) - w_j(\tau)) \right\|_{\infty} \quad (19)$$

which is then approximated by

$$\left\| p_i(0) - p_j(0) + h \sum_{\tau=0}^{t-1} (u_i(\tau) - u_j(\tau)) \right\|_{\infty} \geq \Delta + h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau). \quad (20)$$

Finally, coming back to (12), the constraints can be rewritten as:

$$\begin{aligned}
x_i(0) - x_j(0) + h \sum_{\tau=0}^{t-1} (u_i^x(\tau) - u_j^x(\tau)) &\geq \Delta + h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) - r_{ij}(t) - Md_{i,j}^1(t) \\
x_i(0) - x_j(0) + h \sum_{\tau=0}^{t-1} (u_i^x(\tau) - u_j^x(\tau)) &\leq -\Delta - h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) + r_{ij}(t) + Md_{i,j}^2(t) \\
y_i(0) - y_j(0) + h \sum_{\tau=0}^{t-1} (u_i^y(\tau) - u_j^y(\tau)) &\geq \Delta + h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) - r_{ij}(t) - Md_{i,j}^3(t) \\
y_i(0) - y_j(0) + h \sum_{\tau=0}^{t-1} (u_i^y(\tau) - u_j^y(\tau)) &\leq -\Delta - h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) + r_{ij}(t) + Md_{i,j}^4(t) \\
\sum_{\nu=1}^4 d_{i,j}^\nu(t) &\leq 3, \quad d_{i,j}^\nu(t) \in \{0, 1\} \\
0 \leq r_{ij}(t) &\leq M_r \delta_r(t), \quad \delta_r(t) \in \{0, 1\},
\end{aligned} \tag{21}$$

for all $t \in \{0, \dots, N\}$ and $i, j \in \mathcal{I}$ with $i \neq j$.

A.1.7 Enforcement of inter-sample constraints

To make sure that the separation is not violated, the separation constraints have to be enforced at least as often as the time needed for the difference on the coordinates on one axis to go from $-\Delta$ to Δ . The fastest this could happen occurs when aircraft are flying on opposite directions and the time needed for this to happen depends on their maximum allowed speed. Assuming a maximum allowed speed of 300m/s, which covers all aircraft that will be considered in this study, one can deduce that the constraints should be enforced at least every around 30sec. This is rather small to be used as a time step for the MPC approach (which is usually around 3-5mins), and thus, additional constraints have to be introduced and enforced. This is done by taking between any two time samples t and $t+1$ L further samples at times $\{t + \frac{1}{L}, t + \frac{2}{L}, \dots, t + \frac{L-1}{L}\}$ on which the separation constraints are applied, i.e. the satisfaction of the following set of constraints is required,

additionally to (12):

$$\begin{aligned}
(x_i(t) - x_j(t)) + \frac{hl}{L} (u_i^x(t) - u_j^x(t) + w_i^x(t) - w_j^x(t)) &\geq \Delta - r_{ij}(t) - Md_{i,j}^1(t, l) \\
(x_i(t) - x_j(t)) + \frac{hl}{L} (u_i^x(t) - u_j^x(t) + w_i^x(t) - w_j^x(t)) &\leq -\Delta + r_{ij}(t) + Md_{i,j}^2(t, l) \\
(y_i(t) - y_j(t)) + \frac{hl}{L} (u_i^y(t) - u_j^y(t) + w_i^y(t) - w_j^y(t)) &\geq \Delta - r_{ij}(t) - Md_{i,j}^3(t, l) \\
(y_i(t) - y_j(t)) + \frac{hl}{L} (u_i^y(t) - u_j^y(t) + w_i^y(t) - w_j^y(t)) &\leq -\Delta + r_{ij}(t) + Md_{i,j}^4(t, l) \\
\sum_{\nu=1}^4 d_{i,j}^\nu(t, l) &\leq 3, \quad d_{i,j}^\nu(t, l) \in \{0, 1\} \\
0 \leq r_{ij}(t) \leq M_r \delta_r(t), \quad \delta_r(t) &\in \{0, 1\}
\end{aligned} \tag{22}$$

for all $t \in \{0, \dots, N-1\}$, $w_i^x(t), w_i^y(t), w_j^x(t), w_j^y(t) \in [-3\sigma, 3\sigma]$, $i, j \in \mathcal{I}$ with $i \neq j$ and $l \in \{1, \dots, L-1\}$. As it can be understood by the formulation, in such a setup the number of binaries needed to enforce the separation are L times more than before.

Similarly, taking into account the correlation as described before, additionally to the constraints (21), the following constraints should be enforced:

$$\begin{aligned}
e_{i,j}^x(t, l) &= (x_i(0) - x_j(0)) + h \sum_{\tau=0}^{t-1} (u_i^x(\tau) - u_j^x(\tau)) + \frac{hl}{L} (u_i^x(t) - u_j^x(t)) \\
e_{i,j}^y(t, l) &= (y_i(0) - y_j(0)) + h \sum_{\tau=0}^{t-1} (u_i^y(\tau) - u_j^y(\tau)) + \frac{hl}{L} (u_i^y(t) - u_j^y(t)) \\
e_{i,j}^x(t, l) &\geq \Delta + h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) + 3\frac{hl}{L}\bar{\sigma}_{i,j}(t) - r_{ij}(t) - Md_{i,j}^1(t, l) \\
e_{i,j}^x(t, l) &\leq -\Delta - h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) - 3\frac{hl}{L}\bar{\sigma}_{i,j}(t) + r_{ij}(t) + Md_{i,j}^2(t, l) \\
e_{i,j}^y(t, l) &\geq \Delta + h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) + 3\frac{hl}{L}\bar{\sigma}_{i,j}(t) - r_{ij}(t) - Md_{i,j}^3(t, l) \\
e_{i,j}^y(t, l) &\leq -\Delta - h \sum_{\tau=0}^{t-1} 3\bar{\sigma}_{i,j}(\tau) - 3\frac{hl}{L}\bar{\sigma}_{i,j}(t) + r_{ij}(t) + Md_{i,j}^4(t, l) \\
\sum_{\nu=1}^4 d_{i,j}^\nu(t, l) &\leq 3, \quad d_{i,j}^\nu(t, l) \in \{0, 1\} \\
0 \leq r_{ij}(t) \leq M_r \delta_r(t), \quad \delta_r(t) &\in \{0, 1\}
\end{aligned} \tag{23}$$

where

$$\eta_i \triangleq \begin{bmatrix} \eta_0^{i,x} \\ \eta_0^{i,y} \\ \eta_1^{i,x} \\ \eta_1^{i,y} \\ \vdots \\ \eta_{N-1}^{i,x} \\ \eta_{N-1}^{i,y} \end{bmatrix}, \Theta_i \triangleq \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ \theta_{1,0}^{i,x} & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \theta_{1,0}^{i,y} & 0 & 0 & \cdots & 0 & 0 \\ \theta_{2,0}^{i,x} & 0 & \theta_{2,1}^{i,x} & 0 & \cdots & 0 & 0 \\ 0 & \theta_{2,0}^{i,y} & 0 & \theta_{2,1}^{i,y} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \theta_{N-1,0}^{i,x} & 0 & \theta_{N-1,1}^{i,x} & 0 & \cdots & \theta_{N-1,N-2}^{i,x} & 0 \\ 0 & \theta_{N-1,0}^{i,y} & 0 & \theta_{N-1,1}^{i,y} & \cdots & 0 & \theta_{N-1,N-2}^{i,y} \end{bmatrix}.$$

This formulation reduces the conservatism of the MPC algorithm, allowing a more efficient utilization of the airspace. On the other hand, this comes at a high computation cost, as shown in Section 4.5.

As indirect feedback is present even in open-loop MPC by construction, i.e. when the new input sequence will be calculated, the new state of the system will be known, and thus will be taken into consideration. Therefore, the main difference between open-loop MPC and feedback MPC lies in the fact that in open-loop, when at each time the input sequence is calculated, the fact that only a part of it will be applied is ignored and the input sequence is calculated as if it would all be used. This affects the constraints, as, the longer the horizon, the higher the effect of the uncertainty. This should be clear also from the constraints (21), where the constraints are tightened by an additional $3\bar{\sigma}_{i,j}(\tau)$ for every further step in the horizon they are applied. Guided by a similar approach in [26], an alternate approach is here proposed, saturating this constraint tightening after one step. According to this hypothesis, the constraints (21) become:

$$\begin{aligned} x_i(0) - x_j(0) + h \sum_{\tau=0}^{t-1} (u_i^x(\tau) - u_j^x(\tau)) &\geq \Delta + 3h\bar{\sigma}_{i,j}(0) - r_{ij}(t) - Md_{i,j}^1(t) \\ x_i(0) - x_j(0) + h \sum_{\tau=0}^{t-1} (u_i^x(\tau) - u_j^x(\tau)) &\leq -\Delta - 3h\bar{\sigma}_{i,j}(0) + r_{ij}(t) + Md_{i,j}^2(t) \\ y_i(0) - y_j(0) + h \sum_{\tau=0}^{t-1} (u_i^y(\tau) - u_j^y(\tau)) &\geq \Delta + 3h\bar{\sigma}_{i,j}(0) - r_{ij}(t) - Md_{i,j}^3(t) \\ y_i(0) - y_j(0) + h \sum_{\tau=0}^{t-1} (u_i^y(\tau) - u_j^y(\tau)) &\leq -\Delta - 3h\bar{\sigma}_{i,j}(0) + r_{ij}(t) + Md_{i,j}^4(t) \\ \sum_{\nu=1}^4 d_{i,j}^\nu(t) &\leq 3, \quad d_{i,j}^\nu(t) \in \{0, 1\} \\ 0 \leq r_{ij}(t) &\leq M_r \delta_r(t), \quad \delta_r(t) \in \{0, 1\}, \end{aligned} \tag{27}$$

for all $t \in \{0, \dots, N\}$ and $i, j \in \mathcal{I}$ with $i \neq j$.

Similarly, the constraints in all other cases can be relaxed, simply by setting $\bar{\sigma}_{i,j}(\tau) = 0$, for all $\tau > 0$ and in the case where the correlation structure is not taken into account, simply by assuming that $w_i(\tau) = 0$, for all $\tau > 0$.

A.3 Overall finite horizon optimization problem

Concerning the finite horizon optimization problem to be solved at each time step, several alternatives have been presented. In order to recap, all those approaches are shown in Table 20. Furthermore, some simplified cases are provided in the table as simplifications of other cases. In those cases, this is not the most efficient way to encode the constraints, but for brevity reasons, they are presented in this fashion.

Constraint set	Alternative	Constraints
Dynamical constraints	-	(4), (6)
Priorities	Yes	(7)
	No	(7), $\delta_i = 1, \forall i \in \mathcal{I}$
Constraint relaxation	Yes	(13)
	No	(8), $r_{ij}(t) = 0, \forall i, j \in \mathcal{I}, t \in \{0, \dots, N\}$
Wind	Correlated	(17), (21)
	Uncorrelated	(12)
	No wind	$w_i(t) = 0, \forall i \in \mathcal{I}, t \in \{0, \dots, N-1\}$
Inter-sample constraints	Yes	(22) or (23)
	No	(24)
Feedback	Yes	(26)
	No	-
	Saturation	$\bar{\sigma}_{i,j}(\tau) = 0$ or $w_i(\tau) = 0, \forall \tau > 0, i, j \in \mathcal{I}$

Table 20: Constraints and alternatives presented

The MPC problem to be solved periodically can then be written as:

$$\begin{aligned}
 & \min_{\substack{u_i(t), i \in \mathcal{I} \\ t \in \{0, \dots, N-1\}}} \mathcal{J}(\cdot) \\
 & \text{subject to} \quad \text{Dynamical, priority, relaxation, wind, inter-sample} \\
 & \quad \quad \quad \text{and feedback constraints according to Table 20}
 \end{aligned} \tag{28}$$

where the corresponding constraints have to be substituted from Table 20. In all the cases, the resulting problem is a MILP. Even though in general, MILPs are in general NP-hard, several tools (see for instance [10]) have been developed that are able to effectively solve reasonably sized instances of such problems, as demonstrated in Section 4.5.

As an example, the finite horizon optimization problem for a prioritized algorithm, without allowing relaxation of constraints, using the wind correlation structure, enforcing constraints between the samples and without the use of feedback can be written as:

$$\begin{aligned}
& \min_{\substack{u_i(t), i \in \mathcal{I} \\ t \in \{0, \dots, N-1\}}} \mathcal{J}(\cdot) \\
& \text{subject to} \quad (4), (6), (7), (8), (17), (21), (23) \\
& \quad \quad \quad r_{ij}(t) = 0, \forall i, j \in \mathcal{I}, t \in \{0, \dots, N\}.
\end{aligned} \tag{29}$$

A.4 FMS, autopilot and dynamics

Once the optimal inputs $u_i(t)$ are calculated, a simplified FMS controller translates them into thrust and bank angle commands for the autopilot to implement on the aircraft dynamics through the equations:

$$T_i = \begin{cases} C_{Tdes_i}(\text{FP}) T_{MaxClimb_i} & \text{if } \|u_i(t)\|_2 + \delta_{tol} > V_i \\ 0.95 T_{MaxClimb_i} & \text{if } \|u_i(t)\|_2 - \delta_{tol} < V_i \\ \frac{C_{D_i} S_i \rho}{2} \|u_i(t)\|_2^2 & \text{else} \end{cases} \tag{30a}$$

$$\Psi_i(t) = \tan^{-1} \left(\frac{u_i^y(t)}{u_i^x(t)} \right) \tag{30b}$$

$$\phi_i^1 = k_1 \begin{bmatrix} -\sin \Psi_i(t) \\ \cos \Psi_i(t) \end{bmatrix}^T \begin{bmatrix} x_i - x_i(t) \\ y_i - y_i(t) \end{bmatrix} + k_2 (\psi_i - \Psi_i(t)) \tag{30c}$$

for all $t \in \{0, \dots, N-1\}$ and $i \in \mathcal{I}$, where δ_{tol} is a small tolerance to avoid chattering around the desired airspeed.

To avoid unrealistically large bank angles and aircraft travelling in circles, saturation on the linear controller ϕ_i^1 have to be introduced:

$$\phi_i^2 = \min\{\max\{\phi_i^1, -\frac{\pi}{6}\}, \frac{\pi}{6}\}, \quad \phi_i = \begin{cases} \min\{\phi_i^2, 0\}, & \pi/2 \geq \psi_i - \Psi_i(t) \geq \frac{\pi}{3} \\ \max\{\phi_i^2, 0\}, & \pi/2 \geq \Psi_i(t) - \psi_i \geq \frac{\pi}{3} \end{cases}. \tag{31}$$

Thrust and bank angle commands T_i and ϕ_i respectively are then implemented by the autopilot on the system dynamics (1), according to which the system evolves.

A.5 Overall hierarchical formulation

The scheme proposed for the CR problem is summarized in Algorithm 3. The problem is solved every h minutes and when the optimal solution is calculated, only the first step is applied. Then, the first step of the optimal solution is translated through the FMS into thrust and bank angle commands that the autopilot implements on the aircraft dynamics for h minutes. The

procedure is then repeated in a receding horizon fashion, until all aircraft reach their destination.

Algorithm 3 Prioritized Hierarchical Algorithm

Require: $Q_i(\tau), \tau = 0, p_i^d$ and $s(i) \forall i \in \mathcal{I}$

- 1: **while** $\exists i$ s.t. $\left\| \begin{bmatrix} X_i(\tau) \\ Y_i(\tau) \end{bmatrix} - p_i^d \right\|_2 > \mathcal{D}$ **do**
 - 2: Set $p_i(0) = \begin{bmatrix} X_i(\tau) \\ Y_i(\tau) \end{bmatrix}$, for all $i \in \mathcal{I}$
 - 3: Solve the MPC problem (28)
 - 4: Evolve the system according to (30), (31) in the interval $[\tau h, (\tau + 1)h[$
 - 5: Set $\tau = \tau + h$
 - 6: Measure new aircraft position $Q_i(\tau)$, for all $i \in \mathcal{I}$
 - 7: **end while**
-

A.6 Model Parameters

This subsection of the appendix lists all the parameters that are considered in the formulation of the mid term CR algorithm.

Agent	Parameter Name	Parameter	Description	Value	Source
MPC algorithm	Sampling period	h	How often the resolution is updated	3 min	Assumption
	Prediction Horizon	T	Horizon over which CR is performed	15 min	Assumption
	Inter-sample constraints	L	On how many instants in the sampling period the constraints are enforced	6	Calculations
	CR separation	Δ	Required horizontal separation	5 nm	Concept
	Cost	J	Cost of maneuvers	Deviation from nominal speed	Assumption
	Wind bounds	w_i	Wind bounds as assumed by MPC	3σ	Assumption
	ADS-B	R^{ADS-B}	ADS-B range	∞	Assumption
Aircraft	Type		Type of aircraft assumed (if not defined)	A330	Assumption
	Nominal speed	u_i^{nom}		228 <i>m/sec</i>	Model
	Max speed	u_{max}	Maximum allowed speed	260 <i>m/sec</i>	BADA
	Min speed	u_{min}	Minimum allowed speed	180 <i>m/sec</i>	BADA
	Max acceleration	δu	Maximum allowed speed change between two consecutive timesteps	40 <i>m/sec</i> (0.22 <i>m/sec</i> ²)	Model
Environment	Wind forecast		Meteorological forecast	0	Model
	Wind error standard deviation	σ	Standard deviation of wind error assumed	4.77 <i>m/sec</i>	Model
	Wind on X axis	$Wi1$	Forecast error model	$N(0, \sigma)$	Model
	Wind on Y axis	$Wi2$	Forecast error model	$N(0, \sigma)$	Model
	Wind on Z axis	$Wi3$	Forecast error model	0	Model
	FL	FL	Flight Level	33	Model

B MMPC Technical Details

B.1 Variable Horizon

In this section we briefly recap the variable horizon formulation employed. The target regions which the aircraft are required to enter on completion of any required conflict resolution manoeuvres are rectangular, so that the target region S_n allocated to aircraft n is defined as

$$S_n := \{r_n : Qr_n \leq q_n\}. \quad (32)$$

Optimisation of the horizon length is enabled with the optimisation of a binary input decision variable associated with each step in the prediction horizon, determining the point at which the target region constraints become active. Specifically, if the binary associated with point i in the prediction horizon takes the value 1, so that $t_n(k+i|k) = 1$, the target region constraint defined in (32) is active, and is predicted to be met at the next step in the prediction horizon:

$$t_n(k+i|k) = 1 \Rightarrow r_n(k+i+1|k) \in S_n. \quad (33)$$

This is achieved by the following constraint coupling the binary input variable t to the target region constraint in (32), and the use of the ‘big-M’ [1] formulation described earlier:

$$Qr_n(k+i+1|k) \leq q_n(i) + \mathbf{1}.M(1 - t_n(k+i|k)) \quad (34)$$

where M is a large positive integer, exceeding the largest possible value of the state. We impose the terminal constraint that the target set is reached by the end of the prediction horizon,

$$\sum_{i=0}^{N-1} t_n(k+i|k) = 1. \quad (35)$$

We denote the prediction of the target time of arrival made at time k by $N(k)$, so that the terminal constraint of agent n is given by

$$x_n(k + N_n(k)) \in S_n.$$

B.2 Modified Robust MMPC Algorithm

Before outlining the modified MMPC algorithm in detail, we define first some additional notation required in Table 21. The maximal prediction horizon employed for group \mathcal{G}_j is obtained from

$$N_j = \max_{n \in \mathcal{G}_j} \mu.\bar{N}_n + \tau_n - k_j \quad (36)$$

k_0	Initial time $k_0 = 0$
k_j	Times at which joint solutions for aircraft in cooperating set are obtained
τ_n	Predicted time of entry of aircraft n
\bar{N}_n	Time taken for constant speed (cruising speed) straight line trajectory of aircraft n
\mathcal{G}_j	Cooperating set of aircraft at time k_j
N_j	Maximal prediction horizon for cooperative set at time k_j obtained from (36)
T	Look ahead time for detecting potential entry of new aircraft
m_j	Number of aircraft joining on interval $[k_j, k_j + T]$
G_j	$G_j = \{m_{j-1}, m_{j-1} + 1, \dots, m_{j-1} + m_j\}$

Table 21: Notation

where \bar{N}_n is the time associated with a straight line constant speed trajectory and $\mu > 1$ is a scaling factor to allow for the additional time required for performing a conflict resolution manoeuvre. The second term $\tau_n - k_j$ is the number of steps into the prediction horizon at which aircraft n appears in the scenario.

We define now the joint optimisation problem solved for cooperating sets of aircraft in Problem B.1. The joint problem that must be solved at the predefined times k_j is given by

Problem B.1 *Minimise:*

$$V_C(k) = \sum_{n \in G_j} \sum_{i=0}^{N_j-1} (\gamma \|u_n(k+i|k)\|_1 + it_n(k+i|k)) \quad (37)$$

with respect to inputs $u_n(k+i|k)$ and binary inputs $t_n(k+i|k)$ for all $n \in G_j$,

subject to the nominal prediction model and tightened constraints:

$$x(k|k) = x(k) \quad (38a)$$

$$x(k+i|k) \in \mathcal{X}(i) \quad \forall i \geq (\tau_n - k_j) \quad (38b)$$

$$u_n(k+i|k) \in \mathcal{U}_n(i) \quad \forall i \geq (\tau_n - k_j) \quad (38c)$$

$$y_n(k+i|k) \in \mathcal{Y}_n(i) \quad \forall i \geq (\tau_n - k_j) \quad (38d)$$

$$x_n(k+N(k)|k) \in \mathcal{S}_n(N(k)) \quad \forall i \geq (\tau_n - k_j) \quad (38e)$$

$$t_n(k+i|k) \in \{0, 1\} \quad \forall i \geq (\tau_n - k_j) \quad (38f)$$

$$t_n(k+i|k) = 0 \quad \forall i < (\tau_n - k_j) \quad (38g)$$

the constraints in (34), the terminal constraint that all targets are reached by the end of the horizon

$$\sum_{n \in \mathcal{G}_j} \sum_{i=0}^{N_j-1} t_n(k+i|k) = m_j \quad (38h)$$

and the equality constraints on the non-optimising agents which are not included in the cooperating set \mathcal{G}_j , formed from the solution obtained at the previous time step

$$u_n(k+i|k) = u_n(k+i|k-1) + \tilde{K}_n(i-1)\tilde{L}(i-1)w(k-1) \quad \forall n \neq \sigma(k), \quad (38i)$$

The coupled constraint sets $\mathcal{X}(i)$, the input constraint sets $\mathcal{U}_{\sigma(k)}(i)$, the local constraint sets $\mathcal{Y}_n(i)$ and the terminal target set regions $\mathcal{S}_n(i)$ are only enforced once the aircraft have entered the region of interest, after $(\tau_n - k_j)$, and are tightened according to the constraint tightening relations presented in [6].

The maximal prediction horizon length, N_j , has to be chosen according to (36) so that a solution to (B.1) exists.

The multiplexed problem solved by aircraft $\sigma(k)$ at time k is given by:

Problem B.2 *Minimise:*

$$V_{\sigma(k)}(k) = \sum_{i=0}^{N_j-1} (\gamma \|u_{\sigma(k)}(k+i|k)\|_1 + it_{\sigma(k)}(k+i|k)) \quad (39)$$

with respect to inputs $u_{\sigma(k)}(k+i|k)$ and binary inputs $t_{\sigma(k)}(k+i|k)$, subject to the nominal prediction model and tightened constraints

$$x(k|k) = x(k) \quad (40a)$$

$$x(k+i|k) \in \mathcal{X}(i) \quad (40b)$$

$$u_{\sigma(k)}(k+i|k) \in \mathcal{U}_{\sigma(k)}(i) \quad (40c)$$

$$y_{\sigma(k)}(k+i|k) \in \mathcal{Y}_{\sigma(k)}(i) \quad (40d)$$

$$x_{\sigma(k)}(k+N(k)|k) \in \mathcal{S}_{\sigma(k)}(N(k)) \quad (40e)$$

$$t_{\sigma(k)}(k+i|k) \in \{0, 1\}, \quad (40f)$$

the constraint on the non-optimising agents

$$u_n(k+i|k) = u_n(k+i|k-1) + \tilde{K}_n(i-1)\tilde{L}(i-1)w(k-1) \quad \forall n \neq \sigma(k), \quad (40g)$$

obtained from the solution from the previous time step, and the terminal constraint in (35). The equality constraints on the non-optimising agents in (40g) is initially derived from the solution to the centralised problem outlined in problem B.1. The coupled constraint sets $\mathcal{X}(i)$, the input constraint sets $\mathcal{U}_{\sigma(k)}(i)$, the local constraint sets $\mathcal{Y}_n(i)$ and the terminal target set regions $\mathcal{S}_n(i)$ are tightened according to the constraint tightening relations presented in [6].

We detail now the modified variable horizon multiplexed algorithm executed by aircraft n in Algorithm 4:

Algorithm 4 Variable Horizon Robust Multiplexed MPC with constraint tightening and synchronous initialisation

- 1: Design stabilising $K(i)$
 - 2: Tighten constraint sets $\mathcal{X}(i), \mathcal{U}(i)$ according to the constraint tightening relations presented in [6]
 - 3: Receive centralised solution obtained at initial time k_0 $\mathbf{u}_n^*(k_0), \mathbf{t}_n(k_0)$ to Problem B.1 and if $(\tau_n - k_j) = 0$ apply the first input $u_n^*(k_0|k_0)$;
 - 4: Wait one timestep; $k = k_0 + 1$;
 - 5: **while** $m > 0$ **do**
 - 6: **if** $k = k_j$ **then**
 - 7: $m \leftarrow m + m_j$
 - 8: **if** $n = \sigma(k)$ **then**
 - 9: Obtain $\min\{\mathbf{u}_{\sigma(k)}^*(k), \hat{\mathbf{u}}_{\sigma(k)}(k)\}$ as the solution to Problem B.2
 - 10: Transmit plan and state information to all agents;
 - 11: **else**
 - 12: Renew current plan according to disturbance feedback policy
 - 13: **end if**
 - 14: **else**
 - 15: **if** $n = \sigma(k)$ **then**
 - 16: Obtain $\min\{\mathbf{u}_{\sigma(k)}^*(k), \hat{\mathbf{u}}_{\sigma(k)}(k)\}$ as the solution to Problem B.2
 - 17: Transmit plan and state information to all agents;
 - 18: **else**
 - 19: Renew current plan according to disturbance feedback policy
 - 20: **end if**
 - 21: **end if**
 - 22: Increment control input by first step in plan
 - 23: Wait one timestep, $k \leftarrow k + 1$
 - 24: $m \leftarrow m - \sum_n (1 - t_n(k))$
 - 25: **end while**
-

References

- [1] A Bemporad and M Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35:407–427, 1999.
- [2] A. Ben-Tal, S. Boyd, and A. Nemirovski. Extending scope of robust optimization: Comprehensive robust counterparts of uncertain problems. *Journal of Mathematical Programming*, 107:63–89, 2006.
- [3] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.
- [4] G. Chaloulos, P. Hokayem, and J. Lygeros. Hierarchical Control with Prioritized MPC for Conflict Resolution in Air Traffic Control. In *18th IFAC World Congress*, Milano, Italy, August 2011.
- [5] G. Chaloulos, J. Lygeros, G. Roussos, K. Kyriakopoulos, E. Siva, A. Lecchini-Visintini, and P. Casek. Comparative study of conflict resolution methods. *iFly Deliverable D5.1*, 2007.
- [6] G. Chaloulos, J. Lygeros, G. Roussos, K. Kyriakopoulos, E. Siva, and J.M. Maciejowski. Report on advanced conflict resolution mechanisms for a³ conops. *iFly Deliverable D5.3*, 2010.
- [7] G. Cuevas, I. Echegoyen, J. García, P. Cásek, C. Keinrath, F. Bussink, and A. Luuk. Autonomous Aircraft Advanced (A3) ConOps. Technical Report Deliverable D1.3, iFly Project, 2008.
- [8] Eurocontrol Experimental Centre. *User Manual for the Base of Aircraft Data (BADA)*. 2004.
- [9] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4):523–533, 2006.
- [10] ILOG SA. CPLEX 11.0 Users Manual. Technical report, Gentilly, France, 2008.
- [11] N. Kantas, J.M. Maciejowski, A. Lecchini-Visintini, G. Chaloulos, J. Lygeros, G. Roussos, A. Oikonomopoulos, K. Kyriakopoulos, and P. Casek. Analysis of conflict resolution needs of A³ operational concept. Technical Report Deliverable 5.2 (version 1.0), iFly Project, 2008.
- [12] E.C. Kerrigan, A. Bemporad, D. Mignone, M. Morari, and J.M. Maciejowski. Multi-objective prioritisation and reconfiguration for the control of constrained hybrid systems. In *American Control Conference*, volume 3, pages 1694–1698, Chicago, IL , USA, June 2000.

- [13] J. Löfberg. *Minimax Approaches to Robust Model Predictive Control*. PhD thesis, Linköpings Universitet, 2003.
- [14] J. Lofberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, 2004.
- [15] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *2004 IEEE International Symposium on Computer Aided Control Systems Design*, pages 284–289. IEEE, 2005.
- [16] J. Lofberg. Automatic robust convex programming. *Optimization methods and software*, 2011. Accepted for publication.
- [17] J. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2001.
- [18] A. Richards and J.P. How. Aircraft trajectory planning with collision avoidance using mixed integer linear programming. In *American Control Conference*, volume 3, pages 1936–1941, Anchorage, Alaska, USA, May 2002. IEEE.
- [19] Elor Rimon and Daniel E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–508, 1992.
- [20] Giannis Roussos and Kostas J. Kyriakopoulos. Decentralised navigation and collision avoidance for aircraft in 3d space. *2010 American Control Conference, Baltimore, USA*, 2010.
- [21] Giannis Roussos and Kostas J. Kyriakopoulos. Decentralized & prioritized navigation and collision avoidance for multiple mobile robots. In *10th International Symposium on Distributed Autonomous Robotics Systems, Lausanne, Switzerland.*, 2010.
- [22] Giannis P. Roussos and Kostas J. Kyriakopoulos. Towards constant velocity navigation and collision avoidance for autonomous nonholonomic aircraft-like vehicles. *Conference on Decision and Control*, 2009.
- [23] Single European Sky ATM Research - SESAR. European ATM Master Plan, 2009.
- [24] E. Siva, J. Maciejowski, G. Chaloulos, J. Lygeros, G. Roussos, and K. Kyriakopoulos. Intermediate report on advanced conict resolution mechanisms for a3 conops. technical report deliverable 5.3i (version 1.0). *iFly Project*, 2009.
- [25] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1:146, 1972.

- [26] J. Yan and R.R. Bitmead. Incorporating state estimation into model predictive control and its application to network traffic control. *Automatica*, 41(4):595–604, 2005.