



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 D5.3**  
**Report on advanced conflict resolution mechanisms for A<sup>3</sup>**  
**ConOps**  
Version:Final (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		
<b>PU</b>	Public	✓
<b>PP</b>	Restricted to other programme participants (including the Commission Services)	
<b>RE</b>	Restricted to a group specified by the consortium (including the Commission Services)	
<b>CO</b>	Confidential, only for members of the consortium (including the Commission Services)	

## DOCUMENT CONTROL SHEET

**Title of document:** *Intermediate report on advanced conflict resolution mechanisms for A<sup>3</sup> ConOps*

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

**Deliverable number:** *D5.3*

**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	12.12.2009	All	First draft
0.2	12.02.2010	All	Second draft
0.3	12.07.2010	All	Implementation of comments received
0.4	08.11.2010	All	Third draft
0.4	12.11.2010	All	Review based on WP5 internal review
0.5	10.05.2011	All	Revision based on internal comments provided by NLR
1.0	23.08.2011	All	Revision based on external comments by U. Voelckers
1.0	04.09.2011	All	Final revision based on external comments by U. Voelckers

Version		Organisation	Signature/Date
<b>Authors</b>	E. Siva	UCAM	
	J.M. Maciejowski	UCAM	
	G. Chaloulos	ETHZ	
	J. Lygeros	ETHZ	
	K. Kyriakopoulos	NTUA	
	G. Roussos	NTUA	
	<b>Internal reviewers</b>	A. Lecchini-Visintini	ULES
	H.A.P. Blom	NLR	
<b>External reviewers</b>	U. Voelckers	Independent	

iFly, Work Package WP5, D5.3  
Report on advanced conflict resolution mechanisms for  $A^3$   
ConOps

Edited by:  
UCAM

September 4, 2011

**Abstract**

This is the report deliverable of Work Package 5.3 of the iFly project. Within this work package WP5.3, the partners develop individual designs, focusing on short and mid term conflict resolution. For the short term, a decentralised Navigation Function (NF) approach is proposed. Three alternative methods are proposed for the mid term; the first, a decentralised scheme, combines the NF approach with Model Predictive Control (MPC), a feedback control formulation which underpins all three of the mid term approaches. The MPC and Navigation Functions approach inherits the desirable attributes of both methodologies. The second mid term approach, Distributed Multiplexed Model Predictive Control (MMPC) is presented with a number of possible variations of the general approach. The final method, Hierarchical MPC, maps most directly to the requirements of the Autonomous Aircraft Advanced ( $A^3$ ) Concept of Operations (ConOps) but development in decentralising the scheme is still undergoing. For each of the methods, the algorithms used are described and simulation results are presented. The approaches are assessed against the requirements of the  $A^3$  ConOps, and their relative merits and features summarised. For the long-term, the potential for extending the mid-term approaches is described.

## List of Acronyms

<b>A<sup>3</sup></b>	Autonomous Aircraft Advanced
<b>ADS-B</b>	Automatic Dependent Surveillance Broadcast/Contract
<b>ATC</b>	Air Traffic Control
<b>BADA</b>	Base of Aircraft Data
<b>CD</b>	Conflict Detection
<b>CD&amp;R</b>	Conflict Detection and Resolution
<b>ConOps</b>	Concept of Operations
<b>CR</b>	Conflict Resolution
<b>FMS</b>	Flight Management System
<b>LoS</b>	Loss of Separation
<b>LTACD</b>	Long Term Area Conflict Detection
<b>MCMC</b>	Markov Chain Monte Carlo
<b>MILP</b>	Mixed Integer Linear Program
<b>MMPC</b>	Multiplexed Model Predictive Control
<b>MPC</b>	Model Predictive Control
<b>NF</b>	Navigation Function
<b>RBT</b>	Reference Business Trajectory
<b>SWIM</b>	System Wide Information Management
<b>TFM</b>	Traffic Flow Management
<b>V-MMPC</b>	Variable Update Order MMPC
<b>F-MMPC</b>	Fixed Order MMPC
<b>MB-MMPC</b>	Move-Blocking MMPC
<b>SMPC</b>	Synchronous (Centralised) MPC

# Contents

<b>1</b>	<b>Introduction</b>	<b>7</b>
1.1	iFly WP5 . . . . .	7
1.2	Objectives of this deliverable . . . . .	7
1.3	Interaction with other Work Packages . . . . .	8
1.4	Overview of Conflict Resolution Methods and Organisation of Report . . . . .	8
<b>2</b>	<b>Short term conflict Resolution: Navigation Functions</b>	<b>11</b>
2.1	Navigation Functions with limited sensing range . . . . .	12
2.2	Simulation . . . . .	12
<b>3</b>	<b>Mid term conflict resolution: MPC and Navigation Functions</b>	<b>17</b>
3.1	Introduction . . . . .	17
3.2	Hierarchical formulation . . . . .	17
3.3	Decentralized strategy . . . . .	18
3.4	Simulation setup and Results . . . . .	19
<b>4</b>	<b>Mid term conflict resolution: Distributed Robust Multiplexed Model Predictive Control</b>	<b>24</b>
4.1	Recap of Robust Distributed MMPC . . . . .	24
4.2	Problem Formulation and Assumptions . . . . .	25
4.3	MMPC with disturbance feedback control . . . . .	25
4.3.1	Fixed order MMPC with disturbance feedback control . . . . .	26
4.3.2	Variable update order MMPC with disturbance feedback control . . . . .	26
4.4	Exploiting Disturbance Correlations . . . . .	28
4.5	Aircraft Entering and Leaving the Region of Interest . . . . .	30
4.5.1	MMPC . . . . .	30
4.5.2	Robust MMPC with Disturbance Feedback . . . . .	31
4.6	Simulation Results . . . . .	31
<b>5</b>	<b>Mid term conflict resolution: Hierarchical MPC with Priorities</b>	<b>35</b>
5.1	Introduction . . . . .	35
5.2	Simulation Results . . . . .	36
5.2.1	Simulation Setup . . . . .	36
5.2.2	Effect of priorities . . . . .	39
5.2.3	Effect of wind correlation . . . . .	40
5.3	Conclusions . . . . .	40
<b>6</b>	<b>Compatibility of medium and short-term methods proposed with A<sup>3</sup>ConOps requirements</b>	<b>42</b>
6.1	Mid-term Methods . . . . .	42
6.2	Short-term Methods . . . . .	43
<b>7</b>	<b>Concluding remarks</b>	<b>45</b>
	<b>References</b>	<b>45</b>
	<b>Appendices</b>	<b>49</b>

<b>A</b>	<b>Navigation Functions (NFs) with limited sensing range</b>	<b>49</b>
A.1	Problem Formulation . . . . .	49
A.1.1	Decentralised Navigation Functions with Local Sensing . . . . .	50
A.2	Short-term Conflict Detection and Resolution (CD&R) using Navigation Functions (NFs) . . . . .	52
A.2.1	Preliminaries . . . . .	52
A.2.2	Control Scheme . . . . .	57
A.3	Conflict resolution Analysis . . . . .	58
<b>B</b>	<b>MPC and NFs</b>	<b>64</b>
B.1	Model dynamics . . . . .	64
B.2	Decentralized MPC . . . . .	66
B.2.1	Cooperative cost . . . . .	66
<b>C</b>	<b>Distributed Robust MMPC</b>	<b>67</b>
C.1	System Model . . . . .	67
C.2	Problem Formulation . . . . .	67
C.2.1	Nominal Constraints . . . . .	67
C.2.2	Variable horizon . . . . .	68
C.2.3	Terminal Set . . . . .	70
C.3	Multiplexed MPC . . . . .	70
C.4	Wind disturbance . . . . .	71
C.5	Multiplexed MPC with Disturbance Feedback . . . . .	71
C.5.1	Fixed Order MMPC . . . . .	75
C.5.2	Variable Update Order MMPC . . . . .	75
C.5.3	Choice of feedback gains . . . . .	76
<b>D</b>	<b>Prioritized MPC</b>	<b>77</b>
D.1	Problem formulation . . . . .	77
D.1.1	Dynamics . . . . .	77
D.1.2	Velocity Constraints . . . . .	77
D.1.3	Separation Constraints . . . . .	78
D.2	Prioritized Hierarchical MPC Solution . . . . .	80
D.2.1	Higher level controller - MPC . . . . .	80
D.2.2	Lower level controller - autopilot . . . . .	82
D.2.3	Overall Hierarchical MPC formulation . . . . .	82

# 1 Introduction

## 1.1 iFly WP5

The objective of WP5 is to investigate and push the limits of conflict resolution algorithms for the A<sup>3</sup>ConOps by WP1 [13]. 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 [6] 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, with emphasis 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 [8].
- 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 [21].
- WP5.3: *Further development of conflict resolution methods.* In order to match the A<sup>3</sup>ConOps requirements further development of the conflict resolution methods is necessary. WP5.3 concentrated on developing those methods. Deliverable D5.3i [42] has already documented the initial indications of the methods chosen to further develop within the WP. The current report documents the final results of the work undertaken within this sub-WP5.3
- 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 that are used in WP7 rare event simulations, and are the best currently known by the autonomous aircraft research community and against the requirements identified in WP5.2.

## 1.2 Objectives of this deliverable

In this deliverable, we provide a current report on the development of the Conflict Resolution (CR) methods. The objective of D5.3 is to document the results of the development of mid and short term conflict resolution algorithms to address the needs of the A3 concept of operations and to validate these developments against the requirements imposed by the concept. The results presented here build on earlier development work documented in D5.3i and D5.3ii. The requirements and specifications set by the A<sup>3</sup>ConOps developed in WP1 have been taken into consideration for each different level of Conflict Detection and Resolution (CD&R). An overview of the different control horizons used in the three CD&R levels is shown in Figure 1.

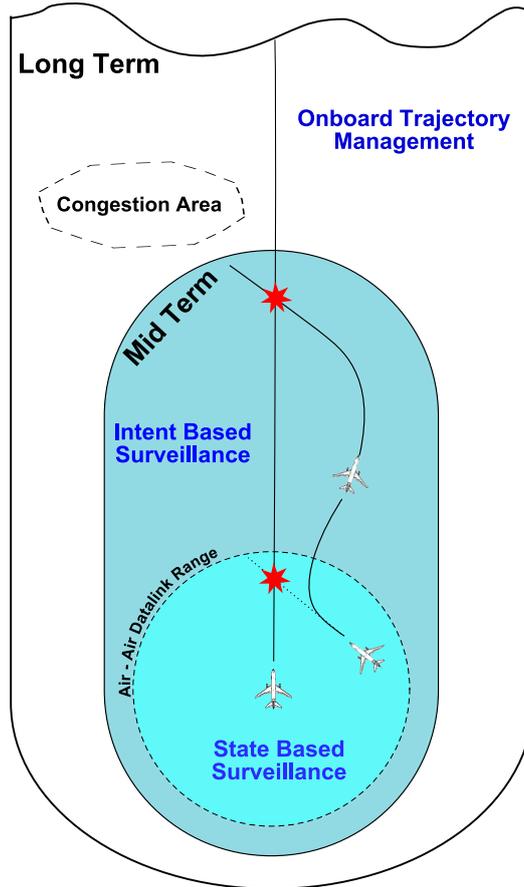


Figure 1: Air Traffic Surveillance in Short, Mid and Long Term CD&R

### 1.3 Interaction with other Work Packages

The operational requirements and the human responsibilities of A<sup>3</sup>ConOps, studied in WP1 [13] and WP2 [2] have been used as an input for the design of the CR methods. The developments of the CR methods presented in this deliverable will be used by WP8 for the concept refinement, as well as WP9 which studies the airborne requirements.

### 1.4 Overview of Conflict Resolution Methods and Organisation of Report

Before introducing the approaches that have been identified for the short, mid and long-term CD&R, we briefly make mention of relevant existing methodologies in the literature. In [29], a pairwise distributed CD&R algorithm, with explicit inclusion of aircraft priorities is presented. A cooperative distributed algorithm termed ‘Satisficing Game Theory’ is presented In [4]. There, rather than seeking to obtain a global optimum for the cost associated with the entire group of aircraft, satisficing aircraft solve for an adequate solution. Cooperation is enforced with the inclusion of a ‘risk’ term in the local objective minimised by each aircraft. This risk term captures the likelihood of conflicts with neighbouring aircraft from the intent information they exchange. The objective minimised by each aircraft incorporates the preferences of other aircraft, enabling a collaborative solution. In [44], a conflict resolution function suitable for tactical intent-based operations is developed.

## **Long Term Conflict Resolution**

Traffic Flow Management (TFM) algorithms (that are operating on ground) have an implicit role, placing some constraints on the flight plan of the aircraft (which might cover self separation parts of the airspace). Their role should be to ensure that the Mid and Short Term algorithms will not be faced with a situation that they cannot handle. The CR methods presented in this deliverable will be used as input to WP8.2 [1] for refining the concept for flow management. With regard to the Long Term Area Conflict Detection (LTACD) defined in the concept [13], the Trajectory Management module should be able to resolve any detected conflicts with ‘areas to avoid’. This part of the Long Term CR could be implemented by using some of the Mid Term CR tools and algorithms, substituting the required input from the Conflict Detection (CD) algorithms with an input from a Complexity Prediction algorithm, along with the ‘areas to avoid’ for the algorithm. Using the hierarchical approach described in Section 3, this could be implemented by adding one more level of hierarchy above the ones mentioned in the Mid Term CR. The necessary steps can be implemented once complexity metrics to be used for this purpose are clarified.

## **Mid and Short term Conflict Resolution**

The Conflict Resolution problem can be viewed as a hierarchically structured problem, as shown in Figure 2. For the short term, Decentralised Navigation Functions have been identified as suitable conflict resolution methods, and are presented in Section 2. Three mid term conflict resolution methods have been developed. The first method which combines the best features of the mid term schemes presented is outlined in Section 3. Technically the formulation involves integrating Model Predictive Control (MPC) with Navigation Functions.

The second formulation, robust multiplexed MPC is presented in Section 4. This method provides stronger theoretical guarantees, but is more difficult to integrate in the hierarchy, as the dynamics of the Short Term are not compatible with the assumed model abstraction. The final method, Hierarchical MPC with priorities, is detailed in Section 5. This formulation maps more directly to the priorities concept described in A<sup>3</sup> ConOps [13], but is still under development and efforts are being concentrated into decentralizing it.

## **Organisation of this report**

In the following sections we briefly summarize the results of the methods developed and their implications for the iFly concept. Specific details of these three methods and the problem formulation are given in the Appendix.

The mappings of the proposed short and mid term approaches to the ConOps requirements are described in Section 6. Finally, concluding remarks are made in Section 7.

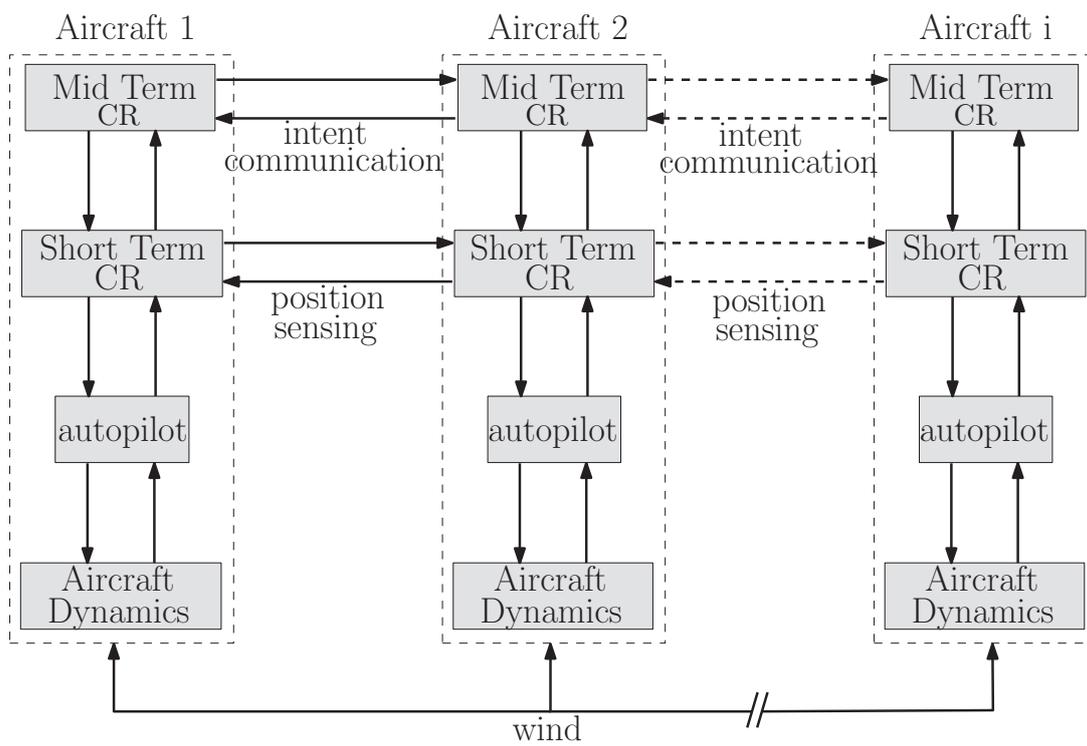


Figure 2: Hierarchical Multi-Level System

## 2 Short term conflict Resolution: Navigation Functions

The requirements for Short-term CD&R within iFly have been described by WP 1 [13] and discussed in previous deliverables of WP 5 [21]. The conclusion of previous WP5 deliverables has been that for short-term CD&R the focus is on safety and decentralisation, while optimality is left to be handled by Mid and Long-term CD&R. Navigation Function (NF) [33] have been identified as a class of algorithms that can be employed in a decentralised fashion and offer performance guarantees, while also complying with most other ConOps requirements. A detailed discussion relating the characteristics of NF-based CD&R with the specific ConOps assumptions and requirements can be found in D5.2 [21]. The main advantages of NF-based approaches are performance guarantees that they can provide, along with their computational efficiency. However compelling though, these characteristics alone are not enough to enable the use of the methodology to the ATM applications.

Application-specific development of the NF methodology has so far been targeted mostly towards robotic applications. NFs have also evolved under previous EU project HYBRIDGE as a powerful methodology for multi-agent navigation and collision avoidance. Developments within HYBRIDGE included a decentralised NF-based algorithm for multi-agent navigation and an approach towards integrating limited sensing range in the algorithm. However, aircraft performance characteristics and constraints were not considered and the algorithm was limited to horizontal manoeuvres.

Within WP5 of iFly our effort has been to adapt the NF framework to aircraft navigation and CD&R, while maintaining its formal properties. Towards this direction, an initial approach towards extending the algorithm to 3 dimensions has been presented [39], enabling the use of both horizontal and vertical maneuvering to minimise separation and thus to maximise airspace capacity. One of the key aspects for the application of NF-based algorithms to aircraft CD&R has been the consideration of specific constraints that apply to aircraft. Such constraints arise mostly from performance limitations of civilian aircraft, though ATM practice can also contribute additional restrictions.

In the initial NF-based approach for 3D CD&R aircraft have been modeled as unicycle agents without any bounds on the inputs. One of the most important constraints that needed to be taken into account by NFs in iFLY has been the limited linear velocity regulation capability of aircraft, and especially the need for a lower bounded velocity. This is necessary to remain within the feasible performance envelope of civil aircraft. Towards this direction, an appropriately adapted NF-based algorithm has been presented in [37] that maintains a constant linear velocity in most cases and is guaranteed to respect a low velocity bound. Furthermore modifications made to the steering control scheme have achieved a measurable reduction in total steering effort and produce more sensible trajectories.

Further development of the algorithms has been presented in [35], where a significantly more realistic way of handling vertical maneuvering has been introduced. Specifically, in this new approach vertical speed is regulated independently of horizontal steering. This decoupling of vertical and horizontal maneuvering has allowed the introduction of additional constraints in the climb and descent angle of aircraft. Directly regulating the vertical speed of aircraft comes natural to ATM and yields trajectories that are much closer to current ATM practice. Moreover, this formulation allows easier integration of higher-level inputs. This can be especially useful in the combined MPC-NFs design for Short and Mid-term CD&R described in 3, where reference inputs provided by MPC can now be directly taken into account by the NFs algorithm.

## 2.1 Navigation Functions with limited sensing range

The Navigation Functions (NFs) methodology belongs to the class of potential field methods, although it offers unique characteristics. In this class of methods, an artificial potential field is created comprising repulsive potentials of obstacles or other agents, and the attractive potential of the destination. The negated gradient of this potential field is then used to derive the control inputs that drive the agent towards the direction that minimises the value of potential. A common weakness of most artificial potential field methods is the existence of local minima away from the destination. Such local minima can attract agents to undesired positions and lead the algorithm to stagnation.

The main advantage of Navigation Functions over other potential field methods is the elimination of all local minima. Navigation functions are used to create potential fields which are repulsive with respect to obstacles or other agents, and attractive with respect to the destination. These potential fields have by construction only a single, global minimum, and can therefore guarantee almost global convergence to the destination, along with provable collision avoidance. The term ‘almost global’ is used here because there are sets of initial conditions that lead to saddle points of the potential field. Since these sets (one for each obstacle or intruder) are of measure zero, convergence to saddle points is extremely rare and very improbable to observe even in simulation or experiments.

In the decentralized implementation of Navigation Functions employed here, each aircraft uses its own potential field, which is calculated using its position and destination, along with the position of other aircraft within its sensing range. A real-time feedback control law based on the potential field’s negated gradient is employed to guide the aircraft. The result is that each aircraft moves along a field flow line, maintaining separation and converging to its destination.

Latest results presented in [36] integrate limited sensing range in the NF algorithm that also significantly improves its practical and numerical behaviour. Thus, the finite communication range of ADS-B can be taken into account naturally in the algorithm. Moreover, aircraft classification and prioritisation has been added as an option in the algorithm, although not required by the ConOps for Short-term CD&R.

It is important to note that the developments described above have not been made at the expense of the formal guarantees provided by the NF framework. The new algorithms manage to produce much more ATM-compliant trajectories while maintaining provable conflict resolution characteristics. In the following sections we present in detail the NFs-based algorithm developed for CD&R within WP 5.

The technical details of the NFs algorithm with limited sensing range can be found in Appendix A. A formal analysis of its convergence and conflict resolution properties is also included. Simulation results of this algorithm are presented below.

## 2.2 Simulation

The proposed control scheme has been used in a simulated scenario. We used a test case consisting of 5 aircraft which start from initial positions near the boundary of the workspace and face inward. The target configurations have been set across the center of the workspace, so that the straight line paths between each start position and the corresponding destination approach each other in the center. The desired horizontal velocity  $u_{id}$  of all the agents has been set to a constant value of  $450knots$  as a typical cruising speed of a commercial aircraft, while the maximum climb and descent angles used are  $\alpha_{iC} = 15^\circ$  and  $\alpha_{iD} = -20^\circ$  respectively. The angle parameters  $\theta_i^0$  and  $\hat{\theta}_i$  have been set to  $\theta_i^0 = 10^\circ$  and  $\hat{\theta}_i = 15^\circ$  for all aircraft. Finally, the radius of all target cylinders  $\mathcal{C}_i$  and target spheres  $\mathcal{S}_i$  has been set to  $c_i = 10nm$ . Regarding CD&R, a minimum separation of  $5nm$  has been used, while the maximum sensing range has

been set to  $37.5nm$ , which amounts to 5 minutes of flight at cruising speed.

The simulation was run on a standard PC that calculated the trajectories with a  $1sec$  resolution. The calculation of all 5 trajectories that represent 20-30 minutes of flight was completed in less than 10 seconds. The results of the simulation are shown in Figures 3 to 5. The aircraft paths are shown in Figures 3 from two different viewing angles. The horizontal linear ( $u_i$ ) and angular velocity ( $\omega_i$ ) are depicted in Figure 4, while the vertical velocities  $w_i$  are shown in Figure 5. As the figures demonstrate, the proposed short-term CD&R algorithm enables all aircraft to reach their destination without any conflicts. Specifically, the following remarks can be made:

- All aircraft maintain their horizontal speed equal to the constant desired value  $u_{id}$ , except for aircraft 4, which uses a higher speed for a limited amount of time, while avoiding a conflict with aircraft 5. In such a case the combination of NFs with MPC will be able to foresee this specific constraint violation and replan in the mid-term to avoid it.
- Aircraft follow level paths, i.e.  $w_i = 0$ , for a significant amount of time. They all approach their destinations with their climb/descent angles converging to zero.
- The bounded angle of climb or descent, in combination with constant horizontal velocity, results in bounded vertical velocity. When  $\alpha_{nh_i}$  is saturated,  $|\alpha_{nh_i}| > |\tilde{\alpha}_i|$ , and  $|u_i| = u_{id}$ , a constant vertical velocity is used. as is common in current ATC practice.
- The combined effects of the two above remarks are obvious in aircraft's 1 path, which follows a *climb-fly level-descent* pattern.
- The initial and final positions of aircraft 4 result in a straight line path with climbing angle greater than  $\alpha_{iC}$ . The aircraft performs a climbing circle to achieve the desired altitude while avoiding conflict with aircraft 5.

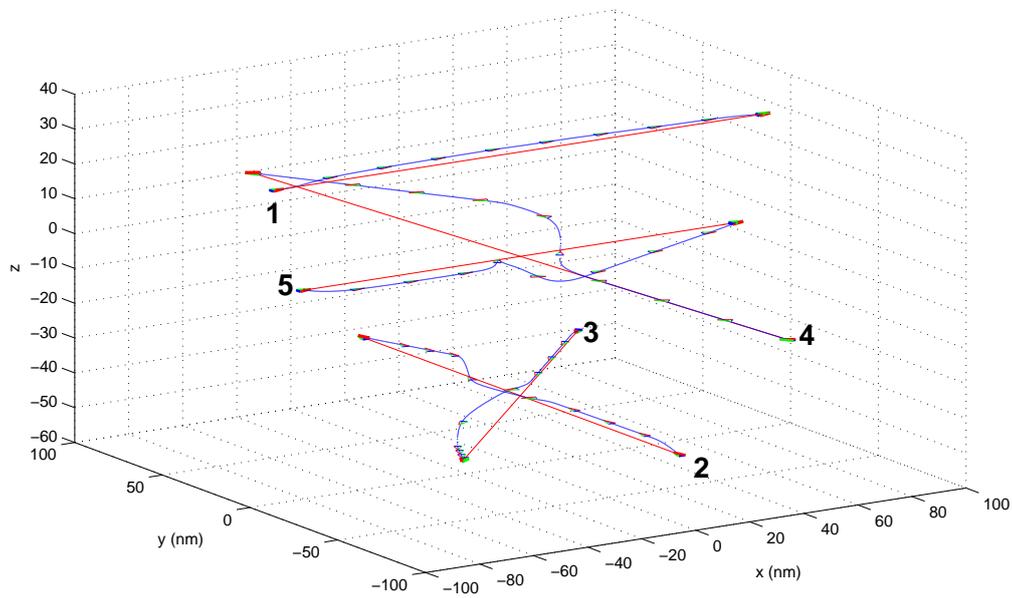
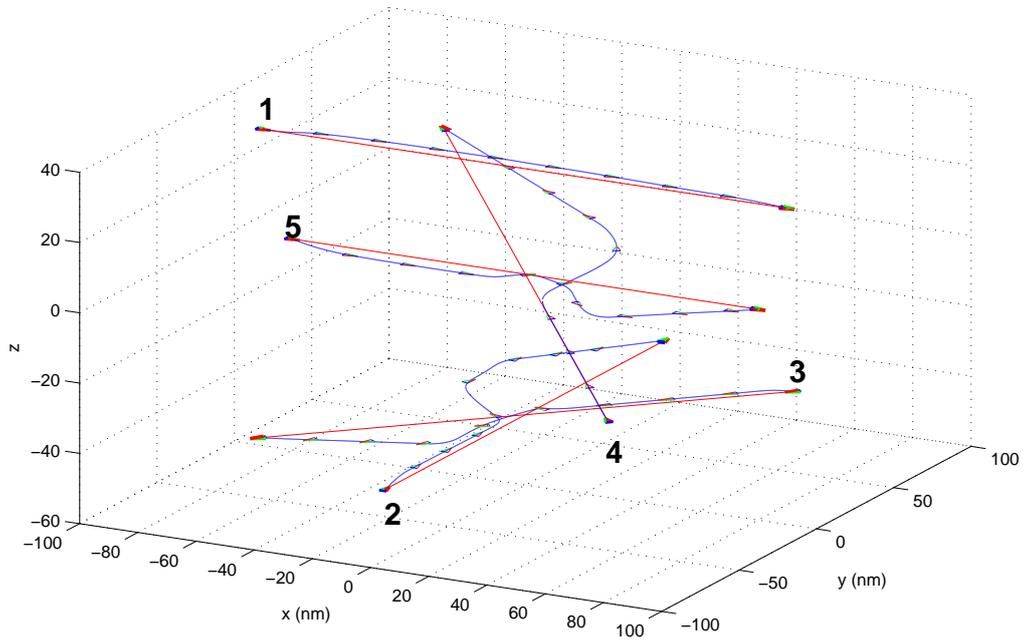


Figure 3: Aircraft's trajectories in 3D space (altitude not in-scale)

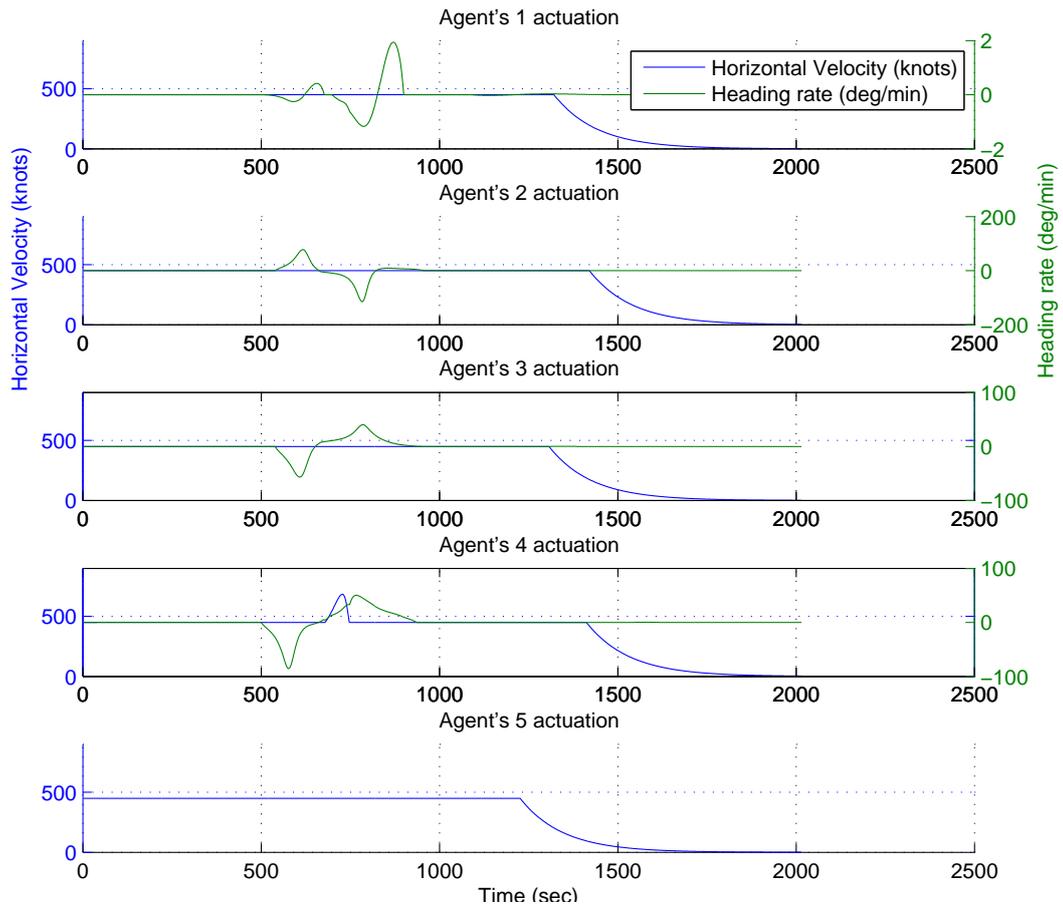


Figure 4: Horizontal and Angular Velocities

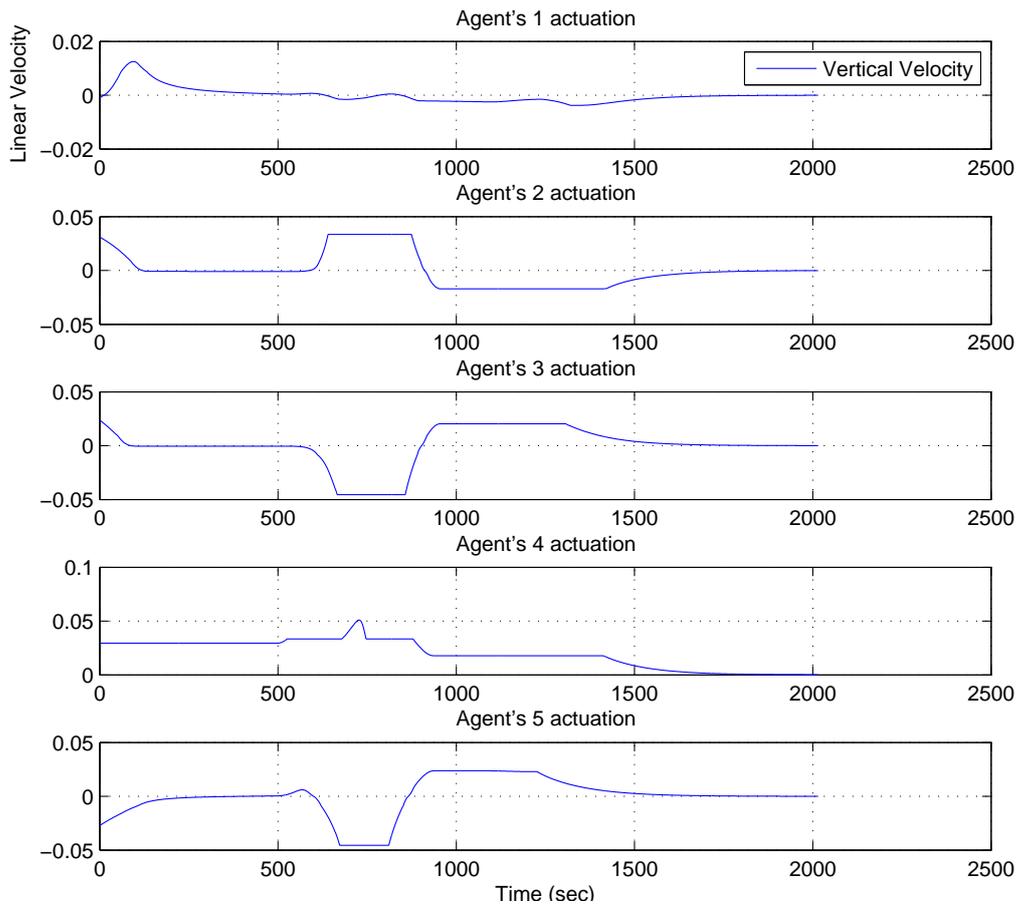


Figure 5: Vertical Velocities

### 3 Mid term conflict resolution: MPC and Navigation Functions

In line with the ConOps requirements described in [13], a formulation combining the best features of the proposed schemes presented in the previous deliverables of WP5 [21] is developed and presented in this section. For completeness reasons, we detail the methods used in this section in the Appendix section B.

#### 3.1 Introduction

Research in this field was focused on producing a more realistic Conflict Resolution algorithm, taking into account the autopilot of the aircraft, as well as wind dynamics, which affect the trajectories of aircraft flying in an  $A^3$  airspace. Thus, we have developed a hierarchical control scheme, where using a low level controller, the aircraft dynamic equations are abstracted to simpler unicycle kinematic equations, enabling the navigation function technology to be used to generate conflict free trajectories for all aircraft involved in a conflicting situation. To ensure that the resulting trajectories respect the aerodynamic constraints of the aircraft, a decentralized model predictive controller is added at a higher level (similar to [21]), to provide preview to the Short Term CR algorithm. We have further explored the decentralization of this scheme, as well as run simulations using some priority rules, in terms of a more enhanced cost function.

Since this section and the following two sections require the use of MPC, we briefly outline its underlying principle.

In the standard form of MPC, a model of the system is parameterised with a sequence of manipulated variables (control inputs) over a finite time horizon. This model is used to form an optimisation problem whose decision variables comprise this sequence of control inputs. The objective to be minimised is an appropriately chosen function of the future output and state trajectories over this horizon starting from the current state. The optimisation problem is solved and the first step of the resulting input sequence is applied. At the next time step, this process is repeated, based on the new measured state of the plant. The horizon length is kept fixed, giving rise to the term ‘receding horizon control’. Whilst the predictions made within the optimisation problem are ‘open-loop’, the recomputation of the optimal finite-horizon trajectory based on the new measurements obtained renders this a ‘closed loop’ control formulation, countering the effect of uncertainty.

#### 3.2 Hierarchical formulation

As discussed, the Navigation Function based control scheme in [39] cannot guarantee that dynamical constraints (e.g. aircraft velocities, etc.) will be respected. To overcome this drawback, we employ the technique of Model Predictive Control (MPC) [28], a control methodology developed specifically to deal with state and input constraints. In an example of such a setting, every 3 minutes a mid-term conflict resolution algorithm decides on the optimal parameters for the Navigation Functions (short-term CR) for the following 21 minutes (7 periods of 3 minutes).

Unfortunately, due to the problem structure, finding the exact optimum control policy at each time step is computationally intractable. Thus, we have to use a method that allows us to approximate this optimum policy. We choose for this purpose to use randomized optimization techniques. Randomized optimization algorithms are a very promising method in this context, since they can inherently deal with the complexity of the problem, with reasonable computational workload. There are several methods falling into this category, such as genetic algorithms, simulated annealing, etc. While all seem to work with more or less the same efficiency, only few have theoretical convergence to the optimum in finite time. This is the reason we chose the

method described in [23]. This method is a variation of Simulated Annealing that works both for deterministic and expected value criteria.

The concept behind this randomized optimization algorithm is that, while randomly searching and trying to find the minimizer of the cost function, from time to time, one may accept a worse solution (instead of accepting only better solutions). This helps the algorithm overcome local minima and continue exploring the space. Details on the method we use for this can be found in [23], as well as specific description of its application in the conflict resolution problem in [9, 34].

Since the unicycle dynamics used by the Navigation Functions can only be considered as an abstraction for real aircraft dynamics, we employ a more realistic autopilot module, converting the Navigation Functions commands to the appropriate variables in the aircraft dynamics. The dynamics of the aircraft follow those presented in [7]. The linear velocity commanded by the Navigation Functions is used as the nominal airspeed that the autopilot has to track, applying the thrust required, while the angular velocity is used for the bank angle control of the aircraft. This model hierarchy is depicted in Figure 2. Further details on this hierarchical approach can be found in [10].

### 3.3 Decentralized strategy

As no ground support is present in the A<sup>3</sup> ConOps [13], the aircraft should be able to identify and resolve all situations that might evolve into a conflict. For this to be possible, we assume that the intent of all aircraft is communicated between them at the Mid Term, see Figure 2.

One immediate way to decentralize the scheme proposed is to have each aircraft try to find an optimal route, such that it does not enter into the protected zone of all other aircraft, while respecting constraints that might be present in the situation. In this case, all aircraft will start with an initial centralized solution. Then at the next time step, each aircraft will have to assume that the already existing solution for all other aircraft is fixed and will not be changed in the near future. This though is very conservative and very frequently leads to infeasibility (in more than 80% of the cases the algorithm was not able to find a solution); as more information will be available, better solutions can be found at later times and as a result other aircraft may also decide to recalculate their solutions. In the approach described above though this is not taken into account.

Another approach is to assume that aircraft solve their trajectories sequentially in a round-robin fashion, i.e. after all aircraft have found a solution, they solve the problem in the next round - after some minutes - in the same order. This approach which is also employed by the MMPC methods in Section 3.1 can be seen as an implicit way to define priorities, giving aircraft in the beginning of each resolution round right of way and more freedom to choose their trajectories. In this case the first aircraft will find a solution that minimizes only its cost function. Then, the first aircraft will broadcast the solution and this solution will be considered as a constraint by the second aircraft. This will proceed until one round of solutions is found and the next round starts again from the first aircraft.

One can reasonably argue that following such a decentralized policy may lead to aircraft with high priority (i.e. the first few aircraft to decide at each round) having an advantage over the remaining aircraft, who might have to do much larger maneuvers to avoid conflicting situations. There are mainly two ways to avoid such a situation; either the sequence that aircraft decide on each round could be random or a “fairness” factor can be entered in the cost function of the first aircraft such that they do not choose maneuvers that may result in such situations. We elaborate more on those two ways of dealing with this in Section 3.4.

### 3.4 Simulation setup and Results

**Simulation Setting** In our simulation setting, we consider several aircraft at the same flight level, performing a level flight, converging to the same point, denoted by  $(0,0)$  in Figure 6, that have to be deconflicted.

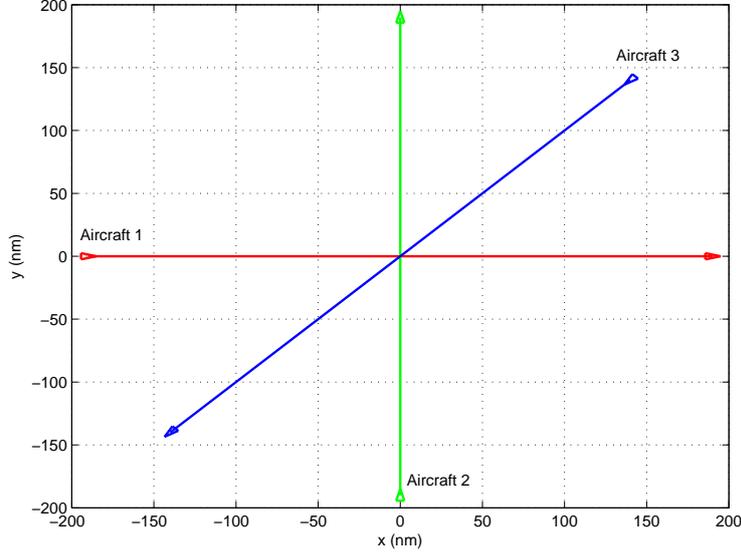


Figure 6: Configuration for 3 aircraft encounter.

We will assume for our simulations that the aircraft are of type Airbus A321, flying at 33000ft, a typical cruising altitude for commercial flights. [7] suggests that the airspeed at this altitude can only vary in the region  $[366, 540]$  knots, with a nominal value of 454 knots. We will enforce these constraints on our controller.

Regarding the uncertainty, we will only consider the wind speed as source of uncertainty. Wind speed (in general) can be modeled as a sum of two components: a nominal, deterministic component (available through meteorological forecasts) and a stochastic component, representing deviations from the nominal. Since the forecasts are available prior to the flights, flight plans are calculated taking them into account, so for simplicity reasons, we set the forecasted wind speed equal to zero. The structure of the forecast inaccuracies is modeled according to [12]. As wind is a source of uncertainty in our system, the algorithm will produce a different set of trajectories for the aircraft for each different wind realization in the system. For demonstration purposes, we only plot one case for each variant of the proposed scheme.

**Fixed Manoeuvre computation sequence** We first consider the case where the aircraft decide their actions at each round according to a fixed sequence, as in round-robin algorithms. The cost function used is the distance of each aircraft from the final destination at the end of the mid term conflict resolution algorithm, i.e. after 21 minutes. The trajectories that the aircraft need to fly in this case are plotted in Figure 7. For comparison purposes, we also include in Figure 8 the trajectories that a centralized conflict resolution algorithm would suggest. We should stress here that this version of the algorithm is compatible with the ConOps requirements, as high priority aircraft maneuver only when there is no other way to respect the constraints, while maintaining self separation. Of course, as in our case, the conflict is bound to happen between more than one aircraft at the same time, it is not possible to resolve it just by having

one aircraft maneuver, but more are needed to replan their trajectories. If this algorithm is tested against a scenario that only pair-wise conflicts occur, then only the lowest priority aircraft will maneuver to resolve each conflict.

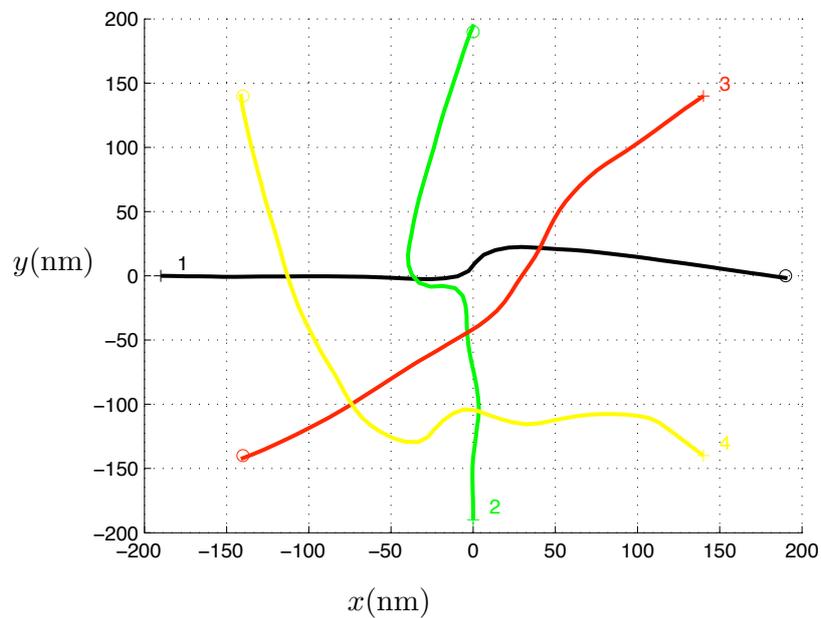


Figure 7: Aircraft trajectories for round robin decentralized conflict resolution

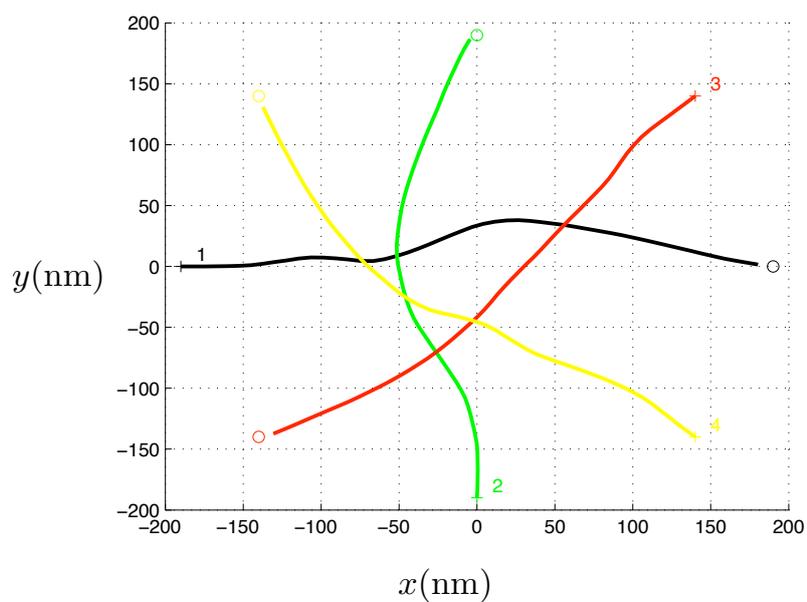


Figure 8: Aircraft trajectories for a centralized conflict resolution

A very important fact is that decentralizing the proposed conflict resolution scheme does not affect the feasibility of the traffic situation, as all cases that could be solved by a centralized algorithm can also be solved in a decentralized fashion. The plots indicate that all aircraft reach their destinations, despite the presence of uncertainty and the “mismatch” between the model used by the Navigation Functions and MPC to resolve the conflicts with the aircraft autopilot.

Comparing now the two different solutions, one can observe the fact discussed in 3.3; in the decentralized scheme, high priority aircraft are clearly favored, being the first to plan their trajectories at each round. Despite the fact that three of them have a quite smooth trajectory to fly, the next one (the last to choose at each round) is forced to perform a very costly maneuver, having to avoid all the others.

**Random sequence** Next, we randomly choose a different sequence of aircraft at each decision round, according to which they will calculate and broadcast their intended trajectories. It is important to note that in our setting this also retains the feasibility properties of the original centralized problem; as long as the centralized conflict resolution can find a solution for the situation, the decentralized will also produce one.

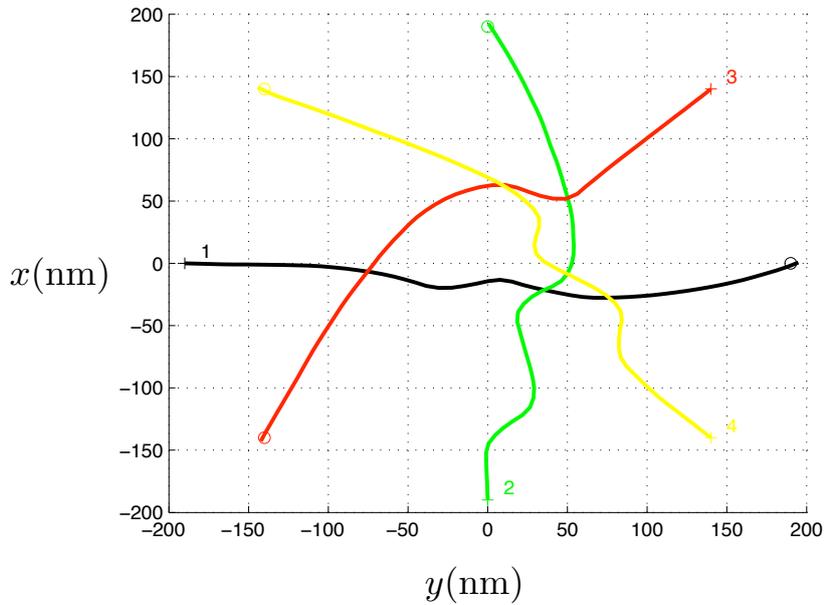


Figure 9: Aircraft trajectories for random order decentralized conflict resolution

Figures 9 and 10 display the simulation results in this specific case for two different random sequences. In this case, an aircraft might start with a high priority, deciding early in the round, but then some other aircraft may gain priority, forcing it to cover a much bigger distance until the destination. Depending on the different random sequence that aircraft decide, this can lead to only a few aircraft being affected, or in some cases even all aircraft might have to follow a longer trajectory.

The purpose of this simulation is to show an alternative to fixed priorities in the case that the trajectories produced are not satisfactory. In a similar fashion (i.e. changing the priorities)

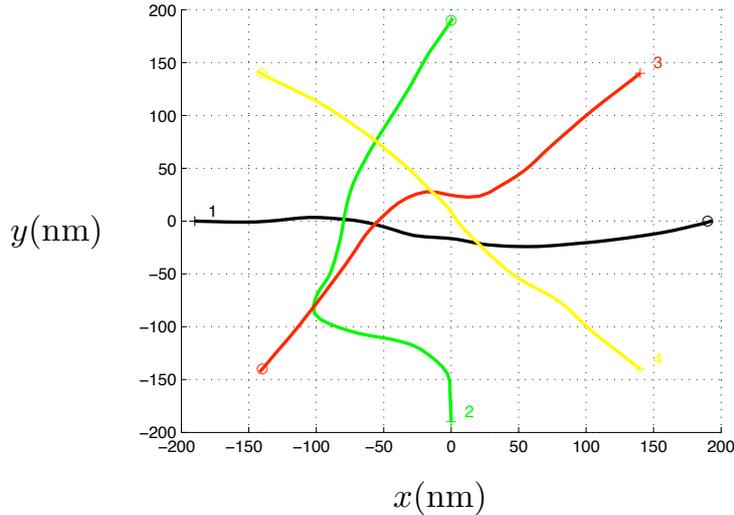


Figure 10: Aircraft trajectories for random order decentralized conflict resolution

we are planning to investigate situations like priority reversal, where aircraft will start planning in the reverse order (instead of a random one).

**Cooperative cost** As another alternative to random priorities, we consider the case where the Mid Term algorithm couples the decentralized systems also through the cost. The cost we will consider in this case is again only terminal (i.e. only at the end of the prediction horizon of the algorithm), but we introduce a “fairness” factor  $\alpha$  to take into account the effect that the solution of one aircraft has on the others. Then, the cost for each aircraft will take into account the costs incurred for the subsequent aircraft in each decision round, multiplied by a depreciation factor  $\alpha$ . We only take into account the effect to the aircraft next in the decision round, as previous aircraft have already announced their solutions. It is easy to see that setting  $\alpha = 0$ , aircraft solve the problem as in the previous cases, while  $\alpha = 1$  makes the first aircraft at each round to solve exactly the centralized problem. In such a way, the priorities can be accommodated, while at the same time, high priority aircraft try to find solutions that will not result in a very costly trajectory for lower priority ones.

Figures 11 and 12 show the trajectories the aircraft follow solving the problem both with a fixed as well as a random decision order at each decision round, with  $\alpha = 0.4$ . One can observe in both cases that there is no aircraft clearly favored by such a scheme, regardless of the order that the decisions are made in each round. Trajectories though in a random order of decision scenario seem much smoother, very similar to a centralized solution. Setting different values of  $\alpha$  defines the trade-off between preserving the priorities and suboptimality of aircraft at the end of each optimization round.

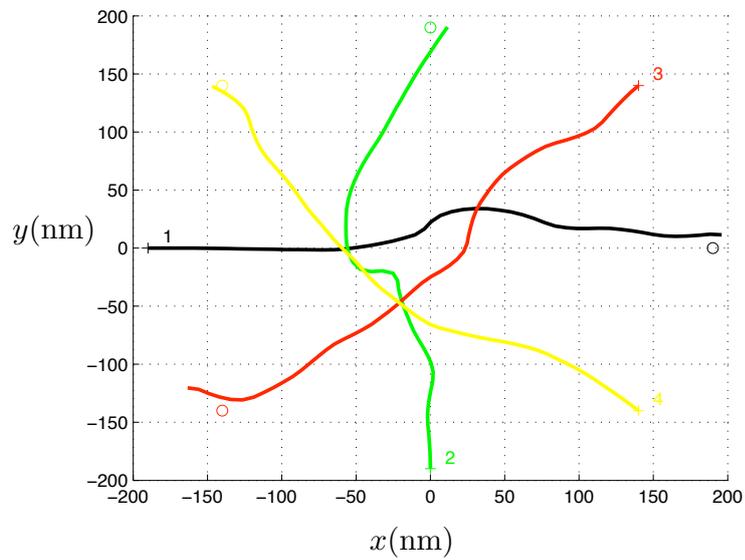


Figure 11: Aircraft trajectories with  $\alpha = 0.4$  for fixed order decentralized conflict resolution

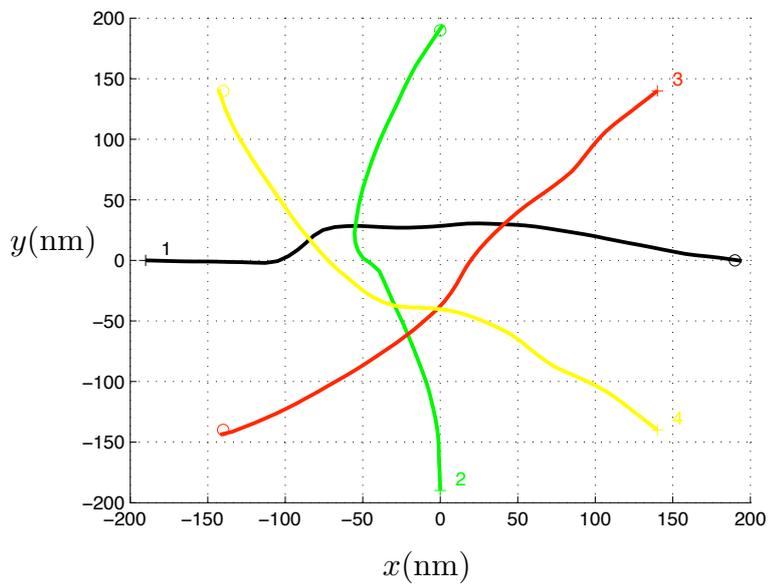


Figure 12: Aircraft trajectories with  $\alpha = 0.4$  for random order decentralized conflict resolution

## 4 Mid term conflict resolution: Distributed Robust Multiplexed Model Predictive Control

In this section, we summarise findings from application of robust multiplexed Model Predictive Control (MPC) (MMPC) to mid term conflict resolution. We use the term multiplexed MPC to refer to the general class of distributed MPC schemes that we detail in this section, and the original MMPC method is the scheme presented in [24]. An outline of the section is as follows. We first summarise briefly in section 4.1 the original multiplexed MPC algorithm presented in [42]. Details of the problem formulation and key assumptions made are then outlined in section 4.2. To overcome the limitations in obtaining robustly feasible solutions on application of the original robust MMPC algorithm to typical mid term conflict resolution scenarios, we present three further variants of the original decentralised robust model predictive control algorithm presented in [42]. The first variation involves exploiting the temporal correlations present in the wind disturbances wherein the state is augmented with the disturbance model, as outlined in section 4.4. The two key MMPC algorithms identified as most suitable for the mid-term conflict resolution problem, belonging to the class of multiplexed algorithms with disturbance feedback are detailed in 4.3.

We discuss the scope of the resolution algorithms in coping with situations in which aircraft enter and leave the scenario in section 4.5, and present initial findings on application to the air traffic simulator in section 4.6.

### 4.1 Recap of Robust Distributed MMPC

Implicit in the class of distributed MPC schemes belonging to multiplexed MPC is a notion of sequence, according to which aircraft update their planned trajectories. The sequence is determined by a single aircraft and broadcast using System Wide Information Management (SWIM).

The determination of the aircraft update sequence is addressed in in Section 4.3.2, where we present a heuristic in the form of Variable Update Order MMPC (V-MMPC). The issue of determining the optimal sequence remains to be explored.

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. Each aircraft executes its planned policy until it is its next turn to replan. In the original multiplexed formulation, as proposed in [24], changes in heading and speed are applied at periodic intervals, with no changes in-between updates.

To enable availability of plan over a future horizon whose length coincides with the time-scale implicit in ‘mid-term’ conflict resolution, it is necessary to make some modifications to the original multiplexed formulation presented in [42]. Robust recursive feasibility means that an initially feasible input sequence guarantees that all subsequent optimisations will be feasible, despite the presence of uncertainty in the dynamics. Recall that to achieve robust recursive feasibility, that is, to ensure that solutions continue to exist despite uncertainty in future positions of the aircraft, the algorithm uses nominal prediction models with tightened constraints. In the conflict resolution context, tighter constraints correspond to enlarged regions of avoidance and narrower bands of allowable speed bounds. Whilst this is not explicitly mentioned in the  $A^3$ Conops, this approach is consistent with it. The extent to which the constraints are tightened depends on the maximum disturbance level, and disturbance feedback policy employed.

## 4.2 Problem Formulation and Assumptions

We present now the problem formulation and outline the key assumptions required in the distributed MPC formulations considered. We require the use of the following definition for our specific problem formulation.

**Definition 4.1.** *A 3D point on the 4D-Reference Business Trajectory (RBT) is a geographical waypoint with the time component removed.*

We require also the following definition concerning identification of the area within which a potential conflict can occur, corresponding to a look-ahead time horizon of the order of tens of minutes. We consider level flight only in this work.

**Definition 4.2.** *The region of interest is a (planar) rectangular region at one flight level which can vary in size between  $120 \times 120$  and  $200 \times 200$  nautical miles and contains a number of aircraft.*

To date, we have not considered a number of aircraft greater than 6. Aircraft are required to traverse the region of interest and reach their specified target areas as quickly as possible and within some bounded time. These target areas can be viewed as relaxed waypoints, typically centred on a point on the 3D-Reference Business Trajectory (RBT) on the boundary of the region of interest, acting as an exit gateway. We require the following assumption:

**Assumption 4.1.** *Aircraft are aware of the presence of all other aircraft involved in their region of interest considered. Additionally, they have access to their current states.*

The boundaries of the region of interest and target regions are possibly identified and prespecified by the conflict detection algorithm. The target regions are assumed to be ‘gateways’ for aircraft to rejoin the 3D-RBT once the conflict resolution manoeuvre has been executed. The proposed methods provide an upper bound on the time at which the target regions should be reached, and an exact time at which the 3D-RBT is rejoined is not given. The robust distributed MPC algorithms considered require that all aircraft in the region of interest have full knowledge of the positions of all the aircraft, and the disturbances they experience. We therefore require the following additional assumption:

**Assumption 4.2.** *The transmission of aircraft positions, wind disturbance measurements<sup>1</sup> future plans to neighbouring aircraft is enabled by SWIM, as proposed in the A<sup>3</sup> Conops concept.*

**Assumption 4.3.** *We assume there are no time delays associated with SWIM. In the event of communications failure, aircraft can execute their disturbance feedback control policies based on the last time they planned their trajectories.*

Since we assume a sampling interval of 1 minute, time delay in SWIM can be neglected in initial investigations.

## 4.3 MMPC with disturbance feedback control

In the variant of multiplexed MPC we consider now, aircraft are no longer constrained to execute moves at predefined intervals according to a fixed sequence, rather, changes in speed and heading can be applied every timestep. The scheme involves a single aircraft optimising its 4D trajectory at any time. In-between optimisation updates, aircraft apply a fixed feedback control policy

---

<sup>1</sup>Wind disturbances can be inferred by evaluating the difference between the aircraft airspeed and groundspeed, both of which are measured on board.

according to the disturbances they encounter and update their 4D trajectory plans. This modified scheme permits longer prediction horizon lengths than the original MMPC due to the improved disturbance rejection. Furthermore, the constraint that aircraft update their feedback control policy according to a fixed prespecified order is relaxed. This property is crucial for dealing with situations in which aircraft appear and disappear from scenarios, which is discussed later. The effect of disturbance feedback is to reduce the impact of the future unknown disturbances, with the choice of a suitable feedback policy. The form of the disturbance feedback policy is given in Appendix C.5, and an explanation of how applying disturbance feedback reduces the conservatism of the constraint tightening is provided in Appendix C.5.3.

#### 4.3.1 Fixed order MMPC with disturbance feedback control

Figure 13 shows a representative example of this MMPC method, again with three aircraft initialised at 454 knots, with a fixed cyclic updating order  $\{1, 2, 3\}$ . We choose a 2-step nilpotent feedback policy, which means constraint tightening is only required for the first two steps in the prediction horizon, and consequently enables the use of a prediction horizon of any length. The aircraft are initialised fairly with trajectories with equal open loop cost trajectories which are plotted in Figure 13(a). In this example a maximal horizon length of 30 minutes is employed, with a predicted time to loss of separation of 15 minutes. It can be seen that safe separation is maintained throughout the simulation, and that the speed constraints are respected.

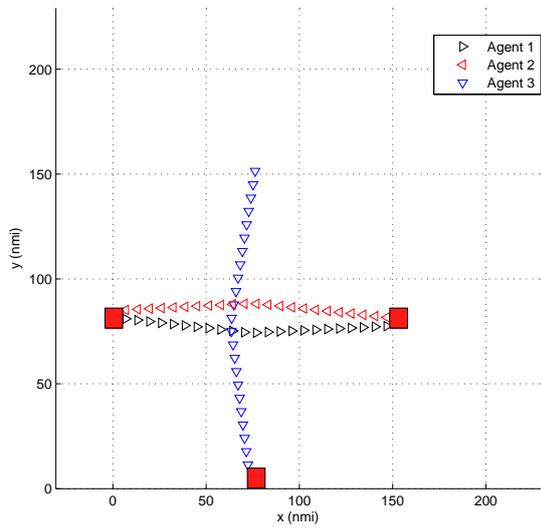
### Communication Requirements

As stated earlier, wind disturbance observations are required to be broadcast by all aircraft at every timestep. Aircraft need only broadcast their intent every time they reoptimise, which would be every  $m$  timesteps, with a system of  $m$  aircraft<sup>2</sup>. Given that global knowledge of the individual aircraft feedback gain matrices is assumed, the optimising aircraft can construct the predicted plans of the other aircraft when it plans its own set of moves.

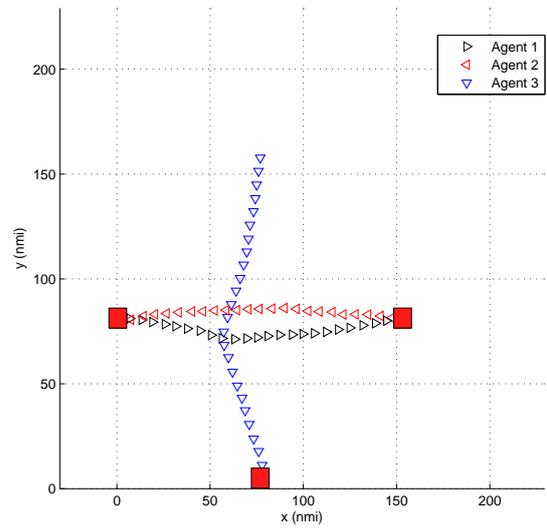
#### 4.3.2 Variable update order MMPC with disturbance feedback control

As stated earlier, with the MMPC with disturbance feedback in between updates, there is no requirement on the policy update order of aircraft. At this juncture we should mention the concept requirement that only lower priority aircraft change their plans to resolve the conflict if possible, whilst high priority aircraft retain their original plans, unless the conflict configuration renders this impossible. According to the concept then, this would mean that when determining the policy update sequence, only the lower priority aircraft would be included in the set of aircraft updating their policies. In this subsection we propose a variable update order formulation, whereby aircraft optimise in parallel for new policies, but a decision on the aircraft changing its policy at any one time is based on satisfaction of some global objective. Each aircraft optimises for a new plan, conditioned on the other aircraft executing their candidate feasible plans. The motivation for permitting variable order is to allow aircraft with ‘greatest need’ to reoptimise their policy, to respond to strong disturbances for instance. In the implementation we have considered, the updating aircraft is chosen to be that which would yield the maximum reduction in the global cost, given that only one aircraft is permitted to execute its newly optimised plan. The specific details are given in Appendix C.5.2. This scheme makes maximum use of the time between updates and incurs no additional computational cost. There is however an increase in communications overhead, as each aircraft would be required to broadcast its reduction in local

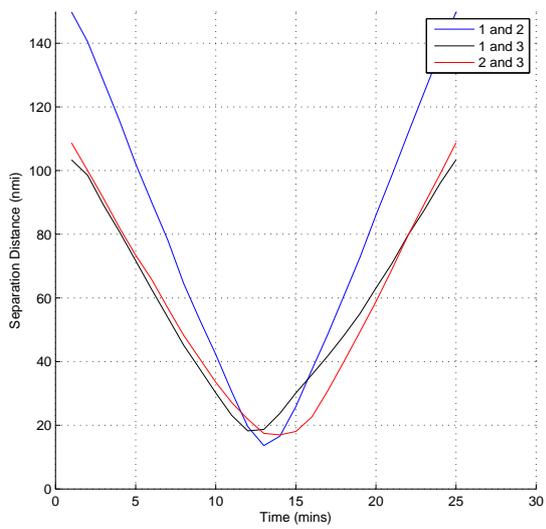
<sup>2</sup>This is because we assume that at each timestep only one aircraft optimises and that aircraft optimise in a fixed sequence.



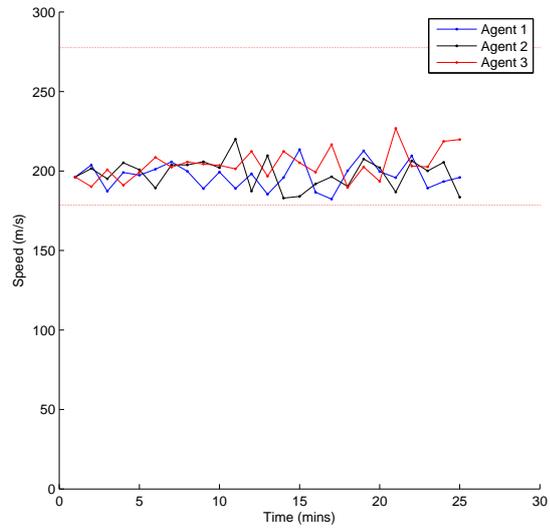
(a) Centralised



(b) MMPC with disturbance feedback control



(c) Separation/nmi



(d) Speed m/s

Figure 13: Fixed Order MMPC

cost, if it were to execute its newly optimised trajectory. This is however minimal, compared with the communications overhead associated with communicating wind disturbances.

### Communication Requirements

Wind disturbance observations are required to be broadcast by all aircraft at every timestep. As the optimisation order is no longer necessarily cyclic, in the worst case, in addition to the wind disturbances, intent is also broadcast at every time step. Figure 14 shows the mean total

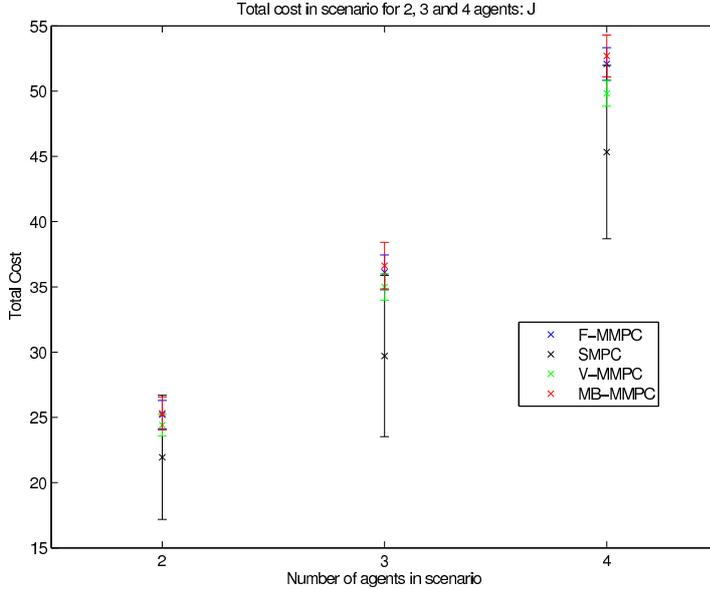


Figure 14: Mean Closed loop cost over 128 simulations with error bars displayed for the different MMPC schemes : Fixed Order MMPC (F-MMPC); Synchronous (Centralised) MPC (SMPC); Variable Update Order MMPC (V-MMPC) and Move-Blocking MMPC (MB-MMPC).

closed loop cost obtained over 128 simulations for each of the disturbance feedback multiplexed schemes implemented. We consider also an input ‘move-blocking’ variation of the fixed update order MMPC, termed MB-MMPC, where the predicted control actions are held constant over sets of multiple prediction steps. For a fixed horizon length  $N$  and a number of aircraft  $m$ , the proposed move blocking scheme involves optimisation over  $\frac{N-1}{m}$  inputs per aircraft, compared with  $N$  inputs for every receding horizon optimisation in the schemes F-MMPC and V-MMPC. A finer discretisation is employed to enable a higher frequency of policy update, with no resulting increase in computational cost. Whilst there is minimal variation between the multiplexed schemes, it can be seen that the variable update order scheme V-MMPC approximates the optimal centralised policy SMPC most closely.

### 4.4 Exploiting Disturbance Correlations

It has been observed in simulation that for typical levels of disturbances experienced and lengths of planning intervals employed in mid-term conflict resolution an initially feasible solution frequently does not exist. Specifically, the condition in (66) presented in Appendix C.5 for obtaining an initially feasible solution is not met. We counter the issue of difficulty in obtaining an initially feasible solution by incorporating a deterministic component into the wind model, and exploiting the temporal correlation of the wind; as meteorological predictions are known in advance, the wind velocity can be modelled as a sum of a deterministic disturbance and a

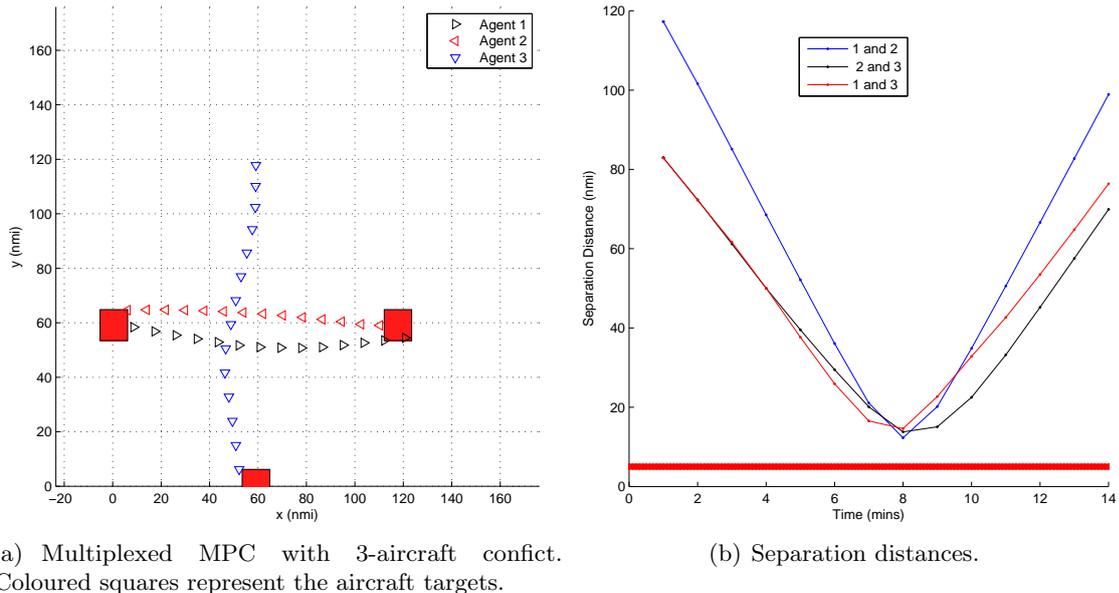


Figure 15: MMPC with wind correlation model

stochastic component corresponding to unknown deviations from this nominal prediction [3]. The wind experienced by any aircraft at a given time is correlated to the wind it experienced at earlier times. This is incorporated in the model, and the wind disturbance is included as an additional state variable. As the deterministic offset has now been removed, and introduced as an additional state variable, the bounds on the unknown disturbance levels are now reduced. This permits horizon lengths of up the order of 20 steps ahead of the current time to be used, when steps of interval one minute, ie the discretisation length, are employed. Specific details of the formulation are given in Appendix C.4.

Figure 15 demonstrates the results obtained for three aircraft with the formulation incorporating the temporal wind correlation. In this MMPC formulation, the prediction horizon length is required to be of the form  $mp + 1$ , where  $m$  is the number of aircraft, and  $p$  is the number of control updates planned over each aircraft’s planning interval. In the example presented, the 3 aircraft plan for 7 moves each, resulting in a prediction horizon length of 22 minutes. The length of the prediction horizon is an upper bound for the time in which the aircraft are required to have reached their respective target regions. In all the simulations considered here, we assume the aircraft are of type Airbus A321, and fly at 33000ft, a typical cruising altitude for commercial flights [3]. It is suggested in [7] that the airspeed at this altitude varies between [366, 540] knots, with a nominal value of 454 knots, and we enforce these speed constraints and minimum horizontal safe separation of 5 nmi.

In order to demonstrate the capabilities of the algorithm, we test it under conditions more extreme than likely to be encountered in practice, and consider initialisation conditions which would result in loss of separation, and convergence of the aircraft to the same point at the same time. The aircraft are initialised at 454 knots each, and a predicted time to loss of separation of 8 minutes.

The points displayed in Figure 15(a) are aircraft positions plotted at equal 1-minute intervals. Figure 15(b) shows the pairwise separation between all three aircraft. The minimum safe separation of 5nmi is displayed as a dotted red line. It is clear that, by interpolating between the points, the minimum safe separation constraint is not violated. For the centralised solution, the initial wind disturbances for each aircraft are drawn from uniform distributions on  $[-5.17, 5.17]$ m/s

[3]. In subsequent distributed optimisations, the stochastic component is drawn from a uniform distribution on  $[-0.33, 0.33]$ m/s.

The conservatism of the original robust multiplexed approach can be further reduced if, instead of open loop control between successive updates, we permit aircraft to execute a fixed feedback policy in between updates to correct for the disturbances they experience. In the original multiplexed formulation, for a fixed horizon length and sampling time, the number of changes in heading and speed made varies inversely with number of aircraft, limiting the number of aircraft that can exist in a scenario. Although this offers computational benefits, with fewer decision variables per optimisation, this constraint is found to be restrictive, particularly when considering scenarios in which aircraft enter and leave the scenario. We next outline the details of an improved formulation in which this restriction no longer applies.

## 4.5 Aircraft Entering and Leaving the Region of Interest

We now consider a scenario in which aircraft enter and exit a potential conflict region. We seek to obtain solutions which enable robust completion guarantees, namely that in the presence of uncertain wind disturbances, aircraft are guaranteed to reach their prespecified target regions in a bounded length of time. Solving for a centralised solution to re-initialise the distributed optimisation scheme to accommodate the new aircraft would incur significant computational cost. To avoid this, we have therefore considered distributed formulations which utilise the predicted plans made by the aircraft in the original scenario prior to the entry of newcomers. It is therefore necessary to make the following assumption:

**Assumption 4.4.** *On entry of an 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.*

The trajectory of the incoming aircraft may have higher associated costs as the plans of the original aircraft have to be accommodated whilst searching for a feasible solution.

We next present an initial evaluation of the capabilities and potential of the robust MMPC formulations described, namely the original MMPC, and the fixed order and variable update order variants of the robust MMPC with disturbance feedback formulations for resolving conflicts in this setting.

### 4.5.1 MMPC

We first consider the situation where an aircraft reaches its target regions and is considered to exit the region for which the scenario has been defined. In the original multiplexed formulation, any feasible solution sequence of moves obtained for the system of aircraft is subject to the constraint that the aircraft execute their moves according to a prespecified timing sequence. Once an aircraft has reached its exit target region, the slot it previously occupied in the update and optimisation sequence becomes free. Given an initial updating sequence  $\{1, 2, 3, 4\}$ , once an aircraft, say 3, has left, a feasible set of moves at the next time step can be constructed from the previous solution for the update sequence  $\{1, 2, -, 4\}$ . The optimisation problem becomes less constrained, but the free slots cannot necessarily be occupied by one of the remaining aircraft without reoptimisation over all the remaining aircraft. Equally, when new aircraft appear, they cannot slot themselves into the updating sequence, without reoptimisation. To overcome this we have considered a few options.

One option would be for new aircraft to share updating slots with existing aircraft, in which new aircraft ‘double up’ with existing aircraft, with whom they share a slot in the updating

sequence. With the addition of a fourth aircraft to a scenario of 3 aircraft for instance, the update sequence could become  $\{(1, 4), 2, 3\}$ . aircraft 1 and 4 would then solve for a joint solution. Clearly to keep the computational complexity low, it is desirable that the number of aircraft joining at any time is kept low.

Another alternative would be to extend the prediction horizon and assign the new aircraft a number of moves at the end of the prediction horizon. This idea is explored with application to the MMPC with disturbance feedback formulations.

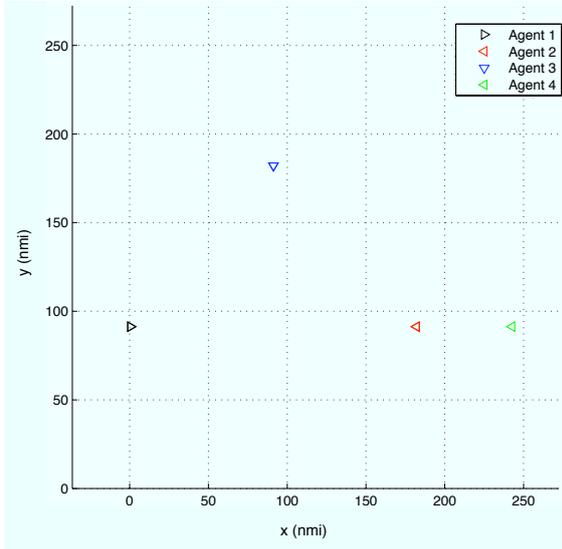
#### 4.5.2 Robust MMPC with Disturbance Feedback

The MMPC with disturbance feedback algorithm detailed in 4.3 is more amenable to performing conflict resolution in instances where the number of aircraft in the scenario is not fixed. We require the use of Assumption 4.4 however, so that when a new aircraft enters, a feasible solution for it exists. Providing such a solution exists, the recursive feasibility properties are retained. This assumption can be satisfied if we exploit the variable horizon feature of our MPC formulation by extending the prediction horizon of the new aircraft by an arbitrary length. If there are no time constraints on the arrival time of the new aircraft at its exit gateway, we can make the maximal prediction horizon as long as we desire, subject to computational constraints; by increasing the control horizon the domain of attraction of the target sets is enlarged, but at the expense of a greater computational burden, which would limit the number of aircraft that can enter a scenario.

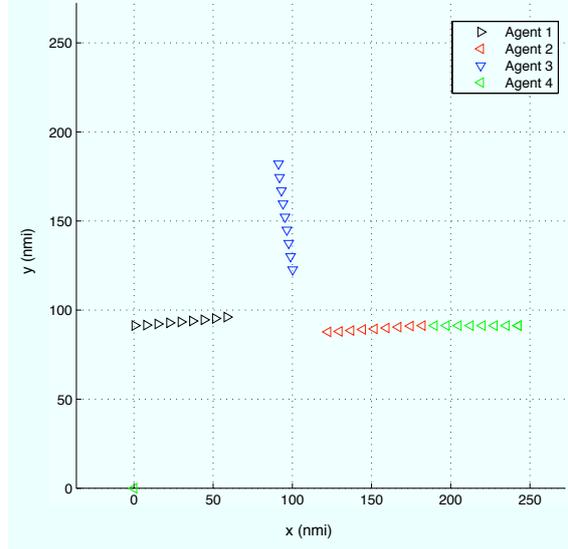
Figure 16 shows the results of a scenario in which a new aircraft enters a region of interest. The original three aircraft involved in the initial encounter are initialised on the boundaries of the region of interest, as in the previous three-aircraft encounters. They update their plans in the cyclic sequence  $\{1, 2, 3\}$  using prediction horizons of length 30 mins and predicted time to collision 15 minutes, unaware that a fourth aircraft flying East-West at 454 knots will enter the region of interest after 8 minutes. The new aircraft is not aware of the region of interest until the point at which it enters the region. As the newcoming aircraft, it is assigned highest priority and upon entry to the alert zone, optimises for a sequence of moves. Its optimisation is constrained by the candidate predicted plans of the other aircraft. Aircraft 2 has to miss its turn to reoptimise for a new plan, and instead executes affine disturbance feedback, the first move of a feasible candidate sequence obtained from the previous timestep, as do 1 and 3. The optimisation order now becomes  $\{1, 4, 2, 3\}$ , so the cycle time increases by one with the addition of every new aircraft. The entry of aircraft 4 constrains the possible actions of the other aircraft in subsequent optimisations. To see this, consider the results of the static scenario in figure 16(d), which depicts the trajectories of the original three aircraft subject to the same sequence of wind disturbances. The closed loop cost of the original trajectories is lower in the static case, with all targets being reached in 2 fewer steps. The trajectories for the fixed number of aircraft scenario have lower cost as the frequency of optimisation update is higher, with therefore improved response to disturbances. In figure 16(c), which depicts the full trajectories of all four aircraft, it can be observed that the newcoming aircraft's path has the greatest cost, as expected. The other aircraft have no incentive to sacrifice their local performance to allow benefit for the new aircraft.

## 4.6 Simulation Results

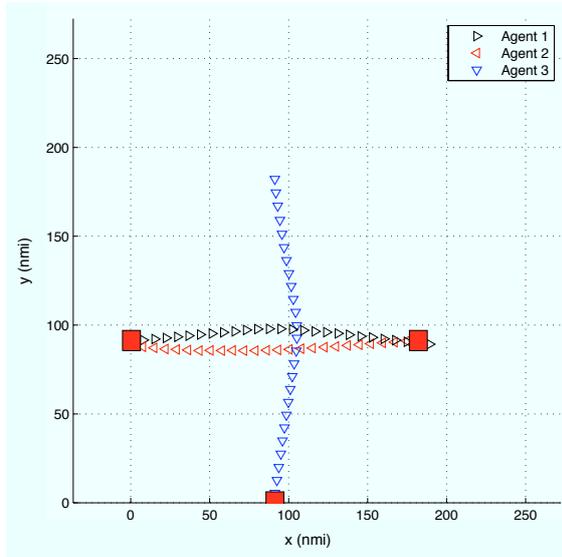
We now present results obtained from application of the fixed order MMPC algorithm with disturbance feedback in a realistic Air Traffic Control (ATC) setting, using a simulator developed by [26]. The ATC model is hybrid, with continuous dynamics arising from the aircraft dynamics, and discrete dynamics arising from the flight plan and Flight Management System (FMS). A



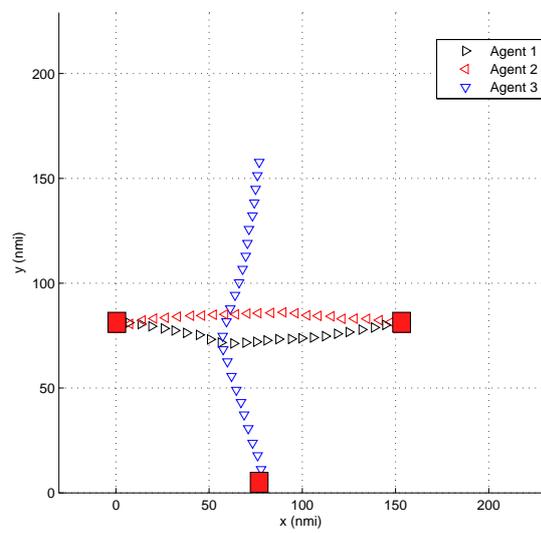
(a)  $t=0$



(b)  $t=8$  mins



(c)  $t=33$ mins



(d) Fixed scenario with 3 aircraft

Figure 16: Fixed Order MMPC

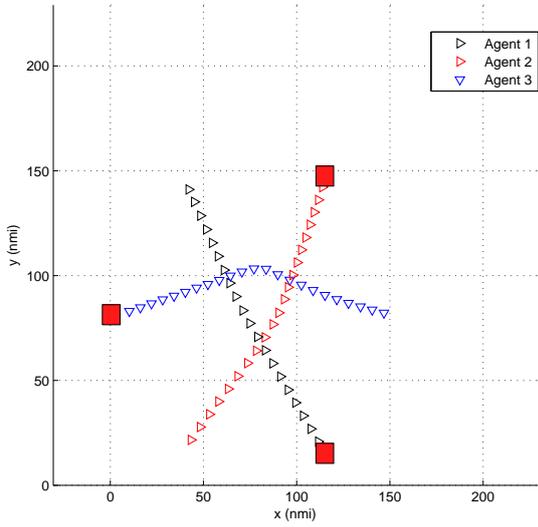
point mass model is used for the aircraft based on the Base of Aircraft Data (BADA) database. The effects of unpredicted wind disturbances are included in an internal wind correlation model. The FMS controller is modelled as a 3D -FMS, where along-track errors are neglected. Further details of the simulator can be obtained from [26]. The MMPC assumes the role of the ATC, providing changes in the flight plans with the evolution of the flights.

The flight plan comprises a reference path of straight lines, formed from a sequence of waypoints and a sequence of airspeeds. At each timestep, the MMPC algorithm is executed to produce a waypoint obtained from the state predictions. This waypoint is used as a flight plan input to the simulator. The aircraft state evolves according to the dynamics incorporated in the simulator model, and the state measurement is used as input to the MMPC, which then produces a waypoint for the next time step. The procedure is repeated until all aircraft have reached their target destinations. The along-track errors are bounded at 1km per minute, and are accounted for with the wind velocity disturbances in our robustification.

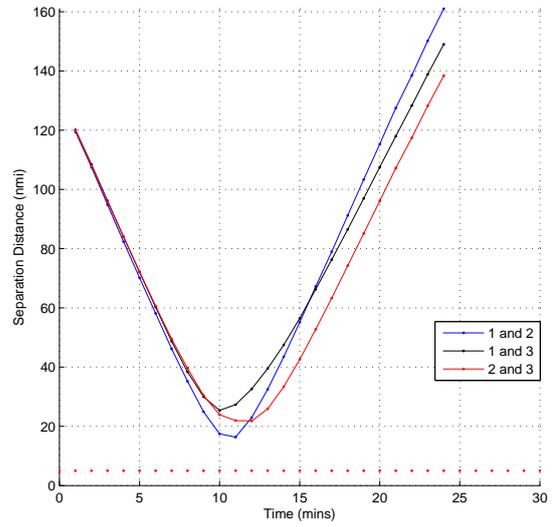
100 simulations were performed with different wind fields for systems of three and four aircraft. Statistics of the results obtained are summarised in Table 1. The minimum separation requirement is maintained in all cases, and flyable trajectories are produced. We present representative plots of trajectories obtained with two aircraft and three aircraft in Figure 17:

	<b>Three aircraft</b>	<b>Four aircraft</b>
Mean minimum separation (nmi)	15.1	14.6
Variance minimum separation (nmi)	0.39	0.58
Minimum separation over all simulations (nmi)	12.1	12.6
Mean time per MMPC stage (s)	16.7	17.0
Variance per MMPC stage(s)	3.80	3.97
Centralised Solution time (s)	77.8	657.6

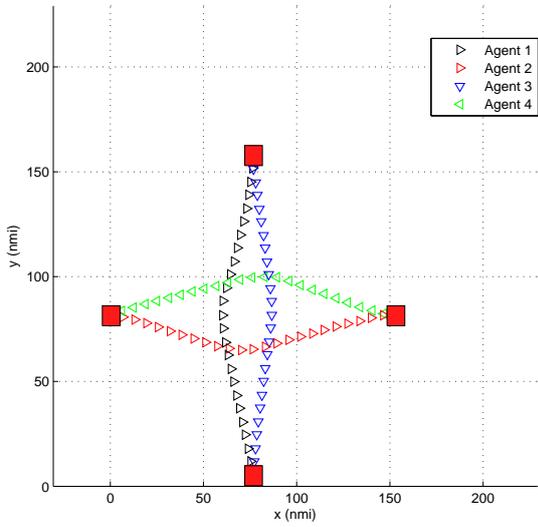
Table 1: Multiplexed MPC with ATC Simulator: Statistics summarising results obtained with 100 simulations performed with different windfields with 3 and 4 aircraft. The centralised solution time corresponds to that required to calculate a joint solution for all aircraft at the initialisation stage.



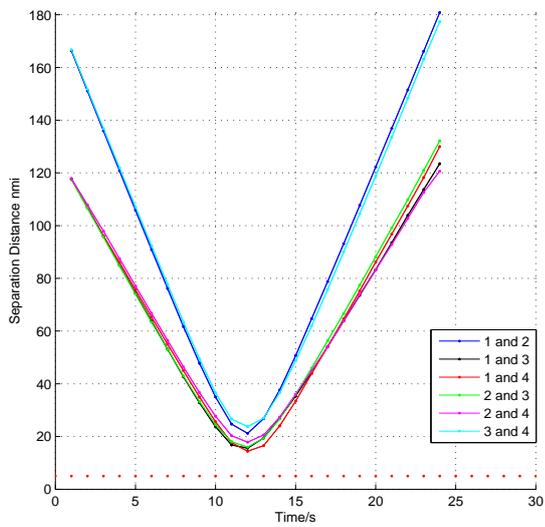
(a) Three aircraft conflict



(b) Separation/nmi



(c) Four aircraft conflict



(d) Separation/nmi

Figure 17: Fixed Order MMPC with ATC Simulator

## 5 Mid term conflict resolution: Hierarchical MPC with Priorities

The methods mentioned before can deal with priorities implicitly, by taking them into account in the cost function. The A<sup>3</sup> ConOps on the other hand demands for a more systematic way to deal with the priorities, in the sense that in medium term conflict situations, lower priority aircraft should maneuver first [13]. In order to accommodate this requirement, efforts were devoted into developing a novel method that can deal efficiently with this issue. In this section, the formulation of this method is outlined, and some initial results are presented. As we discuss later on, the method is currently centralized, though efforts on decentralization aspects are currently under investigation.

### 5.1 Introduction

This section presents a priority-based hierarchical Model Predictive Control solution to the mid term Conflict Resolution problem. First, the physical aircraft dynamics are abstracted to simplified ones. Then, a centralized model predictive controller that takes into account the physical limitations of the aircraft, such as input constraints and turning rates, as well as the minimum separation safety constraints among the aircraft is designed. The effects of bounded winds on the simplified dynamics at the optimization level are taken into consideration and it is shown how to exploit the spatial correlation in the wind statistics in order to reduce the conservatism in the separation constraints. The problem is further complicated by the additional requirement of respecting pre-assigned priority levels for various aircraft. Following the priority concept proposed in the A<sup>3</sup> ConOps (see [13]), higher priority aircraft only maneuver in cases that a maneuver by all lower priority aircraft is not adequate to resolve the conflicting situation. The obtained solution is pushed down the hierarchy onto the flight management system and autopilot, which generates the inputs to be applied to the aircraft.

At the highest level, a centralized optimization problem is solved that takes into account all these constraints and generates over a certain prediction horizon  $N$  an optimal set of inputs for each aircraft. With the use of linear dynamics, we are able to formulate a mixed-integer linear program (MILP) to be solved periodically. The integer part of the optimization problem arises because of the non-convex nature of the conflict avoidance constraints, the minimum bounds on the aircraft speeds, as well as the priorities assigned to aircraft. Despite the fact that MILP problems can scale very badly, in usual air traffic scenarios most of the integer variables are not adding any active constraints on the problem, thus keeping the computation speed in reasonable levels. Once the optimal input sequences have been generated for all aircraft, they are pushed down to the lower level in the hierarchy, namely the autopilot. The autopilot generates the appropriate inputs and applies them to a simulator of the actual aircraft dynamics. The optimization problem is then resolved periodically and applied in a receding horizon fashion. This hierarchical setup is illustrated in Figure 18.

The finite horizon optimization problem solved at each time step accounts for horizon lengths of around 20 minutes. As the actual nonlinear aircraft dynamics are influenced by the wind uncertainty, we need to design a controller that is robust against such uncertainties. Unfortunately, the wind included in the model used for the dynamics is unbounded, which renders robust optimization-based control design impossible. As such, we robustify the MILP formulation against most situations, as discussed in the subsequent sections. The use of the correlation structure of the wind is taken into account to reduce conservatism in the problem.

In this section, we present the results of this formulation. All the details are provided in Appendix D.

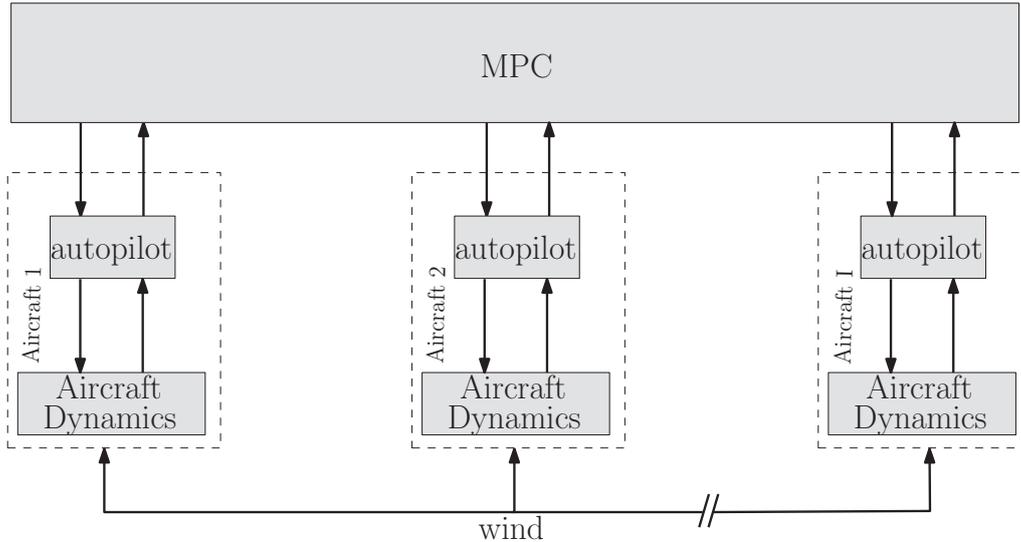


Figure 18: Hierarchical Multi-Level System

## 5.2 Simulation Results

### 5.2.1 Simulation Setup

We constructed a symmetric conflict situation in which six aircraft are initially located on a circle of 300 km radius and are heading to mid-air collision, as shown in Figure 19. The flight plans of all aircraft would lead them to the same point, 150km away from their starting position, at which they would all nominally arrive in around 11 minutes. Although very difficult to have in reality, this scenario allowed us to test the effectiveness of our method as it is quite difficult to tackle due to the symmetry and the closeness of the aircraft. In this setup, aircraft update their trajectories every 3 minutes and a horizon of 18 minutes that they can predict is assumed. In other words, every 3 minutes, aircraft plan their actions for the next 18 minutes, taking into account all other aircraft and uncertainties that enter the scenario up to that time. The conflict constraints are enforced every minute. We used the MILP solver CPLEX [20] through the interface package YALMIP (see [25]) for MATLAB for all simulations.

We compared 3 different scenarios for 1000 different wind realizations:

1. Running the proposed algorithm with the aircraft having different priorities and taking into account the correlated nature of the wind.
2. Running the algorithm with the aircraft having different priorities as in 1, but ignoring the correlation structure of the wind experienced by aircraft. In this case, the reported results are for 100 different wind realizations, as it requires a lot more computational time.
3. Running the algorithm in the case that all aircraft have the maximum priority level I and taking into account the correlated nature of the wind. Thus, the algorithm will first attempt to minimize the number of aircraft maneuvering and then the magnitude of the maneuver.

For visualization purposes, we plot the proposed resolution for the three scenarios 1, 2, and 3 for one wind realization in figures 20, 21, and 22, respectively. Comparing figures 20 and 21, it can be seen that taking into account the correlated nature of the wind improves the resolution in terms of extra flown distance needed for each of the aircraft to avoid the conflict. This happens

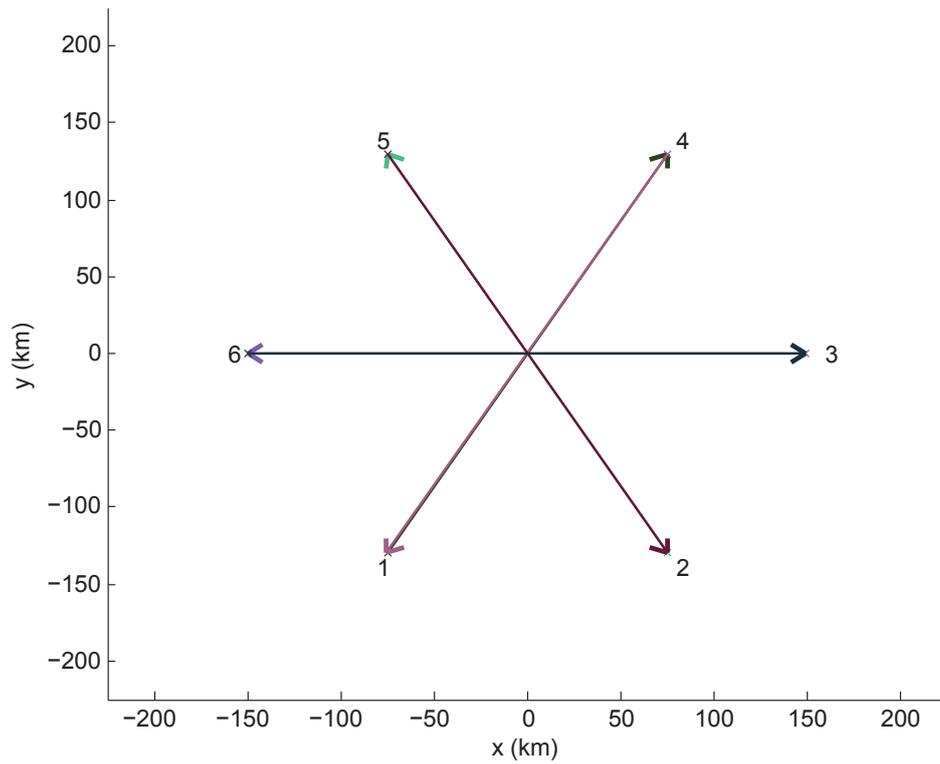


Figure 19: Conflict scenario with 6 aircraft

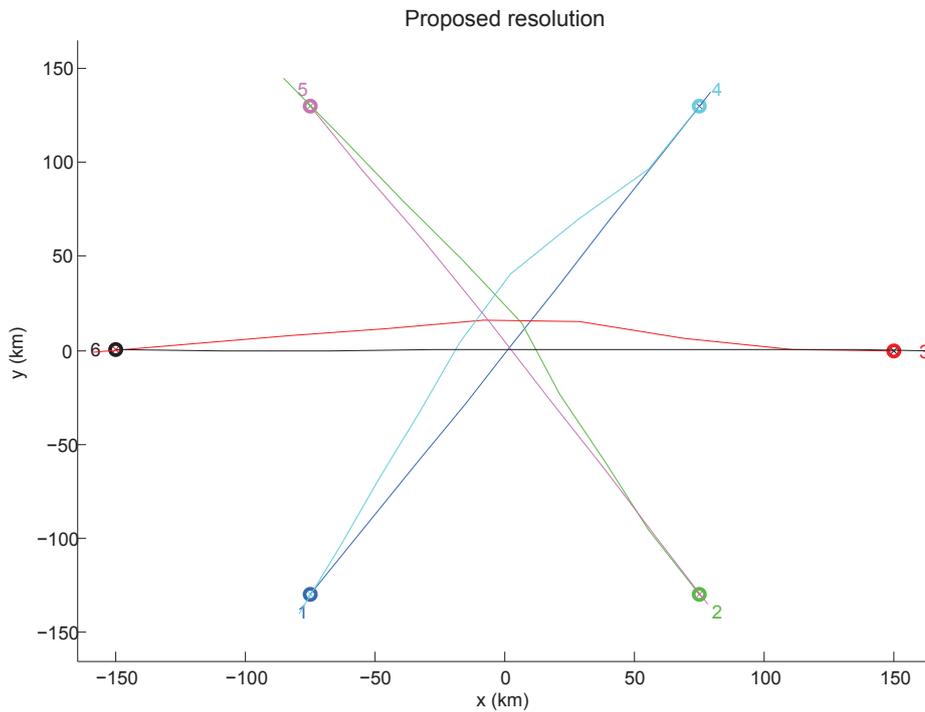


Figure 20: Resolution using scenario 1

because when aircraft are flying close to each other and a manouevre is required to avoid a conflict, the wind they experience is very similar, and thus ignoring the correlation structure

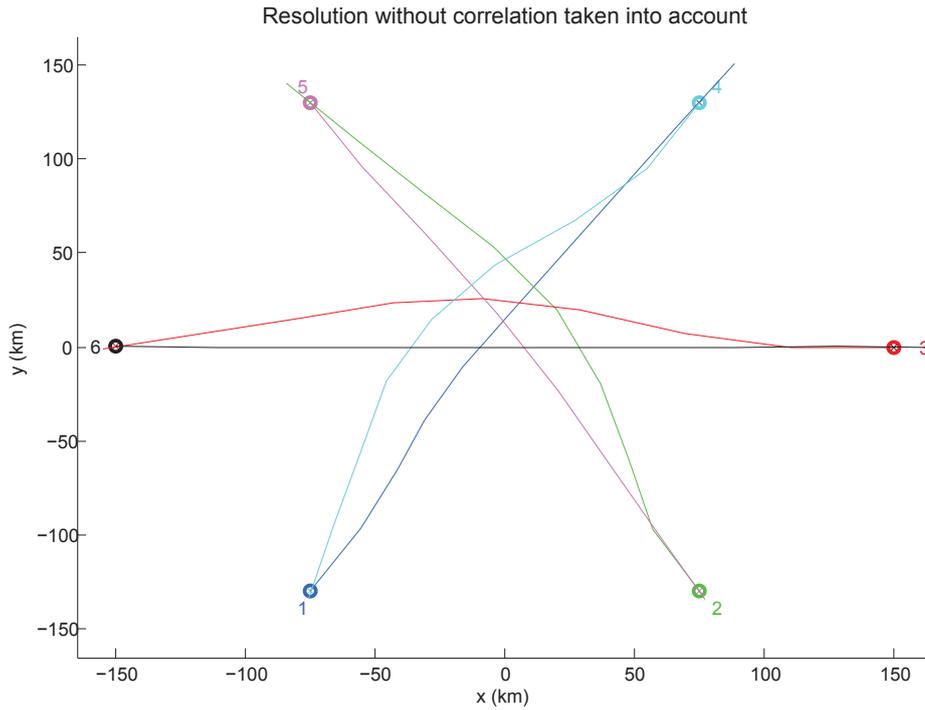


Figure 21: Resolution using scenario 2

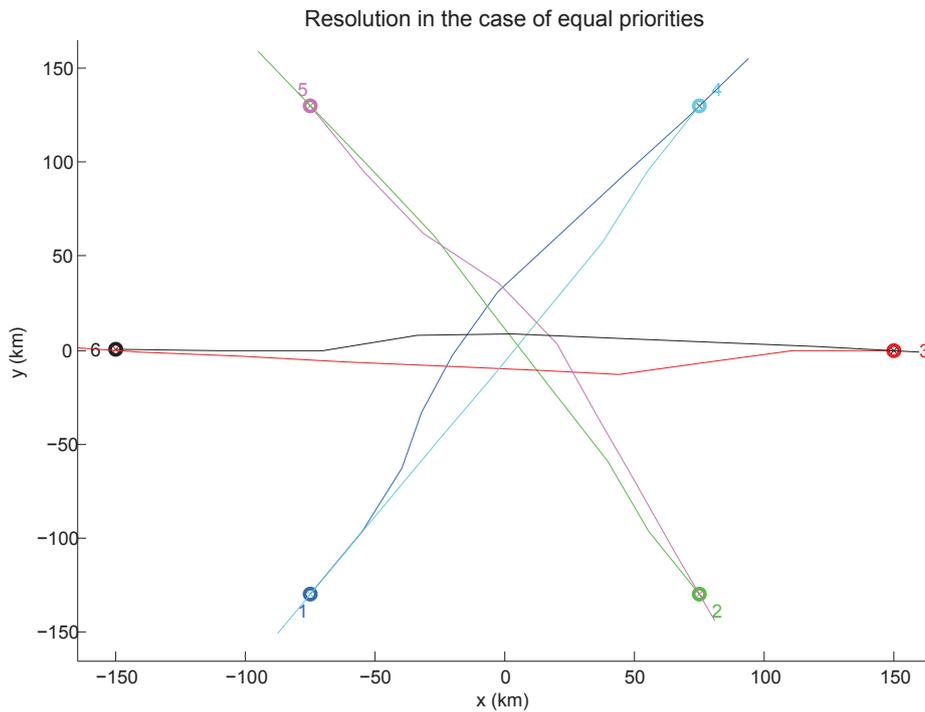


Figure 22: Resolution using scenario 3

produces more conservative results. Also, comparing figures 20, and 22, it can be seen that assigning all aircraft the same priority produces trajectories that are fairer to all aircraft in the situation, in the sense that all aircraft contribute similarly to the conflict resolution, through

similar extra flying distances.

### 5.2.2 Effect of priorities

In order to assess the effect of different priorities, we compared the two scenarios 1 and 3 above in terms of the actual extra distance flown by the aircraft for the cases that the model mismatch and the wind uncertainty did not make the optimization problem infeasible. The results of the Monte Carlo runs are shown as a box-and-whisker diagram in Figure 23.

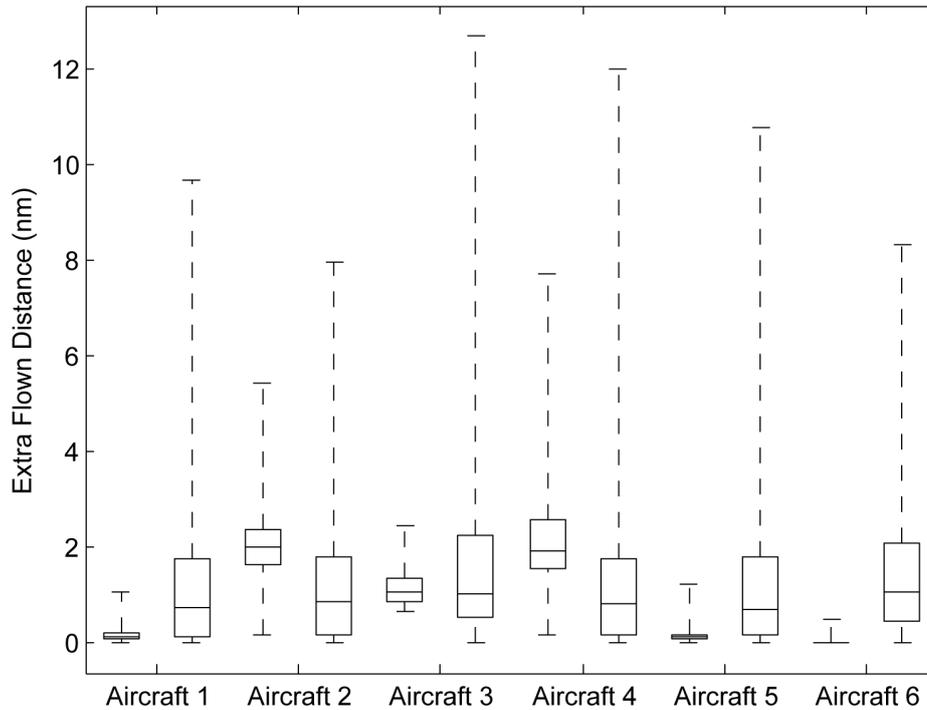


Figure 23: Extra flown distance per aircraft. The statistics of the prioritized solution and those of the equal priorities case are depicted on the left and right, respectively for each aircraft. The central mark inside each box is the median and the box contains 50% of all scenarios. The uppermost and lowermost whiskers represent the most extreme scenarios.

Even though it may seem that some aircraft perform better on average when all aircraft have the same priority, the standard deviation of the distance among the wind scenarios is bigger than when priorities are used in the formulation. This is explained by the implicit information that priorities carry, making the solution of the optimization problem consistent between the different times that the optimization problem is solved. This is also demonstrated in the first column of Table 2, as the total extra flown distance for each scenario is better on average when priorities are introduced. Furthermore, as in Table 2, despite the fact that the algorithms in several cases demonstrated infeasibility, there were no cases that a conflict actually occurred without the algorithm detecting it first. Also, using the priorities results in less infeasible situations, as the algorithm was able to resolve the conflicts for a larger percentage of the wind scenarios. However, a clear disadvantage when introducing priorities is that the computational times are higher, in a worst case setting, not allowing to use the method in real time always. Nevertheless, one should not forget that the situation itself is quite complex and rather unreasonable for an actual air traffic sample.

Case	Mean total extra flown distance (nm)	Actual conflicts	Infeasible cases	Minimum observed separation (nm)	Mean cpu time (mins)	Maximum cpu time (mins)
Full model	5.57	0%	17.3%	5.94	0.8	88.1
No Priorities	7.87	0%	40.5%	5.28	0.3	12.4

Table 2: Comparison of prioritized and unprioritized resolutions

Case	Mean total extra flown distance (nm)	Actual conflicts	Infeasible cases	Minimum observed separation (nm)	Mean cpu time (mins)	Maximum cpu time (mins)
Full model	5.57	0%	17.3%	5.94	0.8	88.1
Uncorrelated	14.19	0%	3%	9.19	10.6	709.8

Table 3: Comparison of using and ignoring the correlation structure of the wind

### 5.2.3 Effect of wind correlation

We then compared the scenarios 1 and 2 above, in order to assess how much ignoring the correlation structure described in this study can affect the performance of the resolution algorithm. As Figure 24 suggests, the aircraft may have to deviate about twice as much as they would have to in the case that the correlation structure is implemented in the optimization. Table 3, summarizes the results for various wind realizations. This highlights the advantage of implementing the wind correlation structure in the optimization, since otherwise the aircraft are forced to perform much more conservative maneuvers. The only advantage of the uncorrelated wind case is the better behaviour against uncertainty; the algorithm in this case is robust in more situations than the correlated case. This is expected, as aircraft fly longer and farther away from each other. On the other hand, the computation times needed are prohibitive, since on average the algorithm performs worse than the real time. Another interesting fact is how the symmetry of the situation affects the extra distance flown in each case. The order of priority is assumed to increase with aircraft number. Intuitively one would expect aircraft 1, which has the lowest priority to make the largest maneuver, but this is not the case. This happens because aircraft 5 and 6 have higher priority and hence force aircraft 2 and 3, respectively, to maneuver in order to avoid collisions. Also, aircraft 4 has to avoid first aircraft 5 and then 6, producing again a large maneuver. Aircraft 1 on the other hand has more freedom in this setup, as aircraft 2 maneuvers away from aircraft 6 (because of its high priority), leaving more maneuverability room for aircraft 1.

## 5.3 Conclusions

In an effort to better align the optimization based mid-term conflict resolution methods developed in WP5 with the A3 concept of operations, a model predictive control scheme to allow for

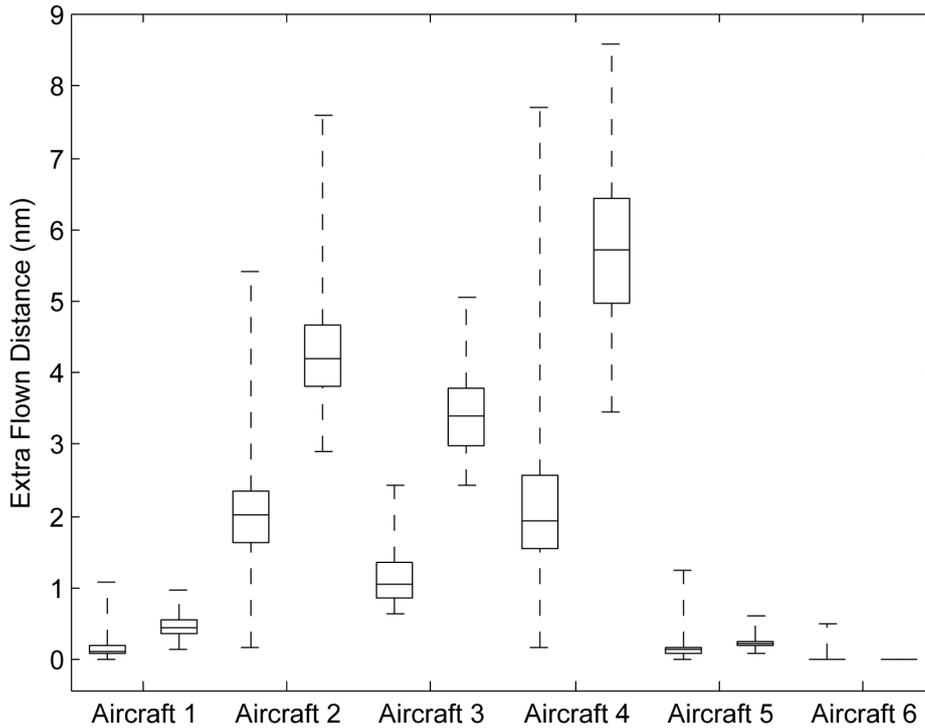


Figure 24: Extra flown distance per aircraft. The statistics of the correlated model and those of the uncorrelated model are depicted on the left and right, respectively for each aircraft. The central mark inside each box is the median and the box contains 50% of all scenarios. The uppermost and lowermost whiskers represent the most extreme scenarios.

directly coding priority rules into the resolution algorithms was developed. In this section the initial feasibility study was conducted, which clearly demonstrates that the proposed formulation is viable in principle. To fully align with the concept this formulation now needs to be merged with earlier developments in WP5 (documented in Sections 4 and 3), in particular with respect to the need for:

1. Decentralization of the resolution maneuver computation; and, 2. Clarification of the interaction with the short term conflict resolution algorithms.

These requirements will be addressed in D5.4 as part of the final validation of the conflict resolution algorithms.

## 6 Compatibility of medium and short-term methods proposed with A<sup>3</sup>ConOps requirements

### 6.1 Mid-term Methods

Following the discussions in D5.3i, we present here the corresponding tables for the ConOps requirements and the capabilities of the algorithms. Table 4 compares the capabilities of Decentralized Robust Model Predictive Control, presented in Section 4. As can be seen by the table, the algorithm already satisfies most of the ConOps requirements and further development to address the remaining issues will be carried out in the future work.

<b>Feature</b>	<b>ConOps Requirement</b>	<b>Robust decentralized MPC</b>
Look-ahead time	15–20 minutes	Requirement met
Coordination	Not required	Requirement met
Principle of use	Intent	Requirement met
Priority rules	Yes	Requirement not met
Secondary conflict creation	Do not	None created
2-minute state vector conflict	Avoid	Not addressed yet No problem in principle
Type of resolution algorithm	Intent-based	Requirement met
Alternative resolutions	Should provide	Not provided yet

Table 4: Comparison of ConOps requirements and properties of the robust decentralised MPC algorithm for mid-term conflict resolution.

Similarly, Table 5 compares the combined approach of MPC&NF against the ConOps requirements for a Mid Term CR algorithm. Most ConOps requirements are already met by this method, but further development and testing of the method in simulations is needed to validate its efficiency and ensure the satisfaction of all requirements. Table 6 compares the ConOps

<b>Feature</b>	<b>ConOps Requirement</b>	<b>MPC &amp; NF</b>
Look-ahead time	15–20 minutes	Requirement met
Coordination	Not required	Requirement met
Principle of use	Intent	Requirement met
Priority rules	Yes	Requirement not met
Secondary conflict creation	Do not	Requirement met
2-minute state vector conflict	Avoid	Not addressed yet
Type of resolution algorithm	Intent-based	Requirement met
Alternative resolutions	Should provide	Requirement met

Table 5: Comparison of ConOps requirements and properties of the combined MPC&NF algorithm for mid-term conflict resolution.

requirements and properties of the Hierarchical MPC with Priorities algorithm for mid-term conflict resolution.

Feature	ConOps Requirement	Hierarchical MPC with Priorities
Look-ahead time	15-20 minutes	Requirement Met
Coordination	Not required	Requirement Met
Principle of use	Intent	Requirement met
Priority rules	Yes	Requirement Met
Secondary conflict creation	Do not	Requirement Met
2-minute state vector conflict	Avoid	Not addressed yet
Type of resolution algorithm	Intent-based	Requirement Met
Alternative resolutions	Should provide	Requirement Met Can provide

Table 6: Comparison of ConOps requirements and properties of the Hierarchical MPC with Priorities algorithm for mid-term conflict resolution.

### Current and future work on mid-term methods

The three approaches that have been presented in this section have been applied in a realistic ATC setting, integrating the previously developed methods with a realistic simulation environment.

For the MMPC, the capabilities of the proposed schemes in handling scenarios in which agents enter and leave have been discussed. An initial solution for resolving the dynamic scenario, which has been found to work in the majority of cases, albeit with higher cost trajectories associated with joining aircraft, has been presented. A heuristic for choosing the update order of aircraft, based on the magnitudes of previous disturbances, has been presented. Other alternatives can be imagined, for instance giving priority to aircraft which would reduce the constrainedness of the problem by updating their policy. Directions for follow-up research include further investigation into the optimal update sequencing, and in establishing methods for enabling the resolution scheme to cope with interactions between neighbouring regions, which are currently treated as separate scenarios.

Regarding the alternative of MPC and NFs, the decentralized scheme proposed offers the same feasibility properties as the initial centralized problem. Current work on this method concentrates on the directions of testing the algorithm in a high density traffic sample, as well as further developing the algorithm to accommodate human factors considerations. Moreover, extensions to 3D cases, as well as the use of more advanced versions of Navigation Functions need also to be investigated.

## 6.2 Short-term Methods

A first discussion on the compatibility of the NFs framework with the ConOps requirements has been presented in D5.3i. It has been shown there that this choice of algorithms already satisfies most of the requirements set by WP1. As in D5.3i, we present the ConOps requirements along with the NFs characteristics in Table 7. We have used **bold** to indicate the areas that have been updated from D5.3i, as a result of the latest algorithm development.

As is denoted in the table and has been shown in the NFs algorithm description, significant progress has been made with respect to constraint handling: constant speed has been made possible and bounds have been introduced for the climb and descent angles. Vertical speed can now be independently regulated, resulting in controls that are compatible with current ATM practice. Lookahead time is implicitly bounded by the limited sensing range that has been integrated in the algorithm.

		ConOps Requirements	Proposed Algorithm	Comments
Inputs	Ownship	State, Intent	State, Intent	
	Traffic	State, Intent (opt.)	State	
Outputs	Resolution Manoeuvre	Requirement met; specifically: <b>Speed</b> <b>Climb-descent rate</b> <b>Rate of heading</b> <b>turn</b>	Manoeuvre defined implicitly <b>Constant Speed, bounded</b> <b>climb-descent angle</b>	
Lookahead Time	Up to 3 to 5 min	Requirement met, <b>only local sensing</b> <b>for Conflict</b> <b>Detection</b>		
Priority Rules	No	Requirement met, with option of priority rules		
Assumptions	Implicit Coordination '1 to N' resolution No new conflicts	No direct coordination All possible conflicts avoided		

Table 7: Comparison of ConOps requirements for short-term CD&R and Decentralized Navigation Functions

## 7 Concluding remarks

In this deliverable, all enhancements of the CR algorithms that were developed in WP5 have been documented. Furthermore, the algorithms have been compared against the requirements of the A<sup>3</sup>ConOps defined in WP1.

For Short-term CD&R, Decentralized NF have been further extended to be able to operate in a 3D space, as well as to accommodate aircraft performance constraints. Further work is still needed to restrict the sensing range for the aircraft, in order to meet the corresponding A<sup>3</sup>ConOps requirement.

In Mid-term CD&R, three separate approaches have been presented. The first, Robust Decentralized MMPC has been further developed to reduce conservatism and has been tested in simulation using a realistic ATC simulator. The second approach described combines the best features of the decentralized NFs and MPC, and the results on testing in simulations with a more realistic FMS have been presented. The third and final approach is a novel formulation which explicitly incorporates priorities, as specified within the A<sup>3</sup>ConOps requirements. This is enabled via the use of integer programming Mixed Integer Linear Programs (MILPs). Work is underway with a view to decentralizing the scheme.

In Long-Term CD&R, following the conclusions of previous WP5 deliverables, no CD&R methods have been discussed in detail; an extension of Mid-Term CR methods is envisioned to be suitable for this purpose.

This deliverable serves as an input for WP8 and WP9, in the process of the refinement of the A<sup>3</sup> ConOps and determining the airborne requirements. It will also form the basis of the validation study to be documented in D5.4.

Details of all the developed models are given in appendices A to C.

## References

- [1] Distributed Air Traffic Flow Management. *iFly Deliverable D8.2*.
- [2] Identification of human factors for improvement of the a3 conops. *iFly Deliverable D2.3*, 2009.
- [3] G Chaloulos and J. Lygeros. Effect of wind correlation on aircraft conflict probability. *Journal of Guidance, Control and Dynamics*, 30, 2007.
- [4] Francesco Bellomi, Roberto Bonato, Vincenzo Nanni, and Alessandra Tedeschi. Satisficing game theory for conflict resolution and traffic optimization. *Air Traffic Control Quarterly*, 16:211–233, 2008.
- [5] A Bemporad and M Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35:407–427, 1999.
- [6] Henk Blom and John Lygeros. HYBRIDGE Final Project Report. Technical report, HYBRIDGE project, April 2005.
- [7] Eurocontrol Experimental Centre. User manual for the base of aircraft data, 2004.
- [8] G. Chaloulos, J. Lygeros, G. Roussos, K. Kyriakopoulos, E. Siva, A. Lecchini-Visintini, and P. Casek. Comparative study of conflict resolution methods. Technical Report Deliverable 5.1, iFly Project, 2007.

- [9] G. Chaloulos, G. Roussos, J. Lygeros, and K. Kyriakopoulos. Ground Assisted Conflict Resolution in Self-Separation Airspace. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, Hawaii, August 2008.
- [10] G. Chaloulos, P. Hokayem, and J. Lygeros. Distributed Hierarchical MPC for Conflict Resolution in Air Traffic Control. In *American Control Conference*, Baltimore, MD, USA, June 2010.
- [11] F. Clarke. *Optimization and Nonsmooth Analysis*. Addison-Wesley, 1983.
- [12] R.E. Cole, C. Richard, S. Kim, and D. Bailey. An assessment of the 60 km rapid update cycle (RUC) with near real-time aircraft reports. Technical Report NASA/A-1, MIT Lincoln Laboratory, July 15, 1998.
- [13] G. Cuevas, I. Echevoyen, J. García, P. Cásek, C. Keinrath, R. Wever, P. Gotthard, F. Bussink, and A. Luuk. Autonomous aircraft advanced (A<sup>3</sup>) Conops. *iFly Deliverable D1.3*, 2010.
- [14] Dimos V. Dimarogonas and Kostas J. Kyriakopoulos. A feedback control scheme for multiple independent dynamic non-point agents. *International Journal of Control*, 79(12):1613–1623, 2006.
- [15] Dimos V. Dimarogonas and Kostas J. Kyriakopoulos. Decentralized navigation functions for multiple robotic agents with limited sensing capabilities. *Journal of Intelligent and Robotic Systems*, 48(3):411–433, 2007.
- [16] Dimos V. Dimarogonas, Savvas G. Loizou, Kostas J. Kyriakopoulos, and Michael M. Zavlanos. A feedback stabilization and collision avoidance scheme for multiple independent non-point agents. *Automatica*, 42(2):229–243, 2006.
- [17] M. Egerstedt and Xiaoming Hu. Formation constrained multi-agent control. *Robotics and Automation, IEEE Transactions on*, 17(6):947–951, Dec 2001. ISSN 1042-296X. doi: 10.1109/70.976029.
- [18] A. Filippov. *Differential equations with discontinuous right-hand sides*. Kluwer Academic Publishers, 1998.
- [19] J. Hu, M. Prandini, and S. Sastry. Optimal Coordinated Maneuvers for Three-Dimensional Aircraft Conflict Resolution. *AIAA Journal of Guidance, Control and Dynamics*, 25(5), 2002.
- [20] ILOG SA. Technical report, Gentilly, France, 2008.
- [21] 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 A<sup>3</sup> operational concept. Technical Report Deliverable 5.2 (version 1.0), iFly Project, 2008.
- [22] 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. volume 3, pages 1694–1698, Chicago, IL , USA, June 2000.
- [23] A. Lecchini-Visintini, J. Lygeros, and J. Maciejowski. Simulated annealing: Rigorous finite-time guarantees for optimization on continuous domains. *Advances in Neural Information Processing Systems*, 20, 2007.

- [24] K.V. Ling, J.M. Maciejowski, and A.G. Richards. Multiplexed model predictive control. In *Proc. European Control Conference*, Kos, Greece, July 2007.
- [25] 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.
- [26] Ioannis Lymeropoulos, Andrea Lecchini, William Glover, Jan Maciejowski, and John Lygeros. A stochastic hybrid model for air traffic management processes. Technical Report CUED/F-INFENG/TR.572, Department of Engineering, Cambridge University, February 2007.
- [27] Ioannis Lymeropoulos, Andrea Lecchini, William Glover, Jan Maciejowski, and John Lygeros. A stochastic hybrid model for air traffic management processes. Technical Report CUED/F-INFENG/TR.572, Department of Engineering, Cambridge University, February 2007.
- [28] J. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2001.
- [29] Marco Porretta, Wolfgang Schuster, Arnab Majumdar, and Washington Ochieng. *THE JOURNAL OF NAVIGATION*, 63:61–88, 2010.
- [30] 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.
- [31] A. Richards and J.P. How. Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility. In *Proceedings of the American Control Conference 2003*, 2003.
- [32] Arthur Richards and Jonathan How. Robust variable horizon for vehicle maneuvering. *International Journal of Robust Nonlinear Control*, 16:333–351, 2006.
- [33] Elor Rimon and Daniel E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–508, 1992.
- [34] G. Roussos, G. Chaloulos, K. Kyriakopoulos, and J. Lygeros. Control of Multiple Non-Holonomic Air Vehicles under Wind Uncertainty Using Model Predictive Control and Decentralized Navigation Functions. In *IEEE Conference on Decision and Control*, Cancun, Mexico, December 2008.
- [35] Giannis Roussos and Kostas J. Kyriakopoulos. Decentralised navigation and collision avoidance for aircraft in 3d space. *2010 American Control Conference, Baltimore, USA*, 2010. to appear.
- [36] Giannis Roussos and Kostas J. Kyriakopoulos. Completely decentralised navigation of multiple unicycle agents with prioritization and fault tolerance. In *49<sup>th</sup> IEEE Conference on Decision and Control*, 2010. submitted.
- [37] Giannis P. Roussos and Kostas J. Kyriakopoulos. Towards constant velocity navigation and collision avoidance for autonomous nonholonomic aircraft-like vehicles. *48<sup>th</sup> IEEE Conference on Decision and Control, Shanghai, China*, pages 5661–5666, 2009.

- [38] Giannis P. Roussos, Dimos V. Dimarogonas, and Kostas J. Kyriakopoulos. 3D navigation and collision avoidance for a non-holonomic vehicle. *2008 American Control Conference, Seattle, Washington, USA*, 2008.
- [39] Giannis P. Roussos, Dimos V. Dimarogonas, and Kostas J. Kyriakopoulos. Distributed 3D navigation and collision avoidance for multiple nonholonomic agents. *European Control Conference*, pages 1830–1835, 2009.
- [40] T. Schouwenaars, B. De Moor, E. Feron, and J. How. Mixed integer programming for multi-vehicle path planning. In *Proceedings of the European Control Conference*, 2001.
- [41] Daniel Shevitz and Brad Paden. Lyapunov stability theory of nonsmooth systems. *IEEE Transactions on Automatic Control*, 39(9):1910–1914, 1994.
- [42] E. Siva, J. Maciejowski, G. Chaloulos, J. Lygeros, G. Roussos, and K. Kyriakopoulos. Intermediate report on advanced conflict resolution mechanisms for A<sup>3</sup> ConOps. Technical Report Deliverable 5.3i (version 1.0), iFly Project, 2009.
- [43] E. Siva, J. Maciejowski, and K.-V. Ling. Robust multiplexed model predictive control for agent based conflict resolution. In *to appear Proceedings of the 49th IEEE Conference on Decision and Control*, 2010.
- [44] David Wing, Robert A. Vivona, and David A. Roscoe. Airborne tactical intent-based conflict resolution capability. In *Proceedings 5th AIAA Aviation Technology, Integration, and Operations Conference*, number AIAA-2009-7020, September 2009.

# Appendices

## A Navigation Functions (NFs) with limited sensing range

We describe here in detail the Navigation Functions (NFs) algorithm proposed for short-term CD&R.

### A.1 Problem Formulation

In order to allow independent regulation of the vertical velocity of each aircraft  $i$ , we use the following kinematic model:

$$\begin{aligned} \dot{\mathbf{n}}_i &= \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \mathbf{J}_i \cdot \mathbf{u}_i, \\ \dot{z}_i &= w_i, \\ \dot{\phi}_i &= \omega_i, \end{aligned} \tag{1}$$

where  $\mathbf{q}_i = [x_i \ y_i \ z_i]^\top$  is the position vector with respect to an earth-fixed frame  $\mathcal{E}$  (see Figure 25), while  $\mathbf{J}_i = [\cos(\phi_i) \ \sin(\phi_i)]^\top$ ,  $\mathbf{n}_i = [x_i \ y_i]^\top$  is the projection of the aircraft's position on the horizontal  $x - y$  plane,  $z_i$  its altitude and  $\phi_i$  the heading angle, i.e. the angle between the heading direction of the aircraft and the global  $x$  axis. The control vector comprises the horizontal velocity  $\mathbf{u}_i$ , the vertical velocity  $w_i$  and the angular heading velocity  $\omega_i$ .

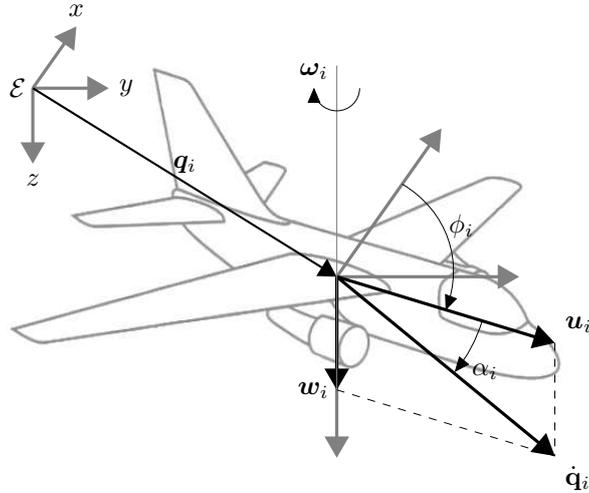


Figure 25: Model Coordinates  $\mathbf{q} = [x_i \ y_i \ z_i]^\top$ ,  $\phi_i$  and controls  $u_i$ ,  $w_i$ ,  $\omega_i$ . Descent angle  $\alpha_i$  and vertical velocity  $w_i$  are shown here negative during descent.

Essentially, the above model is a unicycle on the  $x - y$  plane, augmented with the vertical velocity  $w_i$  that controls the altitude  $z_i$ . We use the climb or descent angle  $\alpha_i$  as the angle between the resultant velocity vector  $\dot{\mathbf{q}}_i = [\dot{x}_i \ \dot{y}_i \ \dot{z}_i]^\top$  and the horizontal  $x - y$  plane,  $\alpha_i = \tan^{-1} \left( \frac{w_i}{|u_i|} \right)$ . Positive values of  $\alpha_i$  represent climbing and negative values descending.

Compared to the model used in earlier NFs-based 3D approaches [38, 37], (1) decouples horizontal and vertical maneuvering, allowing independent regulation of the vertical velocity. The pitch and roll angles are not included in the model here, as we assume that the low level

control systems, i.e. avionics onboard the aircraft, will control these angles to achieve the desired linear and angular velocities ( $u_i$ ,  $w_i$  and  $\omega_i$  respectively). The resulting control inputs of the algorithm presented here can be easily used as reference inputs for the autopilot to fly.

We formulate the problem as the decentralised navigation of a group of aircraft described by (1), towards their destinations  $\mathbf{n}_{id} = [x_{id} \ y_{id}]$  at the desired altitude  $z_{id}$ , with heading angle  $\phi_{id}$ . Each aircraft has a desired absolute horizontal speed  $u_{d_i} > 0$ , that can be constant, or regulated independently of the NFs algorithm (e.g.  $u_{d_i}$  can be the optimal cruising speed for the current altitude or the output of the MPC optimization performed in the Mid-term level), a maximum climb angle  $\alpha_{iC} > 0$  and maximum descent angle  $\alpha_{iD} < 0$ . Our aim is to apply this desired speed  $u_{id}$  for as long as possible, and to ensure that all aircraft's climb and descent angles never exceed the above defined bounds, i.e.  $\alpha_{iD} \leq \alpha_i \leq \alpha_{iC}$  at all times. Since this algorithms is intended for onboard application, the finite range of air-to-air communication must be taken into account, modeled as a limited sensing range between neighboring aircraft.

### A.1.1 Decentralised Navigation Functions with Local Sensing

A decentralised Navigation Function (NF) is of the form

$$\Phi_i = \frac{\gamma_i + f_i}{((\gamma_i + f_i)^k + G_i \cdot \beta_i)^{1/k}}, \quad (2)$$

which is constructed as explained in detail in [16]. Function  $G_i$  represents a measure of proximity to all possible conflicts involving aircraft  $i$ :  $G_i$  is zero when the  $i^{\text{th}}$  aircraft participates in a conflict, i.e. when aircraft  $i$  is at a distance equal to the separation minimum away from another one, and positive away from any conflicts. Function  $\gamma_{di}$  is a measure of proximity to the destination of aircraft  $i$ , while the term  $f_i = f_i(G_i)$  is required in distributed approaches to ensure some level of implicit coordination between aircraft in close proximity. Finally,  $\beta_i$  is responsible for render the boundary of the spherical workspace repulsive, so that all aircraft remain inside it.

In order to integrate limited sensing range in the potential construction and improve the numerical behaviour and practicality of the method, we employ dimensionless forms for the functions  $\gamma_i$ ,  $f_i$ ,  $\beta_i$  and  $G_i$ , as presented in detail in [36]. This facilitates the adaptation of the algorithm to the scale of each problem, including aircraft CD&R, without the need for additional parameter tuning. The construction of the potential field using the dimensionless forms is outlined below.

For the dimensionless obstacle function  $g_{ij}$  the dimensional obstacle function  $\hat{g}_{ij}$  is employed as defined in previous NF approaches:

$$\hat{g}_{ij} = \hat{g}_{ji} = \|\mathbf{q}_i - \mathbf{q}_j\|^2 - r_{ij}^2 \quad (3)$$

where  $r_{ij} \triangleq r_i + r_j$ . By the above definition,  $\hat{g}_{ij}$  is zero when aircraft  $i$ ,  $j$  are in conflict, i.e. when  $\|\mathbf{q}_j - \mathbf{q}_i\| = r_{ij}$ , and increases as they move away from each other.

We assume that each aircraft can sense or communicate with other aircraft that are within a maximum sensing range  $R_s$  away, i.e. when  $\|\mathbf{q}_j - \mathbf{q}_i\| \leq R_s$ . We will use this sensing range to nondimensionalise the repulsive function  $\hat{g}_{ij}$  between aircraft  $i$ ,  $j$  into  $g_{ij}$ :

$$\text{where } g_{ij} = \begin{cases} \frac{L(\hat{g}_{ij})}{R_s^2 - r_{ij}^2}, & \|\mathbf{q}_i - \mathbf{q}_j\| \leq R_s \\ 1, & \|\mathbf{q}_i - \mathbf{q}_j\| > R_s \end{cases} \quad (4)$$

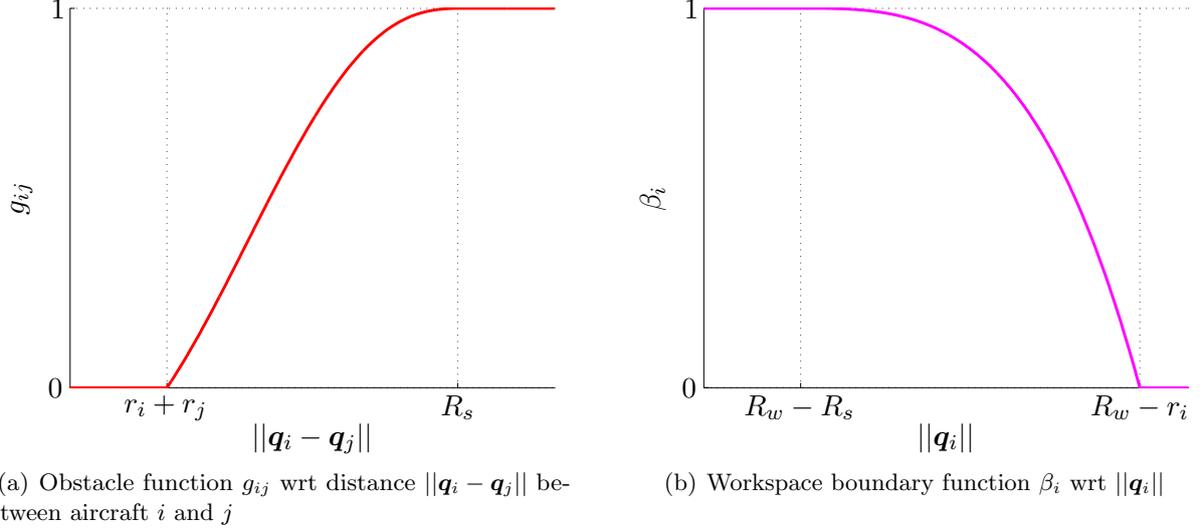


Figure 26: Dimensionless  $g_{ij}$  and  $\beta_i$  functions

where the shaping function  $L(x)$  is chosen to allow the formal properties of the potential field to be maintained:

$$L(x) = x^3 - 3x^2 + 3x \quad (5)$$

The dimensionless obstacle function  $g_{ij}$  defined above is zero when aircraft  $i, j$  are in conflict, i.e.  $\|\mathbf{q}_i - \mathbf{q}_j\| = r_{ij}$  and up to 1 at the boundary of the sensing zone, i.e. when  $\|\mathbf{q}_i - \mathbf{q}_j\| = R_s$ . Outside the sensing range of aircraft  $i$ ,  $g_{ij}$  is constant and equal to 1. The function  $g_{ij} = g_{ij}(\|\mathbf{q}_i - \mathbf{q}_j\|)$  is plotted in Figure 26(a). Since  $g_{ij}$  is constantly 1 when  $\|\mathbf{q}_i - \mathbf{q}_j\| \geq R_s$ , each aircraft  $i$  is only affected by other aircraft  $j \in N_i$  that are up to  $R_s$  away.

The complete obstacle function  $G_i$  is the product of the individual  $g_{ij}$ :

$$G_i = \prod g_{ij} \quad (6)$$

As shown previously, when aircraft  $j$  is outside the sensing zone of aircraft  $i$ , the corresponding obstacle function is  $g_{ij} = 1$  and does not affect  $G_i$ . Thus, calculating  $G_i$  requires only the knowledge about aircraft that are within the sensing range of  $i$ :

$$G_i = \prod_{j \in N_i} g_{ij} \quad (7)$$

where  $N_i = \{\|\mathbf{q}_i - \mathbf{q}_j\| < R_s\}$ .

Similarly to  $g_{ij}$ ,  $\beta_i$  is designed to limit the effect of the workspace boundary in a zone of width  $R_s$  near the boundary. The dimensional workspace boundary function  $\hat{\beta}_i$  is:

$$\hat{\beta}_i = (R_w - r_i)^2 - \|\mathbf{q}_i\|^2$$

The corresponding dimensionless function  $\beta_i$  is calculated similarly to  $g_{ij}$ :

$$\beta_i = \begin{cases} \frac{L(\hat{\beta}_i)}{(R_w - r_i)^2 - (R_w - R_s)^2}, & \|\mathbf{q}_i\| \geq R_w - R_s \\ 1, & \|\mathbf{q}_i\| < R_w - R_s \end{cases} \quad (8)$$

$$(9)$$

Thus,  $\beta_i$  becomes zero when aircraft  $i$  touches the workspace boundary, i.e.  $\|\mathbf{q}_i\| = R_w - r_i$ , and increases up to 1 when aircraft it is at a distance equal to or higher than  $R_s$  away from the workspace boundary, i.e.  $\|\mathbf{q}_i\| \leq R_w - R_s$ , see Figure 26(b).

For the target function  $\gamma_i$  we use the following form:

$$\gamma_i = \frac{\|\mathbf{q}_i - \mathbf{q}_{di}\|^2}{R_w^2} \quad (10)$$

The cooperation function  $f_i$  is used here as in [15]:

$$f_i(G_i) = \begin{cases} a_0 + \sum_{l=1}^3 a_l G_i^l, & G_i \leq X \\ 0, & G_i > X \end{cases} \quad (11)$$

where  $a_0 = Y$ ,  $a_1 = 0$ ,  $a_2 = \frac{-3Y}{X^2}$ ,  $a_3 = \frac{2Y}{X^3}$  and  $X, Y$  are positive parameters.  $X$  sets a threshold for  $G_i$ , such that values of  $G_i$  lower than  $X$  activate the cooperation function  $f_i$ . Parameter  $Y$  defines the maximum value of  $f_i$ , which is attained when  $G_i = 0$ .

The final result of using the above defined  $G_i$ ,  $\beta_i$  and  $\gamma_i$  in (25) for a setup with 3 obstacles is shown in Figures 27 and 28. The target  $\mathbf{q}_{di}$  is set in the center of the workspace and 3 obstacles are included. Figure 28 presents the potential field in the workspace, while Figure 27 shows the values of  $G_i$ ,  $\beta_i$ ,  $\gamma_i$  and  $\Phi_i$  along the positive  $x$  axis, that crosses through the center of one of the obstacles that is placed between the target and the workspace boundary. In this example we have assumed that the cooperation function  $f_i$  is not activated, i.e.  $f_i = 0$  everywhere. As Figure 27 demonstrates,  $G_i$  and  $\beta_i$  become less than 1 only within the sensing range  $R_s$  of the obstacle and workspace boundary, respectively. The dotted blue line represents the value of  $\Phi_i$  for  $G_i = \beta_i = 1$  everywhere, i.e. without the effect of any obstacles or the workspace boundary. As expected, this coincides with the actual  $\Phi_i$  outside the sensing range of the obstacle and the workspace boundary.

The potential described above has been proven to be a valid NF in [36]. Thus it can be employed with the NF-based control scheme presented below for guaranteed CD&R.

## A.2 Short-term CD&R using Navigation Functions (NFs)

### A.2.1 Preliminaries

The aim of the control scheme presented here is to produce trajectories that are compatible with the aircraft characteristics and constraints, as well as with current ATM practice. Thus, we develop a control logic that yields more sensible manoeuvres than [39], while still maintaining the formal guarantees for conflict avoidance and stabilization. The control scheme we suggest relies on a Dipolar NF,  $\Phi_i$  (25) to ensure the conflict resolution and convergence characteristics of the resulting trajectories. We employ the gradient  $\nabla_i \Phi_i = \frac{\partial \Phi_i}{\partial \mathbf{q}_i}$ , where the notation  $\nabla_i \Phi_j = \frac{\partial \Phi_j}{\partial \mathbf{q}_i}$  stands for the gradient of potential  $\Phi_j$  with respect to aircraft's  $i$  position  $\mathbf{q}_i$ . Since  $\nabla_i \Phi_i = [\Phi_{ix} \ \Phi_{iy} \ \Phi_{iz}]^\top$  is expressed in earth-fixed coordinates, we use its projection along the aircraft's  $i$  heading direction, i.e. the direction on the horizontal plane defined by the heading angle  $\phi_i$ :  $P_i = \mathbf{J}_i^\top \cdot [\Phi_{ix} \ \Phi_{iy}]^\top$ . The sign of  $P_i$ ,  $s_i = \text{sgn}(P_i)$ , determines the direction of motion on the horizontal plane, using the modified sign function  $\text{sgn}$ :

$$\text{sgn}(x) \triangleq \begin{cases} 1, & \text{if } x \geq 0 \\ -1, & \text{if } x < 0. \end{cases}$$

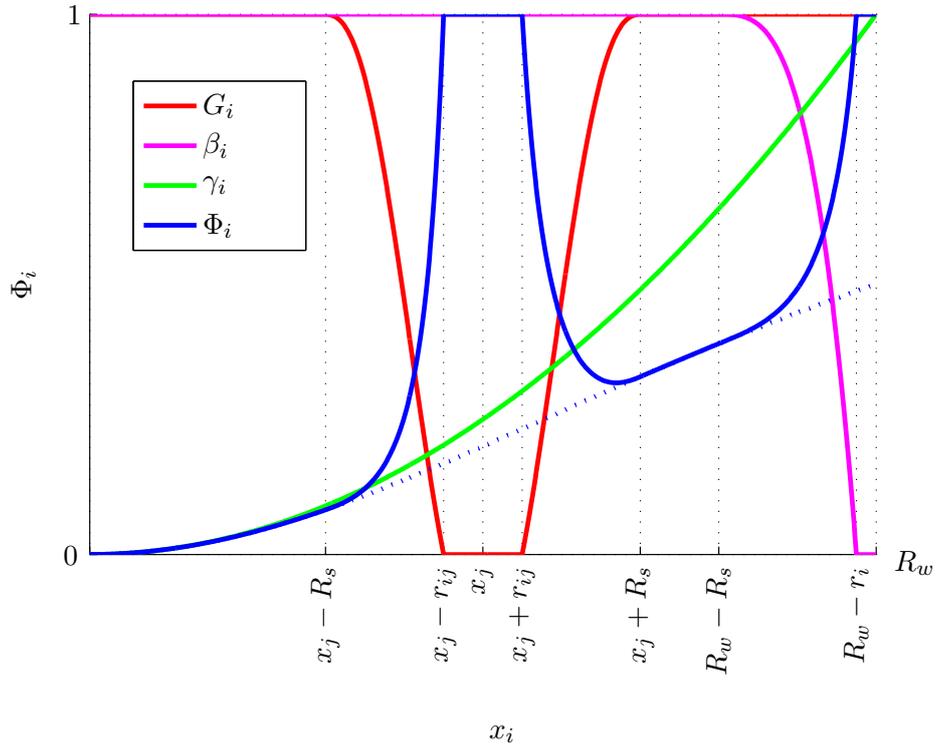


Figure 27: Obstacle function  $G_i$ , workspace boundary function  $\beta_i$ , target function  $\gamma_i$  and the resulting Navigation Function  $\Phi_i$  for  $x_i \in [0, R_w]$ ,  $y_i = 0$ .

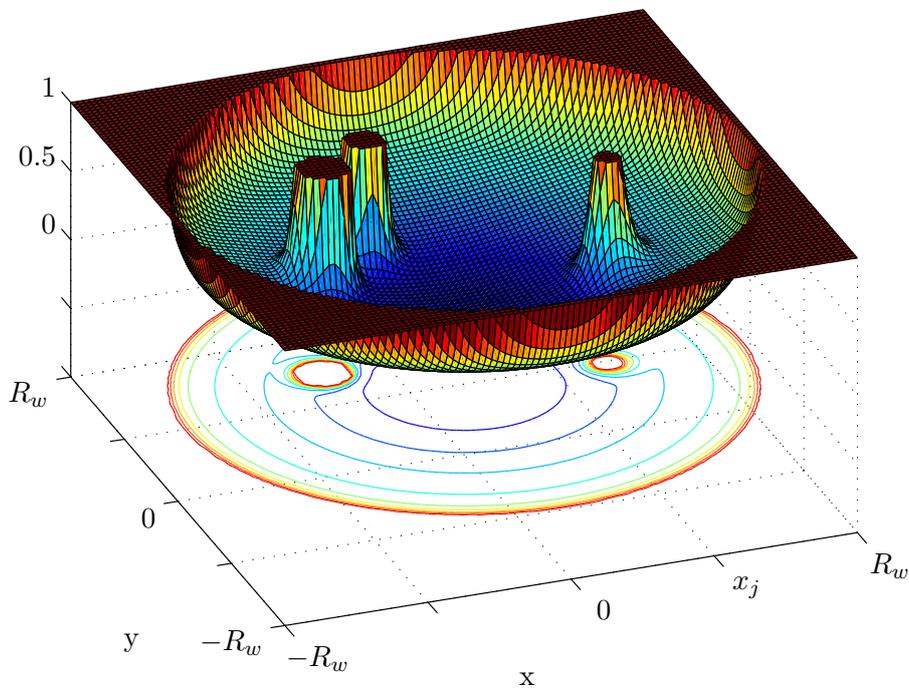


Figure 28: Navigation Function field in a workspace with 3 obstacles and local sensing.

The control law for the vertical velocity  $w_i$  depends on the elevation angle of the negated gradient, i.e. the angle between  $-\nabla_i\Phi_i$  and the horizontal plane, given as  $\alpha_{\mathbf{nh}i} = -\tan^{-1}\left(\frac{\Phi_{iz}}{\sqrt{\Phi_{ix}^2 + \Phi_{iy}^2}}\right)$ .

Since  $\alpha_{\mathbf{nh}i}$  can take any value in  $(-\frac{\pi}{2}, \frac{\pi}{2})$ , we use the reference elevation angle  $\tilde{\alpha}_i$ , which is confined within the aircraft's feasible climb and descent angles:

$$\tilde{\alpha}_i = \begin{cases} \alpha_{iD}, & \alpha_{\mathbf{nh}i} < \alpha_{iD} \\ \alpha_{\mathbf{nh}i}, & \alpha_{iD} \leq \alpha_{\mathbf{nh}i} \leq \alpha_{iC} \\ \alpha_{iC}, & \alpha_{\mathbf{nh}i} > \alpha_{iC}. \end{cases}$$

The corresponding reference slope is  $\tilde{t}_i = \tan \tilde{\alpha}_i$ .

For the heading control law we use the *nonholonomic* heading angle  $\phi_{\mathbf{nh}i}$  as a reference, which represents the heading of  $\text{sgn}(p_i)\nabla_i\Phi_i$ :

$$\phi_{\mathbf{nh}i} \triangleq \text{atan2}(\text{sgn}(p_i)\Phi_{iy}, \text{sgn}(p_i)\Phi_{ix}), \quad (12)$$

where the function  $\text{atan2}$  is

$$\text{atan2}(y, x) \triangleq \arg(x, y), \quad (x, y) \in \mathbb{C},$$

and  $p_i = \mathbf{J}_{id}^\top \cdot (\mathbf{n}_{i1} - \mathbf{n}_{i1d})$  is the position vector with respect to the destination, projected on the longitudinal axis of the desired orientation. Consequently,  $\text{sgn}(p_i)$  is equal to 1 in front of the target configuration and  $-1$  behind it. To ensure the continuity of  $\phi_{\mathbf{nh}i}$  on the destination, where  $\nabla_i\Phi_i = 0$ , we use the following approximation scheme [17]:

$$\hat{\phi}_{\mathbf{nh}i} \triangleq \begin{cases} \phi_{\mathbf{nh}i}, & \rho_i > \epsilon \\ \frac{\phi_{\mathbf{nh}i}(-2\rho_i^3 + 3\epsilon\rho_i^2) + \phi_{id}(-2(\epsilon - \rho_i)^3 + 3\epsilon(\epsilon - \rho_i)^2)}{\epsilon^3}, & \rho_i \leq \epsilon \end{cases}$$

where  $\rho_i = \sqrt{\Phi_{ix}^2 + \Phi_{iy}^2}$  and  $\epsilon$  a small positive constant. Thus, the angle  $\hat{\phi}_{\mathbf{nh}i}$  is continuous when  $\rho_i = 0$ :

$$\lim_{\mathbf{q}_i \rightarrow \mathbf{q}_{id}} \hat{\phi}_{\mathbf{nh}i} = \lim_{\rho_i \rightarrow 0} \hat{\phi}_{\mathbf{nh}i} = \hat{\phi}_{\mathbf{nh}i} \Big|_{\rho_i=0} = \phi_{id}$$

Consequently, whenever  $\mathbf{q}_i = \mathbf{q}_{id}$ , i.e. when aircraft  $i$  is at its target position, we have:

$$\hat{\phi}_{\mathbf{nh}i} = \phi_{id}. \quad (13)$$

For the design of the proposed control scheme we use the following three criteria explained below.

**Separation assurance and Stability** Ensuring a decreasing rate for the potential  $\Phi_i$  over time is crucial to guarantee convergence and conflict avoidance. The time derivative of  $\Phi_i$  can be written

$$\dot{\Phi}_i = \sum_{j=1}^N \nabla_j \Phi_i^\top \dot{\mathbf{q}}_j = P_i u_i + \Phi_{iz} w_i + \frac{\partial \Phi_i}{\partial t},$$

where the partial derivative  $\frac{\partial \Phi_i}{\partial t}$  sums the effect of all but the  $i^{\text{th}}$  aircraft's motion on  $\Phi_i$ :

$$\frac{\partial \Phi_i}{\partial t} = \sum_{j \neq i} \nabla_j \Phi_i^\top \cdot \begin{bmatrix} u_j \mathbf{J}_j \\ w_j \end{bmatrix},$$

This criterion is encoded into the continuous switch

$$\sigma_{\Phi_i} = \text{sat} \left( \frac{|u_i| (\tilde{t}_i \Phi_{iz} - |P_i| + \varepsilon) + \frac{\partial \Phi_i}{\partial t}}{|u_i| \tilde{t}_i \Phi_{iz}} \right) \quad (14)$$

where  $\text{sat}$  is a saturation function:

$$\text{sat}(x) = \begin{cases} 0, & x < 0 \\ x, & 0 \leq x \leq 1, \\ 1, & x > 1 \end{cases}$$

and  $\varepsilon$  is a small positive constant. Thus,  $\sigma_{\Phi_i}$  is:

- 1 when current horizontal velocity  $u_i$  ensures that  $\dot{\Phi} < -|u_i|\varepsilon$ ,
- 0 when  $u_i$  combined with vertical velocity  $\tilde{t}_i u_i$  maintains  $\dot{\Phi}_i > -|u_i|\varepsilon$ ,
- $0 < \sigma_{\Phi_i} < 1$  when  $u_i$  together with a nonzero vertical velocity  $w_i$ , with  $|w_i| < |\tilde{t}|u_i$ , yields  $\dot{\Phi}_i \leq -|u_i|\varepsilon$ .

**Horizontal distance from target** For each aircraft  $i$  we define the vertical Target Cylinder (TC) around its destination  $\mathbf{n}_{id}$ , as shown in Figure 29:

$$\mathcal{C}_i = \{\mathbf{n}_i \mid \|\mathbf{n}_i - \mathbf{n}_{id}\| \leq c_i\}.$$

Each TC  $\mathcal{C}_i$  is surrounded by a belt zone  $\mathcal{B}_i$  of thickness  $b_i$ ,

$$\mathcal{B}_i = \{\mathbf{n}_i \mid c_i < \|\mathbf{n}_i - \mathbf{n}_{id}\| \leq c_i + b_i\}.$$

The space outside  $\mathcal{C}_i$  and  $\mathcal{B}_i$  is the Maneuvering Space  $\mathcal{R}_i$  of each aircraft  $i$ ,

$$\mathcal{R}_i = \{\mathbf{n}_i \mid \|\mathbf{n}_i - \mathbf{n}_{id}\| > c_i + b_i\}.$$

Finally, let us define the Target Sphere  $\mathcal{S}_i$ , which is completely contained in  $\mathcal{C}_i$ ,

$$\mathcal{S}_i = \{\mathbf{q}_i \mid \|\mathbf{q}_i - \mathbf{q}_{id}\| \leq c_i\}.$$

The proposed control strategy uses different control schemes in  $\mathcal{C}_i$ ,  $\mathcal{B}_i$  and  $\mathcal{R}_i$ . Inside  $\mathcal{R}_i$ , the main objective of each aircraft  $i$  is to manoeuvre away from conflicts and towards the direction of the negated gradient  $-\nabla_i \Phi_i$ , while maintaining horizontal speed  $u_{id}$  and horizontal flight ( $w_i = 0$ ) for as long as possible (i.e. as long as conflict avoidance and stability are ensured). Following exactly the slope of the negated gradient is not required in  $\mathcal{R}_i$ . Inside  $\mathcal{C}_i$ , the horizontal speed  $u_i$  is gradually reduced, while the vertical velocity  $w_i$  is adjusted to match the gradient's slope, allowing the aircraft to converge to its target  $\mathbf{q}_{id}$ . The belt zone  $\mathcal{B}_i$  ensures that the transition between  $\mathcal{C}_i$  and  $\mathcal{R}_i$  does not cause discontinuity in the control inputs. The notion described here is captured by the continuous switch  $\sigma_{ni}$ , which expresses whether aircraft  $i$  is in  $\mathcal{C}_i$ ,  $\mathcal{R}_i$  or  $\mathcal{B}_i$ , see also Figure 29:

$$\sigma_{ni} = \text{sat} \left( \frac{\|\mathbf{n}_i - \mathbf{n}_{id}\| - c_i}{b_i} \right), \quad (15)$$

$$\text{so that } \sigma_{ni} = \begin{cases} 0, & \mathbf{n}_i \in \mathcal{C}_i \\ 1, & \mathbf{n}_i \in \mathcal{R}_i \\ a \in (0, 1], & \mathbf{n}_i \in \mathcal{B}_i. \end{cases}$$

It should be noted here that an aircraft may enter its TC and exit afterwards, if driven to do so by the potential's gradient. As it is shown in the stability analysis though, this does not affect the performance of the algorithm, since all aircraft eventually stay in their respective TCs. Since the algorithm presented here is intended for Short-term CD&R, aircraft are not supposed to reach close enough to their target to enter their TC while still flying autonomously. However, the TC is included here for completeness.

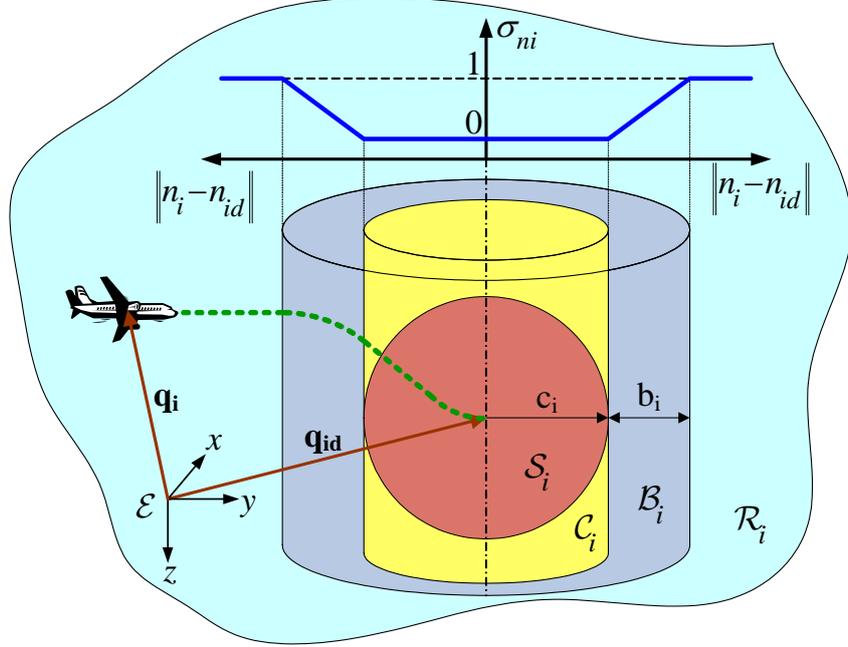


Figure 29: Target Cylinder  $\mathcal{C}_i$ , Target Sphere  $\mathcal{S}_i$ , Belt Zone  $\mathcal{B}_i$  and Maneuvering Space  $\mathcal{R}_i$  around the target  $\mathbf{q}_{id}$ .  $\sigma_{ni}$  varies linearly between 0 and 1 in  $\mathcal{B}_i$ .

**Elevation angle of the negated gradient** the aircraft are allowed to fly horizontally only when the absolute elevation angle  $|\alpha_{\mathbf{nh}i}|$  of the negated gradient  $-\nabla_i \Phi_i$  is lower than a high bound  $\theta_i^0$ . When this bound is exceeded, vertical maneuvering (via  $w_i$ ) is gradually activated, until  $|\alpha_{\mathbf{nh}i}|$  reaches a value of  $\hat{\theta}_i$ ,  $\hat{\theta}_i > \theta_i^0$ , where  $w_i$  is used to yield a total linear velocity  $\dot{\mathbf{q}}_i$  matching exactly the reference elevation angle  $\tilde{\alpha}_i$ . This is realised via the switch  $\sigma_{\alpha i}$ :

$$\sigma_{\alpha i} = \text{sat} \left( \frac{\hat{\theta}_i - |\alpha_{\mathbf{nh}i}|}{\hat{\theta}_i - \theta_i^0} \right).$$

As shown in Figure 30,  $\sigma_{\alpha i}$  is

- 0 when  $|\alpha_{\mathbf{nh}i}| \geq \hat{\theta}_i$ ,
- 1 when  $|\alpha_{\mathbf{nh}i}| \leq \theta_i^0$ , and
- $0 < \sigma_{\alpha i} < 1$  when  $\theta_i^0 < |\alpha_{\mathbf{nh}i}| < \hat{\theta}_i$ .

$\hat{\theta}$  and  $\theta_i^0$  must be selected so that  $\min(\alpha_{iC}, |\alpha_{iD}|) \geq \hat{\theta} > \theta_i^0 > 0$  holds.

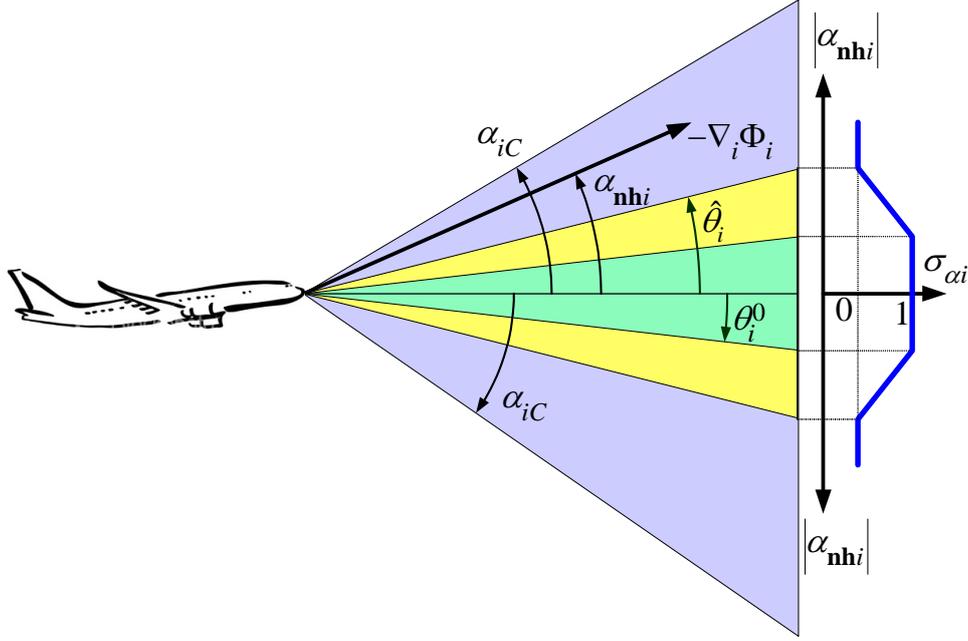


Figure 30: Angle parameters  $\hat{\theta}$ ,  $\theta_i^0$ , aircraft limits  $\alpha_{iC}$ ,  $\alpha_{iD}$  and switch  $\sigma_{\alpha_i}$  with respect to the gradient elevation angle  $\alpha_{nh_i}$ .

### A.2.2 Control Scheme

The control logic is built around the following principles:

- A nominal absolute speed  $U_i$  is used for  $u_i$  regulation.  $U_i$  is equal to the desired absolute horizontal speed  $u_{id}$  when  $q_i \notin \mathcal{S}_i$ , i.e. when aircraft  $i$  is more than  $c_i$  away from its destination, while it is continuously reduced to 0, as aircraft approaches its target inside  $\mathcal{S}_i$ .
- The magnitude of the actual horizontal velocity  $|u_i|$  is kept equal to the nominal signal  $U_i$  when  $\frac{\partial \Phi_i}{\partial t} \leq U_i (|P_i| - \tilde{t}_i \Phi_{iz} - \varepsilon)$ , i.e. the combination of horizontal and vertical velocities  $U_i$  and  $U_i \tilde{t}_i$ , respectively, can maintain  $\dot{\Phi}_i \leq U_i \varepsilon$ .
- Vertical velocity  $w_i$  is kept zero when all three of the criteria described above are met, i.e.:
  1. aircraft  $i$  is in its maneuvering zone,  $\mathbf{n}_i \in \mathcal{R}_i$ .
  2. The horizontal speed  $U_i$  ensures  $\dot{\Phi} = P_i U_i + \frac{\partial \Phi_i}{\partial t} \leq -U_i \varepsilon$ .
  3. The gradient's absolute elevation angle is at most  $\theta_i^0$ ,  $|\alpha_{nh_i}| \leq \theta_i^0$ .

Thus,  $w_i = 0$  when  $\sigma_{n_i} = \sigma_{\Phi_i} = \sigma_{\alpha_i} = 1$ .

- When separation and stability are at risk, vertical maneuvering via  $w_i$  is used up to an elevation slope  $\tilde{t}_i$ , by setting  $w_i = \tilde{t}_i u_i$ . If this alone is not enough to achieve  $\dot{\Phi}_i \leq U_i \varepsilon$ , the magnitudes of both linear velocities are increased proportionally to achieve  $\dot{\Phi}_i = |u_i| \varepsilon$ .
- The aircraft's slope is made equal to  $\tilde{t}_i$  when any of  $\sigma_{\Phi_i}$ ,  $\sigma_{n_i}$ ,  $\sigma_{\alpha_i}$  become zero, i.e. any of the following conditions hold:
  1. aircraft  $i$  is in its TC,  $\mathbf{n}_i \in \mathcal{C}_i$ .

2. The combination of horizontal speed  $U_i$  and vertical speed  $\tilde{t}_i U_i$  does not satisfy  $\dot{\Phi}_i \leq -U_i \varepsilon$ , i.e.  $U_i P_i + U_i \tilde{t}_i \Phi_{iz} + \frac{\partial \Phi_i}{\partial t} \geq -U_i \varepsilon$ .
  3. The gradient's absolute elevation angle is at least  $\hat{\theta}_i$ ,  $|\alpha_{\mathbf{nh}_i}| \geq \hat{\theta}_i$ .
- Continuous transition is desired for the horizontal velocity  $u_i$  and the vertical velocity  $w_i$ .
  - The heading velocity  $\omega_i$  should ensure that the heading error  $|\phi_i - \phi_{\mathbf{nh}_i}|$ , i.e. the absolute difference between the  $i$ -th aircraft's direction  $\phi_i$  and the heading of  $\text{sgn}(p_i) \nabla_i \Phi_i$ , is always decreasing, while also keeping steering effort low. Thus, whenever  $M_i = \dot{\phi}_{\mathbf{nh}_i} (\phi_i - \phi_{\mathbf{nh}_i}) \geq \varepsilon_\phi > 0$ , the error  $|\phi_i - \phi_{\mathbf{nh}_i}|$  is already decreasing, so the angular velocity is kept zero, i.e.  $\omega_i = 0$ . Otherwise, when  $M_i < \varepsilon_\phi$ , we use a stabilizing feedback law as in [39] to ensure that  $|\phi_i - \phi_{\mathbf{nh}_i}|$  is decreasing over time. The small constant  $\varepsilon_\phi$  is used here similarly to  $\varepsilon$  in the linear velocity control law, to ensure continuous transition for the heading velocity.

Based on these principles, we propose the following control scheme for the linear velocities  $u_i$  and  $w_i$  of each aircraft  $i$ :

$$u_i = \begin{cases} -s_i U_i, & \frac{\partial \Phi_i}{\partial t} \leq U_i (|P_i| - \tilde{t}_i \Phi_{iz} - \varepsilon) \\ -s_i \frac{U_i \varepsilon + \frac{\partial \Phi_i}{\partial t}}{|P_i| - \tilde{t}_i \Phi_{iz}}, & \frac{\partial \Phi_i}{\partial t} > U_i (|P_i| - \tilde{t}_i \Phi_{iz} - \varepsilon) \end{cases} \quad (16a)$$

$$w_i = (1 - \min(\sigma_{\Phi_i}, \sigma_{n_i}, \sigma_{\alpha_i})) \tilde{t}_i |u_i|. \quad (16b)$$

Note that the magnitude of the horizontal velocity  $|u_i|$  increases in the second case of (16a), and that the transition is continuous by construction,

$$\frac{\partial \Phi_i}{\partial t} > U_i (|P_i| - \tilde{t}_i \Phi_{iz} - \varepsilon) \Rightarrow \frac{U_i \varepsilon + \frac{\partial \Phi_i}{\partial t}}{|P_i| - \tilde{t}_i \Phi_{iz}} > U_i.$$

The nominal absolute horizontal velocity  $U_i$  is

$$U_i = \begin{cases} u_{id}, & \mathbf{n}_i \notin \mathcal{C}_i \\ \frac{\|\mathbf{q}_i - \mathbf{n}_{id}\|}{c_i} \cdot u_{id}, & \mathbf{n}_i \in \mathcal{C}_i. \end{cases} \quad (17)$$

The angular velocity  $\omega_i$  is given by:

$$\omega_i = \begin{cases} 0, & M_i \geq \varepsilon_\phi \\ \Omega_i \cdot \left(1 - \frac{M_i}{\varepsilon_\phi}\right), & 0 < M_i < \varepsilon_\phi \\ \Omega_i, & M_i \leq 0, \end{cases} \quad (18)$$

$$\begin{aligned} \text{where: } M_i &\triangleq \dot{\phi}_{\mathbf{nh}_i} (\phi_i - \phi_{\mathbf{nh}_i}), \\ \Omega_i &\triangleq -k_\phi (\phi_i - \phi_{\mathbf{nh}_i}) + \dot{\phi}_{\mathbf{nh}_i}. \end{aligned}$$

and  $k_\phi$  is a positive real gain.

### A.3 Conflict resolution Analysis

**Theorem 1.** *A team of aircraft described by (1) under the control law (16) for linear velocities remains always conflict free, i.e. no loss of separation occurs at any time.*

*Proof.* Since the aircraft are considered spherical, Loss of Separation (LoS) can occur only by translation. Thus, to ensure separation, it suffices to show that each aircraft  $i$  uses its linear velocities  $u_i, w_i$  to stay sufficiently far away from its neighbors. By definition, a Navigation Function is uniformly maximum on the boundary of obstacles, which represent the safety zones of conflicting aircraft. As a result, on the boundary of conflicts the negated gradient of a NF points away from them. It can be shown by (16) that for each aircraft  $i$  the inner product  $\nabla_i \Phi_i^\top \cdot \dot{\mathbf{q}}_i$  is non-positive. To do so, consider the definition of  $\alpha_{\text{nh}i}$ ,  $\tilde{\alpha}_i$  and  $\tilde{t}_i$ , to verify that  $\tilde{t}_i \Phi_{iz} \leq 0$ . From the control law (16a), we derive that  $P_i u_i \leq -P_i s_i U_i = -|P_i| U_i \leq 0$ . Additionally, (16b) yields  $\Phi_{iz} w_i = \tilde{t}_i \Phi_{iz} (1 - \min(\sigma_{\Phi_i}, \sigma_{n_i}, \sigma_{\alpha_i})) |u_i| \leq 0$ . Consequently, we deduce:

$$\nabla_i \Phi_i^\top \cdot \dot{\mathbf{q}}_i = P_i u_i + \Phi_{iz} w_i \leq 0. \quad (19)$$

Let us assume that a group of aircraft, which initially are sufficiently far apart from each other so that  $\Phi_i|_{t=0} < 1 \quad \forall i$ , cause a conflict. Since each  $\Phi_i$  is continuous and differentiable in space, this would mean that at least one conflicting aircraft  $i$  moved towards the direction of  $\nabla_i \Phi_i$ , causing the potential NF to attain its maximum value of 1. As shown in (19), this cannot be true and therefore no conflicts can occur between aircraft under the control law (16).  $\square$

**Theorem 2.** *Each aircraft  $i$  described by (1) under the control laws (16), (18) is asymptotically stabilised to its target  $\mathbf{q}_{id}$  with the desired heading angle  $\phi_{id}$ .*

*Proof.* As the control scheme is discontinuous, we employ Lyapunov analysis for nonsmooth systems to prove the stability of the system under the control laws (16),(18). The following candidate Lyapunov function is used:

$$V = \sum_{i=1}^N V_i, \quad V_i = \Phi_i + \frac{1}{2} (\phi_i - \phi_{\text{nh}i})^2. \quad (20)$$

The generalised derivative [11] of  $V = V(\mathbf{q})$ , where

$$\mathbf{q} = [\mathbf{q}_i^\top \dots \mathbf{q}_N^\top \phi_1 \dots \phi_N \phi_{\text{nh}1} \dots \phi_{\text{nh}N}]^\top,$$

$$\text{is } \partial V = \begin{bmatrix} \sum_i \nabla_1 \Phi_i \\ \vdots \\ \sum_i \nabla_N \Phi_i \\ 1/2 \nabla_{\phi_1} (\phi_1 - \phi_{\text{nh}1})^2 \\ \vdots \\ 1/2 \nabla_{\phi_N} (\phi_N - \phi_{\text{nh}N})^2 \\ 1/2 \nabla_{\phi_{\text{nh}1}} (\phi_1 - \phi_{\text{nh}1})^2 \\ \vdots \\ 1/2 \nabla_{\phi_{\text{nh}N}} (\phi_N - \phi_{\text{nh}N})^2 \end{bmatrix} = \begin{bmatrix} \sum_i \nabla_1 \Phi_i \\ \vdots \\ \sum_i \nabla_N \Phi_i \\ (\phi_1 - \phi_{\text{nh}1}) \\ \vdots \\ (\phi_N - \phi_{\text{nh}N}) \\ -(\phi_1 - \phi_{\text{nh}1}) \\ \vdots \\ -(\phi_N - \phi_{\text{nh}N}) \end{bmatrix}. \quad \text{Consider the complete multi-agent system}$$

$\dot{\mathbf{x}} = f(\mathbf{x})$  resulting from the composition of (1), and its Filippov set [18]  $K[f(\mathbf{x})]$ , where:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_N \\ \phi_1 \\ \vdots \\ \phi_N \\ \phi_{\text{nh}1} \\ \vdots \\ \phi_{\text{nh}N} \end{bmatrix}, \quad f(\mathbf{x}) = \begin{bmatrix} u_1 \mathbf{J}_1 \\ w_1 \\ \vdots \\ u_N \mathbf{J}_N \\ w_N \\ \omega_1 \\ \vdots \\ \omega_N \\ \dot{\phi}_{\text{nh}1} \\ \vdots \\ \dot{\phi}_{\text{nh}N} \end{bmatrix}, \quad K[f] = \begin{bmatrix} K[u_1] \mathbf{J}_1 \\ K[w_1] \\ \vdots \\ K[u_N] \mathbf{J}_N \\ K[w_N] \\ \omega_1 \\ \vdots \\ \omega_N \\ \dot{\phi}_{\text{nh}1} \\ \vdots \\ \dot{\phi}_{\text{nh}N} \end{bmatrix}.$$

Using the chain rule given in [41], we calculate the generalised time derivative of  $V$ ,

$$\begin{aligned}
\dot{\tilde{V}} &= \bigcap_{\xi \in \partial V} \xi^\top K[f] = \\
&= \sum_i^N \sum_j^N K[u_i] \nabla_i \Phi_j^\top \begin{bmatrix} \mathbf{J}_i \\ 0 \end{bmatrix} + \sum_i^N \sum_j^N K[w_i] \frac{\partial \Phi_j}{\partial z_i} + \\
&\quad + \sum_i (\phi_i - \phi_{\mathbf{n}h_i}) (\omega_i - \dot{\phi}_{\mathbf{n}h_i}) = \\
&= \sum_i K[u_i] P_i + \sum_i \sum_{j \neq i} K[u_j] \nabla_j \Phi_i^\top \begin{bmatrix} \mathbf{J}_j \\ 0 \end{bmatrix} + \\
&\quad + \sum_i K[w_i] \Phi_{iz} + \sum_i \sum_{j \neq i} K[w_j] \frac{\partial \Phi_i}{\partial z_j} - \\
&\quad - \sum_i \theta_i \dot{\theta}_i,
\end{aligned}$$

where  $\theta_i = (\phi_i - \phi_{\mathbf{n}h_i})$ . By (18) we deduce:

$$\begin{aligned}
\dot{\theta}_i &= (\omega_i - \dot{\phi}_{\mathbf{n}h_i}) \stackrel{(18)}{=} \\
&= \begin{cases} -\dot{\phi}_{\mathbf{n}h_i}, & M_i \geq \varepsilon_\phi \\ -\left[ k_\phi \left( 1 - \frac{M_i}{\varepsilon_\phi} \right) + \frac{\dot{\phi}_{\mathbf{n}h_i}^2}{\varepsilon_\phi} \right] \cdot \theta_i, & 0 < M_i < \varepsilon_\phi \\ -k_\phi \theta_i, & M_i \leq 0. \end{cases}
\end{aligned}$$

Because of the switching linear velocity control law, we discriminate between the following three sets of agents:

$$\begin{aligned}
Q_1 &\triangleq \left\{ i \in \{1, \dots, N\} \mid \left| \frac{\partial \Phi_i}{\partial t} - U_i |P_i| + U_i \varepsilon \leq 0 \right. \right\}, \\
Q_2 &\triangleq \left\{ i \in \{1, \dots, N\} \mid 0 < \frac{\partial \Phi_i}{\partial t} - U_i |P_i| + U_i \varepsilon \leq -\tilde{t}_i U_i \Phi_{iz} \right\}, \\
Q_3 &\triangleq \left\{ i \in \{1, \dots, N\} \mid \left| \frac{\partial \Phi_i}{\partial t} - U_i |P_i| + U_i \varepsilon > -\tilde{t}_i U_i \Phi_{iz} \right. \right\}.
\end{aligned}$$

Similarly, we define the following non-intersecting sets:

$$\begin{aligned}
T_1 &\triangleq \{i \in \{1, \dots, N\} \mid M_i \geq \varepsilon_\phi\}, \\
T_2 &\triangleq \{i \in \{1, \dots, N\} \mid 0 < M_i < \varepsilon_\phi\}, \\
T_3 &\triangleq \{i \in \{1, \dots, N\} \mid M_i \leq 0\}.
\end{aligned}$$

By the control law (16) we deduce:

$$K[u_i] = \begin{cases} -K[s_i] \cdot U_i, & i \in Q_1 \cup Q_2 \\ -K[s_i] \frac{U_i \varepsilon + \frac{\partial \Phi_i}{\partial t}}{|P_i| - \tilde{t}_i \Phi_{iz}}, & i \in Q_3 \end{cases} \quad (21)$$

$$K[w_i] = (1 - \min(\sigma_{\Phi_i}, \sigma_{n_i}, \sigma_{\alpha_i})) \tilde{t}_i |u_i|. \quad (22)$$

Using the above set definitions, we proceed with  $\tilde{V}$ :

$$\begin{aligned}
\dot{\tilde{V}} &\stackrel{(22)}{=} \sum_{Q_1 \cup Q_2} \left\{ -K[s_i]P_i U_i + \frac{\partial \Phi_i}{\partial t} \right\} + \\
&+ \sum_{Q_3} \left\{ -K[s_i]P_i \frac{U_i \varepsilon + \frac{\partial \Phi_i}{\partial t}}{|P_i| - \tilde{t}_i \Phi_{iz}} + \frac{\partial \Phi_i}{\partial t} \right\} + \\
&+ (1 - \min(\sigma_{\Phi_i}, \sigma_{ni}, \sigma_{\alpha_i})) \tilde{t}_i |u_i| \Phi_{iz} - \sum_{T_1} \dot{\phi}_{\mathbf{nh}i} \theta_i - \\
&- \sum_{T_2} \left[ k_\phi \left( 1 - \frac{M_i}{\varepsilon_\phi} \right) + \frac{\dot{\phi}_{\mathbf{nh}i}^2}{\varepsilon_\phi} \right] \theta_i^2 - \sum_{T_3} k_\phi \theta_i^2.
\end{aligned}$$

Using the control law (16a), equation (14) yields

$$\sigma_{\Phi_i} = \begin{cases} 1, & i \in Q_1 \\ \frac{U_i(\tilde{t}_i \Phi_{iz} - |P_i| + \varepsilon) + \frac{\partial \Phi_i}{\partial t}}{U_i \tilde{t}_i \Phi_{iz}} \in [0, 1), & i \in Q_2 \\ 0, & i \in Q_3. \end{cases}$$

Note that  $\tilde{t}_i \Phi_{iz} \leq 0$  and  $(1 - \min(\sigma_{\Phi_i}, \sigma_{ni}, \sigma_{\alpha i})) \geq (1 - \sigma_{\Phi_i}) \geq 0$ . Using the above results yields:

$$\begin{aligned}
\dot{\tilde{V}} &\leq \sum_{Q_1 \cup Q_2} \left\{ -|P_i| U_i + (1 - \sigma_{\Phi_i}) \tilde{t}_i U_i \Phi_{iz} + \frac{\partial \Phi_i}{\partial t} \right\} + \\
&+ \sum_{Q_3} \left\{ -|u_i| (|P_i| - \tilde{t}_i \Phi_{iz}) + \frac{\partial \Phi_i}{\partial t} \right\} - \\
&- \sum_{T_1} \theta_i \dot{\phi}_{\mathbf{nh}i} - \sum_{T_2} \left[ k_\phi \left( 1 - \frac{M_i}{\varepsilon_\phi} \right) + \frac{\phi_{\mathbf{nh}i}^2}{\varepsilon_\phi} \right] \theta_i^2 - \\
&- \sum_{T_3} k_\phi \theta_i^2 = \\
&= \sum_{Q_1} \left\{ -|P_i| U_i + \frac{\partial \Phi_i}{\partial t} \right\} - \\
&- \sum_{Q_2} \left\{ |P_i| U_i - \frac{U_i (|P_i| - \varepsilon) - \frac{\partial \Phi_i}{\partial t} \tilde{t}_i U_i \Phi_{iz}}{U_i \tilde{t}_i \Phi_{iz}} \tilde{t}_i U_i \Phi_{iz} - \frac{\partial \Phi_i}{\partial t} \right\} - \\
&- \sum_{Q_3} \left\{ \frac{U_i \varepsilon + \frac{\partial \Phi_i}{\partial t}}{|P_i| - \tilde{t}_i \Phi_{iz}} (|P_i| - \tilde{t}_i \Phi_{iz}) + \frac{\partial \Phi_i}{\partial t} \right\} - \\
&- \sum_{T_1} M_i - \sum_{T_2} \left[ k_\phi \left( 1 - \frac{M_i}{\varepsilon_\phi} \right) \theta_i^2 + \frac{M_i^2}{\varepsilon_\phi} \right] - \\
&- \sum_{T_3} k_k \theta_i^2 = \\
&= \sum_{Q_1} \left\{ -|P_i| U_i + \frac{\partial \Phi_i}{\partial t} \right\} - \sum_{Q_2} U_i \varepsilon - \sum_{Q_3} U_i \varepsilon - \\
&- \sum_{T_1} M_i - \sum_{T_2} \left[ k_\phi \left( 1 - \frac{M_i}{\varepsilon_\phi} \right) \theta_i^2 + \frac{M_i^2}{\varepsilon_\phi} \right] - \\
&- \sum_{T_3} k_k \theta_i^2.
\end{aligned}$$

Taking into account the conditions that hold within each set, we derive that  $\dot{\tilde{V}} \leq 0$ . Since each  $V_i$ , and consequently  $V$ , is regular [11] and the level sets of  $V$  are compact, the nonsmooth version of LaSalle's invariance principle [41] can be applied. We conclude that the trajectory of the closed-loop system converges to the largest invariant subset  $S$ :  $S \triangleq \left\{ [\mathbf{q}^\top, \phi]^\top \mid 0 \in \tilde{V} \right\}$ . From the definitions of  $T_1, T_2, T_3$  we deduce:

$$\begin{aligned}
\sum_{T_1} M_i &> 0, \\
\sum_{T_2} \left[ k_\phi \left( 1 - \frac{M_i}{\varepsilon_\phi} \right) \theta_i^2 + \frac{M_i^2}{\varepsilon_\phi} \right] &> 0.
\end{aligned}$$

Consequently, for  $\dot{\tilde{V}} = 0$  to hold, all  $i$  must be in  $T_3$ . Therefore the set  $S$  is:

$$\begin{aligned}
S = \{ \mathbf{n} : (|P_i| U_i - \frac{\partial \Phi_i}{\partial t} = 0 \forall i \in Q_1) \wedge (\varepsilon U_i = 0 \forall i \in Q_2) \wedge \\
\wedge (\theta_i = \phi_i - \phi_{\mathbf{nh}i} = 0 \forall i) \}.
\end{aligned}$$

For each  $i$  that is in  $Q_1$ , we have  $|P_i|U_i - \frac{\partial \Phi_i}{\partial t} \geq \varepsilon U_i$ ; therefore the equality must hold inside  $S$ , since  $\varepsilon U_i$  is always non-negative and zero only if  $U_i = 0$ . Similarly, for  $\varepsilon U_i = 0$  to hold when  $i \in Q_2 \cup Q_3$ ,  $U_i$  must be zero too. Thus, inside  $S$  we have  $U_i = 0$  and  $\phi_i = \phi_{nh_i} \forall i$ . The condition  $U_i = 0$  holds only when  $\mathbf{q}_i = \mathbf{q}_{id}$ , i.e. when each aircraft  $i$  has reached its target position  $\mathbf{q}_i$ . Finally, by (13) and the condition  $\phi_i = \phi_{nh_i} \forall i$ , we deduce that the set  $S$  reduces to the singleton  $\{\mathbf{n} : (\mathbf{q}_i = \mathbf{q}_{id} \forall i) \wedge (\phi_i = \phi_{id} \forall i)\}$ , i.e. all the aircraft reach their destinations with the desired heading angle.  $\square$

## B MPC and NFs

### B.1 Model dynamics

For the aircraft dynamics, a hybrid point mass model following the dynamics in [27] is used. Those dynamics for level flight can be simplified to:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{V} \\ \dot{\psi} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} V \cos(\psi) + W_1 \\ V \sin(\psi) + W_2 \\ -\frac{C_D S \rho V^2}{2m} + \frac{1}{m} T \\ \frac{C_L S \rho V}{2m} \sin(\phi) \\ -\eta T \end{bmatrix}, \quad (23)$$

where  $X$  and  $Y$  denote the aircraft position in the horizontal plane,  $V$  the true aircraft airspeed,  $\psi$  is the heading angle,  $m$  the mass and  $\phi$  the bank angle of the aircraft,  $T$  is the engine thrust,  $S$  is the surface area of the wings,  $\rho$  is the air density,  $\eta$  is the fuel flow coefficient and  $C_D$ ,  $C_L$  are aerodynamic lift and drag coefficients whose values depend on aircraft type and configuration. Noise enters through the wind ( $W_1$  and  $W_2$ ), which is unbounded and has correlation and distribution properties according to [12]. A stable, hybrid controller for  $\phi$  and  $T$ , such that the aircraft follows a given flight plan is presented in [27].

The model (23) is too complex to use with navigation functions. For this purpose these dynamics are abstracted by the following kinematic equations:

$$\dot{\mathbf{q}}_i = \begin{bmatrix} u_i \cos \theta_i \\ u_i \sin \theta_i \end{bmatrix} \quad (24a)$$

$$\dot{\theta}_i = \omega_i \quad (24b)$$

where  $u_i$  is the longitudinal (linear) and  $\omega_i$  the angular velocity of vehicle  $i$  and  $\mathbf{q}_i = [x_i \ y_i]^T$  and  $\theta_i$  denote the position and orientation of the vehicle.

The navigation function for each aircraft  $i$  used in this approach is:

$$\Phi_i = \frac{\gamma_{di} + f_i}{((\gamma_{di} + f_i)^k + H_{nhi} \cdot G_i \cdot \beta_{0i})^{1/k}}. \quad (25)$$

The above Navigation Function is constructed as explained in detail in [16]. Briefly, the function  $G_i$  reflects the proximity to any possible collisions involving vehicle  $i$ :  $G_i$  is zero when vehicle  $i$  participates in a conflict, i.e. when the sphere occupied by agent  $i$  intersects with other agents' spheres, and takes positive values away from any conflicts, while  $\gamma_{di} = \|\mathbf{q}_i - \mathbf{q}_{id}\|^2$  is the distance from the destination position  $\mathbf{q}_{id}$ . The function  $f_i = f_i(G_i)$  is necessary in a decentralized approach as it is used in proximity situations in order to ensure that  $\Phi_i$  attains positive values even when agent  $i$  has reached its destination.  $\beta_{0i}$  is the workspace bounding obstacle. The factor  $H_{nhi}$  is used to align the trajectories at the origin with the desired orientation  $\theta_{di}$ :

$$H_{nhi} = \epsilon_{nh} + n_{nhi} \quad (26)$$

$$n_{nhi} = ([\cos \theta_i \ \sin \theta_i] \cdot (\mathbf{q}_i - \mathbf{q}_{id}))^2 \quad (27)$$

where  $\epsilon_{nh}$  is a small positive constant. Finally,  $k$  is a positive tuning parameter for this class of Navigation Functions.

It can be shown that this navigation function has proven navigation properties i.e. it provides global convergence to the destination along with guaranteed collision avoidance [14].

For the given navigation function, each vehicle  $i$  is then governed by the following control law [39]:

$$u_i = -\text{sgn}(\mathbf{P}_i) \cdot \mathbf{F}_i - \left( \frac{\partial \Phi_i}{\partial t} + \left| \frac{\partial \Phi_i}{\partial t} \right| \right) \frac{1}{2\mathbf{P}_i} \quad (28a)$$

$$\omega_i = -k_{\theta_i}(\theta_i - \theta_{\mathbf{nh}i}) + \dot{\theta}_{\mathbf{nh}i} \quad (28b)$$

where

$$\begin{aligned} \mathbf{F}_i &= k_u \cdot \|\nabla_i \Phi_i\|^2 + k_z \cdot \|\mathbf{q}_i - \mathbf{q}_{id}\|^2 \\ \mathbf{P}_i &= \mathbf{J}_{Ii}^T \cdot \nabla_i \Phi_i \\ \mathbf{J}_{Ii} &= \mathbf{J}_{Ii}(\theta_i) = [\cos \theta_i \quad \sin \theta_i]^T \\ \nabla_i \Phi_j &= \frac{\partial \Phi_j}{\partial \mathbf{q}_i} \\ \frac{\partial \Phi_i}{\partial t} &= \sum_{j \neq i} u_j \nabla_j \Phi_i^T \cdot \mathbf{J}_{Ij} \end{aligned}$$

and  $k_u$ ,  $k_z$ ,  $k_{\phi_i}$  are positive real gains. The angle  $\theta_{\mathbf{nh}i}$  is the angle of the gradient  $\nabla \Phi_i$ . The control law for  $u_i$  and  $\omega_i$  is completely decentralized and only requires measurement of the current state and knowledge of the target destination of all other agents.

The problem with the decentralized controller (28a) and (28b) is that there is no way to enforce input constraints on speed, turning rate, etc. To enforce such operational constraints, we will use MPC. For notational simplicity, we also define  $u_i[k] \triangleq \{u_i(t), t \in [kT, (k+1)T)\}$ ,  $\forall k = 0, \dots, N-1$ . We denote by  $N$  the horizon of the Mid Term CR algorithm (in minutes), by  $\mathbf{q}_{id}^F$  the desired final configuration of each aircraft  $i$ , by  $\bar{\mathbf{q}}_{id} = [\mathbf{q}_{id}[1] \ \mathbf{q}_{id}[2] \ \dots \ \mathbf{q}_{id}[N]]^T$  and  $\bar{\theta}_{id} = [\theta_{id}[1] \ \theta_{id}[2] \ \dots \ \theta_{id}[N]]^T$  the desired configuration at each time step of the horizon and by  $\bar{u}_i = [u_i[0] \ u_i[1] \ \dots \ u_i[N-1]]^T$  the longitudinal velocities during all intermediate periods of the horizon. Then, Mid Term CR problem for  $m$  aircraft, solved by MPC at each time step, can be described as:

$$\begin{aligned} \min_{\bar{\mathbf{q}}_{1d}, \dots, \bar{\mathbf{q}}_{md}, \bar{\theta}_{1d}, \dots, \bar{\theta}_{md}} \quad & J(\bar{\mathbf{q}}_1, \dots, \bar{\mathbf{q}}_m) \\ \text{subject to} \quad & (24)-(28) \quad \forall i \\ & \bar{u}_i \in [u_{\min}, u_{\max}] \quad \forall i \end{aligned} \quad (29)$$

This problem is not convex, because of (24)-(28). To tackle the non-convex nature of the problem, the MPC optimization will be carried out by a randomized optimization algorithm to determine the intermediate targets for the navigation functions at each time step. The algorithm we use is a variation of Simulated Annealing, based on Markov Chain Monte Carlo (MCMC) methods [23]. Of course, since our control has a receding horizon policy, at every time  $t$ , the optimal inputs for the time instants  $t, t+T, \dots, t+(N-1)T$  have to be calculated, but only the first will be applied. In such a formulation the problem size grows exponentially with the horizon  $N$ . We therefore choose only to optimize over the first intermediate destination  $\mathbf{q}_{id}[1]$ ,  $\theta_{id}[1]$  and then assume that this will be just moved forward in the same direction for the rest of the horizon, i.e.  $\mathbf{q}_{id}[k] = \mathbf{q}_{id}[k-1] + \mathbf{q}_i[k-1] - \mathbf{q}_i[k-2]$  and  $\theta_{id}[k] = \theta_{id}[1]$ ,  $\forall k \in \{2, \dots, N\}$ . Due to uncertainties and conflict resolution maneuvers, aircraft might not arrive at their exact final destination, thus we will consider that aircraft reach their destination when the Euclidean distance is less than some tolerance value  $\Delta$ .

Finally, we return to the question of how can the navigation function controls  $u_i$  and  $\omega_i$  be

translated to the corresponding FMS inputs. This is done through

$$T = \begin{cases} C_T T_{Max} & \text{if } u_i + \delta > V \\ 0.95 T_{Max} & \text{if } u_i - \delta < V \\ \frac{C_D S \rho}{2} u_i^2 & \text{else} \end{cases} \quad (30a)$$

$$\dot{\psi} = \omega_i \quad (30b)$$

where  $T_{Max}$  and  $C_T$  are parameters depending on the aircraft type and flight phase of the aircraft [?] and  $\delta$  a small tolerance to avoid chattering around the nominal airspeed.

## B.2 Decentralized MPC

We assume that aircraft solve their trajectories sequentially in a round-robin fashion, i.e. after all aircraft have found a solution, in the next round they solve the problem in the same order. In this case, the optimization problem (29) for each aircraft  $j$  is transformed to:

$$\begin{aligned} \min_{\bar{\mathbf{q}}_{jd}, \dots, \bar{\mathbf{q}}_{md}, \bar{\theta}_{jd}, \dots, \bar{\theta}_{md}} \quad & J(\bar{\mathbf{q}}_j) \\ \text{subject to} \quad & (24)-(28) \quad \forall i \\ & \bar{u}_i \in [u_{\min}, u_{\max}] \quad \forall i \end{aligned} \quad (31)$$

Taking into consideration the alternative of changing the optimization order at each step, the resulting MPC algorithm used is outlined in Algorithm 1.

---

### Algorithm 1 Decentralized MPC algorithm

---

**Require:**  $\mathbf{q}_i(t), t = 0$  and  $\mathbf{q}_{id}^F, \forall i \in \{1, \dots, m\}$

- 1: **while**  $\exists i$  s.t.  $\|\mathbf{q}_i(t) - \mathbf{q}_{id}^F\|_2 > \Delta$  **do**
  - 2:   Fix a priority for the aircraft
  - 3:   **for**  $j = 1$  to  $m$  **do**
  - 4:     Solve problem (31) for
  - 5:     Broadcast  $\bar{\mathbf{q}}_{jd}$  to all aircraft
  - 6:   **end for**
  - 7:   Evolve the system according to (23) and (30) from  $t$  to  $t + T$
  - 8:   Set  $t = t + T$
  - 9:   Measure new aircraft position  $\mathbf{q}_i(t)$
  - 10: **end while**
- 

### B.2.1 Cooperative cost

Finally, we present how the “fairness” factor  $\alpha$  is taken into account for the simulations. In this case, the cost for aircraft  $j$  is modified to:  $J(\bar{\mathbf{q}}_j, \dots, \bar{\mathbf{q}}_m) = \|\mathbf{q}_j(t + NT) - \mathbf{q}_{jd}^F\|_2^2 + \alpha \sum_{i=j+1}^m \|\mathbf{q}_i(t + NT) - \mathbf{q}_{id}^F\|_2^2$ . We only take into account the effect to the aircraft next in the decision round, as previous aircraft have already announced their solutions. It is easy to see that setting  $\alpha = 0$ , aircraft solve the problem as in the previous cases, while  $\alpha = 1$  makes the first aircraft at each round to solve exactly the centralized problem.

## C Distributed Robust MMPC

We recap now the problem formulation which was introduced in [42].

### C.1 System Model

We model the aircraft as point mass particles, and consider a planar problem representing aircraft flying at constant altitude. The  $n$ 'th aircraft has position

$$r_n(t) = [r_{x_n} \ r_{y_n}]$$

and velocity

$$v_n(t) = [v_{x_n} \ v_{y_n}]$$

in ground-fixed orthogonal axes ( $x$ :East,  $y$ :North for example), and is subject to control input accelerations

$$u_n = [f_{x_n} \ f_{y_n}].$$

The model is time-discretised with zero order hold and converted to standard state space form with subsystem state

$$x_n(k) = [r_{x_n}(kT) \ r_{y_n}(kT) \ v_{x_n}(kT) \ v_{y_n}(kT)]^T$$

and control input

$$u_n(k) = [f_{x_n}(kT) \ f_{y_n}(kT)]^T$$

where the sampling and update interval is  $T$ . The dynamics are thus

$$x_n(k+1) = A_n x_n(k) + B_n u_n(k). \quad (32)$$

We assume that there are  $m$  aircraft, so that  $n \in N_m$  in the equations above.

The disturbances act on the state and are assumed to be unknown but bounded, with known bounds, so that

$$w_n(k) \in \mathcal{W} \ \forall \ n, k \quad (33)$$

with

$$\mathcal{W} = \{w(k) : \|w(k)\|_\infty \leq W_{\max}\}. \quad (34)$$

It is assumed that the states are perfectly measured at each time step.

### C.2 Problem Formulation

#### C.2.1 Nominal Constraints

Given a minimum separation distance  $R$ , we have the following positional constraints:

$$\mathbb{X} := \{x_k \in \mathbb{R}^{4m} : \|[I_2 \ 0](x_p(k) - x_q(k))\|_2 > R\} \quad \forall \ k \ \forall \ p, q \in N_m, p \neq q \quad (35)$$

Representing the individual aircraft states as a single combined state, by stacking the state vectors  $x_n(k) \in \mathbb{R}^4$  to obtain  $x(k) \in \mathbb{R}^{4n}$ , the constraints coupling the aircraft can be expressed as

$$x(k) \in \mathbb{X}. \quad (36)$$

In addition to the collision avoidance constraints, each aircraft is subject to local minimum and maximum speed constraints:

$$Cx_n(k) \in \mathbb{Y}, u_n(k) \in \mathbb{U} \forall k, n \in N_m \quad (37)$$

where the matrix  $C = [0 \ I_2]$  extracts velocity information from the full state. The speed constraint is specifically of the form

$$V_{\min} \leq \|v_n(k)\|_2 \leq V_{\max}. \quad (38)$$

These specifications require circular regions of avoidance, and the velocity vectors are constrained to lie outside circular regions as shown below.

We replace these nonlinear constraints by linear constraints obtained by approximating the circular regions by circumscribing polygons. We use the simplest polygonal approximation, namely rectangular regions, to keep the number of binary variables low.

For example the *exclusion* region corresponding to the separation constraints is approximated by

$$\tilde{\mathbb{X}} = \{\vec{x} = (x_1 \dots x_m) \in \mathbb{R}^{4m} : \|[I_2 \ 0](x_p - x_q)\|_\infty < R\} \quad (39)$$

The motivation for these approximations is that the constraints can be represented as linear inequality constraints, which allows the problem to be formulated as a MILP problem. The nonconvex collision avoidance and speed constraints in (37) and (39) are enforced using a ‘big-M’ formulation [5]. The use of mixed integer programming for obstacle avoidance has been employed for instance in [40].

Details of the implementation of the variable horizon formulation with constraint tightening as a mixed integer linear program (MILP) are presented in [31] for the single aircraft case.

As the terminal target region constraints are tightened along the prediction horizon, in addition to state and input constraints, a dependence on the point in the prediction horizon exists, warranting the introduction of notational dependence on  $i$  for the target set constraints.

### C.2.2 Variable horizon

We also define *target regions* which aircraft are required to reach after the completion of any required conflict resolution manoeuvres. These are represented as rectangular regions, yielding linear convex constraints. The target region  $S_n$  allocated to aircraft  $n$  is defined as

$$S_n := \{r_n : Qr_n \leq q_n\}. \quad (40)$$

As the requirement is for the aircraft simply to reach their target regions, and not remain therein, the target set need not be an invariant set. Instead of adopting a stabilising MPC formulation, it is more appropriate to use a variable (shrinking) horizon formulation, whereby the horizon length is itself a variable in the optimisation to be solved. This yields a non-convex formulation, but non-convexities are already present in the constraints, so do not significantly add to the the number of binary variables required. The point in the prediction horizon at which the target region set constraints are active, i.e. the length of the horizon, is determined by a binary input decision variable to be minimised within the MILP.

This permits each aircraft to reach its target region at a time which is not specified as a ‘target time’. Details of this hybrid formulation developed by [31] are presented next.

When the binary variable  $t_n(k) \in \{0, 1\}$  takes the value 1, aircraft  $n$  is required to enter its target region  $S_n$  at the next time  $k + 1$ , so that

$$r_n(k + 1) \in S_n \quad (41)$$

This constraint is enforced using a *big-M* [5] formulation, with the following constraint coupling the binary input variable  $t$  to the system:

$$Qr_n(k+i+1|k) \leq q_n(i) + 1 \cdot \mu(1 - t_n(k+i|k)) \quad (42)$$

where  $\mu$  is a large positive integer. If  $t_n(k+i|k) = 1$ , the target region constraint defined in (40) is active, and met at the next step in the prediction horizon:

$$t_n(k+i|k) = 1 \Rightarrow r_n(k+i+1|k) \in S_n. \quad (43)$$

The cost function to be minimised by each aircraft  $n$  is a combination of the time  $T_{n,f}$  to reach its target region  $S_n$  and the weighted one norm in control input

$$V(k) = \sum_{i=0}^{T_{n,f}} (\gamma |u_n(k+i|k)| + 1). \quad (44)$$

It is shown in [31, 32] how minimisation of (44) is achieved by minimisation of the hybrid objective

$$V_n(k) = \sum_{i=0}^{N-1} (\gamma |u_n(k+i|k)| + i \cdot t_n(k+i|k)) \quad (45)$$

where  $N$ , identified by the user, denotes the maximal horizon time and provides an upper bound on the arrival time  $T_{n,f}$ ; the second term in the objective,  $i \cdot t_n(k+i|k)$ , counts the number of time steps the aircraft remains outside the target region, going to zero on entry to the target set, from (42) and hence  $\sum_{i=0}^{N-1} i \cdot t_n(k+i|k) = T_f$ . The cost in (44) is equivalent to that in (45) if  $u(k+i|k) = 0$  and  $t(k+i|k) = 0 \forall i > T_f$ , which can be ensured by relaxing operational constraints on entering the target region, as will be described next. We impose the terminal constraint that the target set is reached by the end of the prediction horizon,

$$\sum_{i=0}^{N-1} t_n(k+i|k) = 1. \quad (46)$$

For the purposes of guaranteeing that all aircrafts eventually reach their destination targets, it is necessary to relax the state and input constraints associated with an aircraft in (37), including avoidance constraints (36), once it is predicted to have reached its destination target. This ensures that zero control inputs in the prediction sequence are admissible once all the targets are predicted to have been reached. The cost incurred by the tail of the prediction sequence beyond this point would then be zero

The operational constraints are relaxed using a big-M formulation as follows. Given  $n_y$  polyhedral constraints acting on an aircraft, originally of the form

$$\mathcal{Y}_n(i) = \{y_n : p_c^T y_n \leq s_c(i) \forall c \in N_{n_y} := \mathbb{Z}_{1,\dots,n_y}\} \quad (47)$$

with modifications applied for the variable horizon implementation they become

$$p_c^T y_n(k+i|k) \leq s_c(i) + \tilde{\mu} \sum_{l=0}^{i-1} t_n(k+l|k) \quad (48)$$

so that the constraints are relaxed if  $T_{n,f} < l$ , ie if the arrival time is shorter than  $l$ . A similar procedure is performed with the collision avoidance constraints.

### C.2.3 Terminal Set

Defining

$$z_n := 1 - \sum_{i=0}^{N-1} t_n(k+i|k),$$

the terminal set can be expressed as

$$\mathcal{T}_n = \{x_n, z_n : z_n = 0\}$$

where the state  $x_n$  is unrestricted. The associated sequence of terminal control laws  $\kappa_{f,\sigma(k)} = 0$ . The set is invariant, as at any step  $k$ ,  $z_n(k) = 0 \Rightarrow z_n(k+1) = 0$ , as the relaxation of the constraints in (48) ensures that the minimiser is obtained at  $t_n(k+i|k), u_n(k+i|k) = 0 \forall i > T_{n,f}$ .

### C.3 Multiplexed MPC

In multiplexed MPC, the aircraft predictions and control moves are updated in a sequential and cyclic manner, with only one aircraft's control input updating at any one time. Without loss of generality, we use the indexing function

$$\sigma(k) = (k \bmod m) + 1 \quad (49)$$

to identify the aircraft updating at time instant  $k$ . The single update feature of multiplexed MPC is captured by the following constraint on the  $n$ th control input  $u_n(k) \in \mathbb{R}^{n_u}$ ,

$$u_n(k) = 0 \text{ if } n \neq \sigma(k). \quad (50)$$

We refer to the discretisation interval length as  $\tau$ , and assuming each aircraft or channel updates once in an update interval period of length  $T$ , we have  $T = m\tau$ .

The dynamics of the system according to the original multiplexed algorithm, can be expressed as a linear periodic time varying system with composite state  $x(k) \in \mathbb{R}^{n_s}$ , and  $m$  inputs  $\{u_n(k)\}_{n \in N_m}, \tilde{B}_{\sigma(k)} \in \mathbb{R}^{n_s \times n_u}$ ,

$$x(k+1) = \tilde{A}x(k) + \tilde{B}_{\sigma(k)}u_{\sigma(k)}(k) \quad (51)$$

where the one-step matrices for the system are given by

$$\tilde{A} = \begin{bmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_m \end{bmatrix}, \tilde{B} = \begin{bmatrix} B_1 & 0 & \dots & 0 \\ 0 & B_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & B_m \end{bmatrix},$$

$$\tilde{B}_{\sigma(k)} = \begin{bmatrix} 0 \\ \vdots \\ B_{\sigma(k)} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

## C.4 Wind disturbance

In our ATM example, the wind velocity can be modelled as a sum of a deterministic disturbance and stochastic component corresponding to unknown deviations from a nominal prediction [3] obtained from meteorological predictions known in advance. The wind experienced by any aircraft at a given time is correlated to the wind it experienced at earlier times, and we include this in the model. The unknown disturbance levels are thus reduced. The wind disturbance is now included as an additional state variable, augmented with the original state, and we assume wind disturbance dynamics of the form:

$$w_n(k+1) = \alpha w_n(k) + \epsilon_n(k) \quad (52a)$$

$$\alpha \in \mathbb{R}, 0 < \alpha < 1$$

$$\epsilon_n(k) \in \mathcal{E} \forall k, \mathcal{E} = \{\epsilon_n(k) : \|\epsilon_n(k)\|_\infty \leq E_{max}\} \quad (52b)$$

with  $E_{max} < W_{max}$ .

## C.5 Multiplexed MPC with Disturbance Feedback

The proposed disturbance feedback modification to the original multiplexed algorithm is now described. In this variant, changes in control input are performed at every timestep. The non-optimising aircraft adopt their candidate feasible plans based on their predicted plans made at the last time they performed an optimisation. Control inputs are applied to each subsystem at every time step, but the renewal of control policy applied is multiplexed. The system now evolves according to

$$x(k+1) = \tilde{A}x(k+1) + \tilde{B}_{\sigma(k)}u_{\sigma(k)}(k) + \sum_{n \in N_m \setminus \sigma(k)} \tilde{B}_n u_n(k)$$

where at time  $k$ , aircraft  $\sigma(k)$  applies the first part of its newly optimised input sequence, whilst non-optimising aircrafts  $n \in N_m \setminus \sigma(k)$  execute a fixed disturbance rejection policy in between updates, to enable tracking of a nominal trajectory. The proposed feedback policy applied between updates is an affine function of the sequence of past disturbances, so that

$$u(i) = v(i) + \sum_{j=0}^{i-1} P_{i,j} w(j) \quad \forall i \in \mathbb{Z}_{[0, N-1]} \quad (53)$$

where each  $P_{i,j} \in \mathbb{R}^{n_u \times n_x}$  and  $v_i \in \mathbb{R}^{n_u}$

Defining

$$P := \begin{bmatrix} 0 & \dots & \dots & 0 \\ P_{1,0} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ P_{N-2,0} & \dots & P_{N-1, N-2} & 0 \end{bmatrix},$$

the strictly lower block triangular matrix  $P \in \mathbb{R}^{Nn_u \times Nn_x}$ , we can express the control input sequence as

$$\mathbf{u} = \mathbf{v} + P\mathbf{w} \quad (54)$$

with

$$\mathbf{u} = [u(k)^T \ u(k+1)^T \ \dots \ u(k+N-1)^T]^T \quad (55a)$$

$$\mathbf{v} = [v(k)^T \ v(k+1)^T \ \dots \ v(k+N-1)^T]^T \quad (55b)$$

$$\mathbf{w} = [w(k)^T \ w(k+1)^T \ \dots \ w(k+N-1)^T]^T. \quad (55c)$$

In between updates, aircrafts apply the disturbance rejection policy in (53) so that the control input corresponding to an aircraft is given by

$$u(k+i) = v(k+i|k) + \sum_{j=0}^{i-1} P_{i,j} w(k+j) \quad (56)$$

and  $i$  is the number of steps since the aircraft last performed an optimisation.

We show now how the sensitivity parameter  $\tilde{L}(i)$ , relating the deviations in the candidate policy from the nominal control sequence to the ensuing perturbations to state predictions evolves within the prediction horizon with the introduction of disturbance feedback in between updates for a general time-varying disturbance feedback gain  $K(j)$ . We discuss the specific choice of disturbance feedback gain  $K(j)$  later.

If we adopt a state feedback control law correction policy  $u_n(j) = K_n(j)x_n(j)$ , time-varying in the general case, the corresponding state transition matrix for the closed loop system  $L_n(j)$  is given by

$$L_n(j+1) = (A_n + B_n K_n(j))L_n(j) \quad (57)$$

with  $L_n(0) = I$ . A possibility for the disturbance feedback policy matrix  $P_n$  is given by

$$P := \begin{bmatrix} 0 & \dots & \dots & 0 \\ K_n(0) & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ K_n(N-2)L_n(N-2) & \dots & K(0) & 0 \end{bmatrix}$$

so that

$$u_n(k+i) = v_n(k+i|k) + \sum_{j=0}^{i-1} K_n(j)L_n(j)w_n(k+i-j-1) \quad (58)$$

If we define the state transition matrices for the composite system  $\tilde{L}(i)$  under the control  $u_n(j) = K_n(j)x_n(j)$  for all  $j$

$$\tilde{L}(i) := \begin{bmatrix} L_1(i) & 0 & \dots & 0 \\ 0 & L_2(i) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & L_m(i) \end{bmatrix}$$

where  $L_n(\cdot)$  evolves along the prediction horizon according to (57), and define the matrix of feedback gains

$$\tilde{K}_{\sigma(k)}(i) = \begin{bmatrix} 0 \\ \vdots \\ K_{\sigma(k)}^T(i) \\ 0 \\ \vdots \\ 0 \end{bmatrix}^T,$$

and choose the following parameterisation for the matrix of feedback gains for the composite system  $\tilde{M}_{\sigma(k)}(i)$

$$\tilde{M}_{\sigma(k)}(i) = \tilde{K}_{\sigma(k+i)}(i)\tilde{L}(i) \quad (59)$$

then the state transition matrices  $\tilde{L}(i)$  for the composite system evolve according to

$$\begin{aligned}
\tilde{L}(i+1) &= \tilde{A}\tilde{L}(i) + \sum_{\sigma(k+i) \in N_m} \tilde{B}_{\sigma(k+i)}\tilde{M}_{\sigma(k)}(i) \\
&= \tilde{A}L(i) + \sum_{\sigma(k+i) \in N_m} \tilde{B}_{\sigma(k+i)}\tilde{K}_{\sigma(k+i)}(i)\tilde{L}(i) \\
&= (\tilde{A} + \tilde{B}\tilde{K})\tilde{L}(i).
\end{aligned} \tag{60}$$

with  $\tilde{L}_{\sigma(k)}(i) = \tilde{L}_{\sigma(k+1)}(i) \forall k$ . It is shown in [43] that given

$$x(k+1) = \tilde{A}x(k) + \tilde{B}_{\sigma(k)}u_{\sigma(k)}(k) + \sum_{n \in N_m \setminus \sigma(k)} \tilde{B}_n u_n \tag{61}$$

and

$$u_n(k+i|k) = u_n(k+i|k-1) + \tilde{K}_n(i-1)\tilde{L}(i-1)w(k-1) \forall n \tag{62}$$

Then

$$x(k+i+1|k+1) = x(k+i+1|k) + \tilde{L}(i)w(k). \tag{63}$$

**Constraint Tightening for robustness** For robust feasibility, the state constraint sets must be constructed so that the state predictions remain feasible when perturbed by the amount in (63), on application of the disturbance feedback policy. The state and input constraint sets must be tightened according to

$$\tilde{\mathcal{X}}(i+1) = \tilde{\mathcal{X}}(i) \sim \tilde{L}(i)\mathcal{W} \tag{64a}$$

$$\mathcal{U}_{\sigma(k)}(i) = \mathcal{U}_{\sigma(k+1)}(i-1) \sim \tilde{M}_{\sigma(k+1)}(i-1)\mathcal{W} \tag{64b}$$

$$= \mathcal{U}_{\sigma(k+1)}(i-1) \sim \tilde{K}_{\sigma(k+1)}(i-1)\tilde{L}(i-1)\mathcal{W} \tag{64c}$$

The sequence of sets obtained from propagating the disturbance set  $\mathcal{W}$  along the prediction horizon through the dynamics under the feedback law  $\tilde{K}_{\sigma(k)}$  are given by

$$\mathcal{Z}_{0,i} := \bigoplus_{j=0}^i \tilde{L}(j)\mathcal{W}. \tag{65}$$

We shall refer to  $\{\mathcal{Z}_{0,i}\}_i$  henceforth as ‘state perturbation’ sets. For robust feasibility, the nominal state sets must be tightened along the prediction horizon by the corresponding state perturbation sets, whose sizes are monotonically non-decreasing along the prediction horizon. It follows that a necessary (but not sufficient) condition for feasibility is that the final state perturbation set which must be subtracted from the nominal constraint set at the end of the prediction horizon, given by  $\mathcal{Z}_{0,N-1}$ , is contained in the nominal state constraint set (and hence also the terminal set), so that

$$\mathcal{Z}_{0,N-1} \subset \text{interior}(\mathbb{X}) \tag{66a}$$

$$\tilde{K}\mathcal{Z}_{0,N-1} \subset \text{interior}(\mathbb{U}). \tag{66b}$$

The disturbance feedback gains  $\tilde{K}$  must be appropriately chosen to achieve this.

We define now the centralised problem required to induce subsequent multiplexed optimisations is defined first in Problem C.1.

**Problem C.1.** *Minimise:*

$$V_C(k) = \sum_{n \in N_m} \sum_{i=0}^{N-1} (\gamma |u_n(k+i|k)| + it_n(k+i|k)) \quad (67)$$

with respect to inputs  $u_n(k+i|k)$  and binary inputs  $t_n(k+i|k)$  for all  $n$ , subject to the nominal prediction model and tightened constraints:

$$x(k|k) = x(k) \quad (68a)$$

$$x(k+i|k) \in \mathcal{X}(i) \quad (68b)$$

$$u_n(k+i|k) \in \mathcal{U}_n(i) \quad (68c)$$

$$y_n(k+i|k) \in \mathcal{Y}_n(i) \quad (68d)$$

$$x_n(k+i|k) \in \mathcal{S}_n(i) \quad (68e)$$

$$t_n(k+i|k) \in \{0, 1\} \quad (68f)$$

the constraints coupling the binary to the continuous variables in (42) and the terminal constraint that all targets are reached by the end of the horizon

$$\sum_{n \in N_m} \sum_{i=0}^{N-1} t_n(k+i|k) = m \quad (68g)$$

where 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 (64).

Termination of Algorithm 2 occurs when all aircrafts have reached their targets. Every time an aircraft completes, it is considered to have exited the scenario, and the aircraft count is decremented correspondingly in Line 14. The problem each aircraft solves in the receding

---

**Algorithm 2** 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 (64)
  - 3: Receive centralised solution  $\mathbf{u}_n^*(k_0), \mathbf{t}_n(k_0)$  to problem and apply the first input  $u_n^*(k_0|k_0)$ ;
  - 4: Wait one timestep;  $k = k_0 + 1$ ;
  - 5: **while**  $m - \sum_n t_n(k) > 0$  **do**
  - 6:   **if**  $n = \sigma(k)$  **then**
  - 7:     Obtain  $\min\{\mathbf{u}_{\sigma(k)}^*(k), \hat{\mathbf{u}}_{\sigma(k)}(k)\}$  as the solution to Problem C.2
  - 8:     Transmit plan and state information to all aircrafts;
  - 9:   **else**
  - 10:     Renew current plan according to disturbance feedback policy
  - 11:   **end if**
  - 12:   Increment control input by first step in plan
  - 13:   Wait one timestep,  $k \leftarrow k + 1$
  - 14:    $m \leftarrow m - \sum_n (1 - t_n(k))$
  - 15: **end while**
- 

horizon optimisations is presented in problem C.2.

**Problem C.2.** *Minimise:*

$$V_{\sigma(k)}(k) = \sum_{i=0}^{N-1} (\gamma |u_{\sigma(k)}(k+i|k)| + it_{\sigma(k)}(k+i|k)) \quad (69)$$

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

$$x(k|k) = x(k) \quad (70a)$$

$$x(k+i|k) \in \mathcal{X}(i) \quad (70b)$$

$$u_{\sigma(k)}(k+i|k) \in \mathcal{U}_{\sigma(k)}(i) \quad (70c)$$

$$y_{\sigma(k)}(k+i|k) \in \mathcal{Y}_{\sigma(k)}(i) \quad (70d)$$

$$x_{\sigma(k)}(k+i|k) \in \mathcal{S}_{\sigma(k)}(i) \quad (70e)$$

$$t_{\sigma(k)}(k+i|k) \in \{0, 1\}, \quad (70f)$$

the constraint on the non-optimising aircrafts

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

the constraints coupling the binary to the continuous variables in (42) and the terminal constraint in (46). 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 (64).

### C.5.1 Fixed Order MMPC

As with the original MMPC, we use a cyclical optimisation timing, using  $\sigma(k) = k \bmod n$  as the ‘arbitrating function’ referencing the aircraft updating at time  $k$ .

### C.5.2 Variable Update Order MMPC

For the MMPC formulation we have just described, there is no notion of order in the open loop initial centralised optimisation, and subsequently no requirement on the policy update order of aircraft. We propose a variable update order formulation, whereby all aircraft optimise in parallel for new policies, but a decision on which aircraft changes its policy at any one time is based on satisfaction of some global objective. A number of possible heuristics exist for determining the optimal sequencing. The updating aircraft is chosen to be that which would yield the maximum reduction in the global cost, given that only one aircraft is permitted to execute its newly optimised plan. The sequencing function is now no longer not a function of time, but depends on the current state, the candidate feasible plans of all aircraft formed from their previous plans, and their hypothetical new plans obtained from their respective optimisations performed at the current time step.

Defining  $\hat{\mathbf{u}}_p(k)$  to be the candidate feasible sequence for aircraft  $p$  formed from the tail of the previous solution

$$\hat{\mathbf{u}}_p(k) = \{\mathbf{u}_p(k|k-1) + w_p(k-1), \dots, \mathbf{u}_p(k+N-1|k-1) + K_p(N-1)L_p(N-1)w_p(k-1)\},$$

and the collection of candidate feasible sequences as

$$\hat{\mathbf{u}}(k) = \{\hat{\mathbf{u}}_1(k), \dots, \hat{\mathbf{u}}_m(k)\} \quad (71)$$

and the set of input sequences obtained from the respective solutions to Problem C.2 by each aircraft  $n$  at time  $k$  as

$$\mathbf{u}^*(k) = \{\mathbf{u}_1^*(k) \dots \mathbf{u}_m^*(k)\}, \quad (72)$$

we can express the sequencing function  $\sigma(x(k), \mathbf{u}^*(k), \hat{\mathbf{u}}(k))$  as

$$\sigma(x(k), \mathbf{u}^*(k), \hat{\mathbf{u}}(k)) = \underset{n}{\operatorname{argmax}} (V(x(k), \hat{\mathbf{u}}_{p \in N_m}(k)) - V(x(k), \mathbf{u}_n^*, \hat{\mathbf{u}}_{p \in N_m \setminus n}(k))). \quad (73)$$

### C.5.3 Choice of feedback gains

After  $n$  steps of constraint tightening, we now have a disturbance set

$$Z_n := \bigoplus_i^n L_i \mathcal{W} \quad (74)$$

$$= \bigoplus_{i=0}^n (A + BK)^i \mathcal{W}. \quad (75)$$

$$Z_\infty =: Z \quad \forall k \quad (76)$$

A number of possibilities for stabilising  $K$  now exist. In this work we use a nilpotent controller, obtained from  $(A + BK)^2 = 0$ , to keep the number of Pontryagin differences small in the constraint tightening, yielding

$$K = \left[ \frac{1}{T^2} I, -\frac{3}{2T} I \right]. \quad (77)$$

## D Prioritized MPC

### D.1 Problem formulation

#### D.1.1 Dynamics

Consider  $I$  aircraft flying within an area of interest, with the following simplified continuous-time dynamics for level flight [27]:

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{v}_i \\ \dot{\psi}_i \\ \dot{m}_i \end{bmatrix} = \begin{bmatrix} v_i \cos(\psi_i) + w_i^x \\ v_i \sin(\psi_i) + w_i^y \\ -\frac{C_{i,D} S_i \rho}{2} \frac{v_i^2}{m_i} + \frac{1}{m_i} T_i \\ \frac{C_{i,L} S_i \rho}{2} \frac{v_i}{m_i} \sin(\phi_i) \\ -\eta_i T_i \end{bmatrix} \quad (78)$$

for  $i \in \mathcal{I} \triangleq \{1, \dots, I\}$ , where  $p_i \triangleq [x_i \ y_i]^\top$  denotes the aircraft position in the horizontal plane,  $v_i$  the true aircraft airspeed,  $\psi_i$  is the heading angle,  $m_i$  the mass,  $\phi_i$  is the bank angle,  $T_i$  is the engine thrust,  $S_i$  is the surface area of the wings,  $\rho_i$  is the air density,  $\eta_i$  is the fuel flow coefficient, and  $C_{i,D}$ ,  $C_{i,L}$  are aerodynamic drag and lift coefficients, respectively, whose values depend on aircraft type and configuration. Noise enters the system via the wind elements  $w_i^x$  and  $w_i^y$  which are correlated Gaussians, as described in [12]. For simplicity the effect of wind on the accelerations and turning moments is neglected here. It can, however, be introduced in the simulation through the so-called wind gradient factors, as shown in [27].

Since the dynamics (78) are highly nonlinear, solving a conflict resolution problem using these dynamics is computationally intractable. Thus, we abstract the dynamics (78) to a linear discrete-time model, based on single integrator dynamics, as follows:

$$p_i(t+1) = p_i(t) + hu_i(t) + hw_i(t), \quad (79)$$

where  $p_i(t) = [x_i(t) \ y_i(t)]^\top$ ,  $w_i(t) = [w_i^x(t) \ w_i^y(t)]^\top$ ,  $u_i(t) = [u_i^x(t) \ u_i^y(t)]^\top$  is the velocity input, and  $h$  is the sampling period. We denote by

$$\tilde{p}_i(t+1) = \tilde{p}_i(t) + hu_i(t), \quad (80)$$

the dynamics (79) in the absence of wind, and by  $\bar{p}_i(t)$  the nominal discrete-time flight plan (see Figure 31). The nominal trajectory  $\bar{p}_i(t)$  is computed using an ideal straight flight at nominal speed from the current point to the destination  $p_i^d$ .

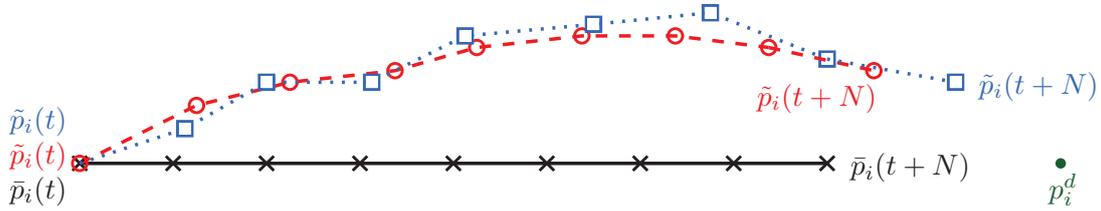


Figure 31: Possible scenario of the nominal flight plan  $\bar{p}_i(t)$ , the simplified model dynamics  $p_i(t)$ , and the wind-free dynamics  $\tilde{p}_i(t)$

#### D.1.2 Velocity Constraints

The aircraft have dynamical constraints on their true airspeed, as well as their turning rate. Corresponding to the simplified dynamics (79), we associate the following two constraints

pertaining to the admissible inputs and their rates of change

$$\begin{aligned} \|u_i(t)\|_1 &\geq u_{\min} \\ \|u_i(t)\|_\infty &\leq u_{\max} \\ \|u_i(t+1) - u_i(t)\|_\infty &\leq \delta u \end{aligned} \tag{81}$$

for all  $t \in \mathbb{N}$ . In general, the input constraints are given in terms of 2-norms and are present to ensure that the aircraft do not stall or fly with extremely high velocities. However, this would lead to an alternative set of constraints in (81), which would be quite inefficient to handle in terms of computation. We have chosen to relax the lower constraint on the inputs into an inner 1-norm constraint and the upper constraint on the inputs into an outer  $\infty$ -norm constraint. Similarly, the originally 2-norm input rate constraint is relaxed into an  $\infty$ -norm one. This leads to a more tractable optimization problem, as discussed next, and the relaxations can be further refined using polytopic norms at the expense of more computational effort. The first constraint in (81) is non-convex; however, it can be implemented using the so-called big-M technique and 4 binary variables, transforming the first input constraint in (81) to

$$\begin{aligned} u_i^x(t) + u_i^y(t) &\geq u_{\min} - M_u c_i^1(t) \\ u_i^x(t) - u_i^y(t) &\geq u_{\min} - M_u c_i^2(t) \\ -u_i^x(t) + u_i^y(t) &\geq u_{\min} - M_u c_i^3(t) \\ -u_i^x(t) - u_i^y(t) &\geq u_{\min} - M_u c_i^4(t) \\ \sum_{\nu=1}^4 c_i^\nu(t) &\leq 3, \quad c_i^\nu(t) \in \{0, 1\} \end{aligned} \tag{82}$$

where  $M_u$  is a sufficiently large number. The last constraint in (82) ensures that at least one of the inequality constraints is active, and consequently that the speed remains above the desired minimum.

### D.1.3 Separation Constraints

With the most critical factor in Air Traffic Control being safety, it is natural to pose constraints that guarantee conflict avoidance. Given a minimum separation among the aircraft  $\Delta$  (typically taken to be 5nm for en-route flights), we pose the following set of constraints at the sample instants:

$$\|p_i(t) - p_j(t)\|_2 \geq \Delta, \tag{83}$$

for all  $t \in \mathbb{N}$  and  $i, j \in \mathcal{I}$  with  $i \neq j$ . However, the constraint (83) does not guarantee that the inter-sample trajectories of (79) do not come closer than the allowed separation  $\Delta$ . This issue can be addressed by looking at a finer time grid: between any two time samples  $t$  and  $t+1$  we take  $L$  subintervals  $\left\{t, t + \frac{t}{L}, t + \frac{2t}{L}, \dots, t + \frac{(L-1)t}{L}\right\}$  on which we test the constraint (83), i.e., we require the satisfaction of the following set of constraints:

$$\left\| p_i(t) - p_j(t) + \frac{lh}{L} [u_i(t) + w_i(t) - u_j(t) - w_j(t)] \right\|_2 \geq \Delta, \tag{84}$$

for all  $t \in \mathbb{N}$ ,  $l \in \{1, \dots, L\}$ , and  $i, j \in \mathcal{I}$  with  $i \neq j$ . Unfortunately, as we have assumed that the noise variables  $w_i(t)$  and  $w_j(t)$  are normally distributed the constraint (84) becomes impossible to satisfy for all possible noise realizations. Instead, we choose to robustly satisfy the separation

constraint with a high level of confidence. Using the triangle inequality, we can conservatively approximate the constraint (84) by

$$\left\| \tilde{p}_i(t) - \tilde{p}_j(t) + \frac{lh}{L}(u_i(t) - u_j(t)) \right\|_2 \geq \Delta + \Delta_{i,j}(t, l), \quad (85)$$

where  $\tilde{p}_i(t)$  corresponds to the dynamics (80) and

$$\Delta_{i,j}(t, l) = \frac{lh}{L} \|w_i(t) - w_j(t)\|_2. \quad (86)$$

We simply choose to robustly satisfy the constraints (85) for all noise realizations in the 99.7% confidence interval, i.e., for all  $w_i^x(t), w_i^y(t), w_j^x(t), w_j^y(t) \in [-3\sigma, 3\sigma]$ . With this choice, we obtain the following upper bound

$$\Delta_{i,j}(t, l) \leq 6\sigma h \frac{l}{L}. \quad (87)$$

This upper bound assumes that the noise sequences are uncorrelated in space and may render the constraint (85) too conservative.

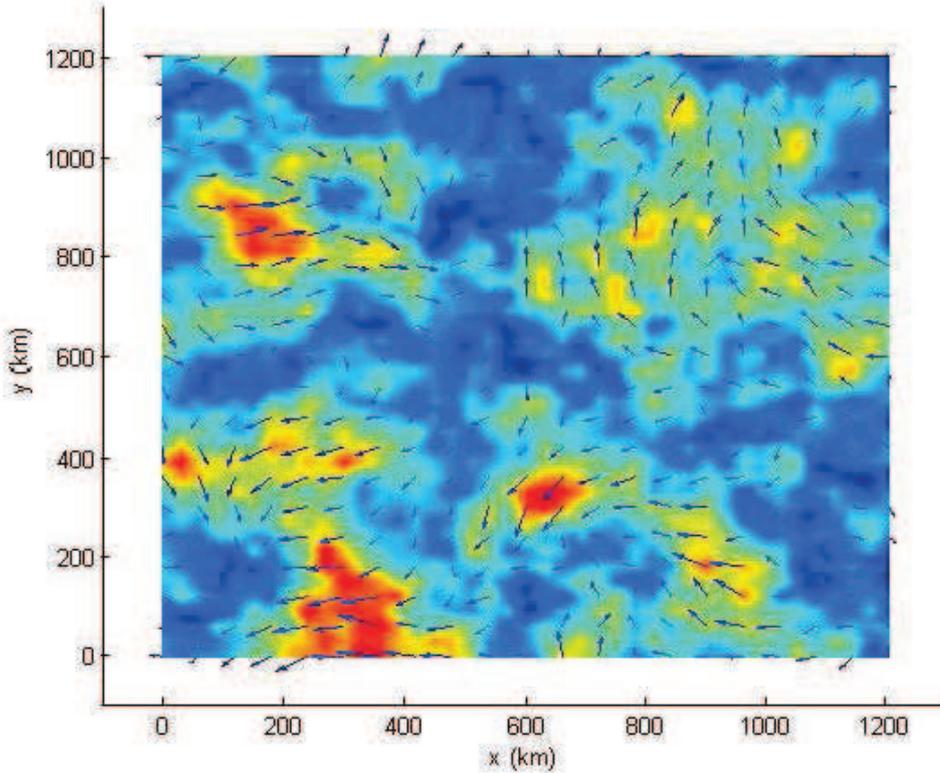


Figure 32: Correlated wind field

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

$$\rho_{xy}(\|\tilde{p}_i(t) - \tilde{p}_j(t)\|_2) = -0.006 + 1.006e^{-\frac{\|\tilde{p}_i(t) - \tilde{p}_j(t)\|_2}{337000}}, \quad (88)$$

where  $\|\tilde{p}_i(t) - \tilde{p}_j(t)\|_2$  is the horizontal separation in meters between two aircraft. As depicted in Figure 32, and since the wind speed enters in a linear fashion in our simplified model dynamics, the closer the aircraft are, the smaller the difference in the wind that the two aircraft experience, and, consequently, the smaller the uncertainty on their relative position. Utilizing (88), the difference in the wind speeds experienced by the two aircraft in (86) becomes normally distributed as  $(w_{i,x}(t) - w_{j,x}(t)) \sim N(0, \tilde{\sigma}_{i,j}(t))$ , where  $\tilde{\sigma}_{i,j}(t) = \sigma \sqrt{2 - 2\rho_{xy}(\|\tilde{p}_i(t) - \tilde{p}_j(t)\|_2)}$ . Since  $\tilde{\sigma}_{i,j}(t)$  depends in a non-convex fashion on the inputs via the dynamics (80), using this exact constraint is not possible. Instead, we use

$$\bar{\sigma}_{i,j}(t, l) = \sigma \sqrt{2 - 2\rho_{xy}(\|\bar{p}_i(t) - \bar{p}_j(t)\|_2 + h \frac{l}{L} \delta v)}, \quad (89)$$

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  $\delta v$  is a constant related to the maximum change of the airspeed magnitude at each step. Following the same procedure as in equation (87), we obtain the less restrictive

$$\bar{\Delta}_{i,j}(t, l) = 6\bar{\sigma}_{i,j}(t, l), \quad (90)$$

with  $\Delta_{i,j}(t, l) \geq \bar{\Delta}_{i,j}(t, l)$ .

Using this less conservative approximation on the ‘robustifying’ factor  $\Delta_{i,j}$ , we look again at the non-convex constraint (85) and tighten it, in a similar way as in (81), using the norm inequality  $\|\cdot\|_2 \geq \|\cdot\|_\infty$  and the formulation in [30] as:

$$\begin{aligned} \tilde{e}_{i,j}^x(t, l) &= \tilde{x}_i(t) - \tilde{x}_j(t) + \frac{lh}{L}[u_i^x(t) - u_j^x(t)] \\ \tilde{e}_{i,j}^y(t, l) &= \tilde{y}_i(t) - \tilde{y}_j(t) + \frac{lh}{L}[u_i^y(t) - u_j^y(t)] \\ \tilde{e}_{i,j}^x(t, l) &\geq +\Delta + \bar{\Delta}_{i,j}(t, l) - M_{\bar{p}}d_{i,j}^1(t) \\ \tilde{e}_{i,j}^x(t, l) &\leq -\Delta - \bar{\Delta}_{i,j}(t, l) + M_{\bar{p}}d_{i,j}^2(t) \\ \tilde{e}_{i,j}^y(t, l) &\geq +\Delta + \bar{\Delta}_{i,j}(t, l) - M_{\bar{p}}d_{i,j}^3(t) \\ \tilde{e}_{i,j}^y(t, l) &\leq -\Delta - \bar{\Delta}_{i,j}(t, l) + M_{\bar{p}}d_{i,j}^4(t) \\ \sum_{\nu=1}^4 d_{i,j}^\nu(t) &\leq 3, \quad d_{i,j}^\nu(t) \in \{0, 1\} \end{aligned} \quad (91)$$

where  $M_{\bar{p}}$  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  $x$  or  $y$  axes.

## D.2 Prioritized Hierarchical MPC Solution

### D.2.1 Higher level controller - MPC

In a conflict resolution setting, one may not consider all aircraft having the same priority (see for instance the related work in [19]). Priorities in such a situation might signify timing constraints, e.g. how far behind schedule aircraft are, fuel constraints, etc. In our setting, inspired by the concept of operations proposed by the iFly project[13], higher priority aircraft do not deviate from their flight plan, unless all lower priority aircraft cannot resolve the conflict without them deviating. We assume that the  $I$  aircraft are ordered by increasing priority according to their indices  $\{1, \dots, I\}$ , i.e., the aircraft with index  $I$  has the highest priority. Guided by the setup in [22], define  $I$  binary variables  $\delta_1, \dots, \delta_I$ , one for each aircraft. Given the nominal flight

plan sequence  $\bar{p}_i(t+k)$  for  $k \in \{1, \dots, N\}$ , define the following set of *deviation* constraints corresponding to the  $i$ -th aircraft:

$$\begin{aligned} \|\tilde{p}_i(t+k) - \bar{p}_i(t+k)\|_\infty &\leq \epsilon_i(t+k) \\ 0 &\leq \epsilon_i(t+k) \leq M_\epsilon \delta_i, \end{aligned} \quad (92)$$

where  $M_\epsilon$  is a finite constant. The constraint (92) penalizes any deviation by the model (80) due to the designed control inputs away from the nominal flight plan. If a deviation occurs, the binary variable  $\delta_i$  is set to one and results in a higher cost. Given the optimization horizon  $N$  we define the cost

$$\begin{aligned} \mathcal{J}(t) = & \underbrace{\sum_{i=1}^I \left\| [\epsilon_i(t+1) \quad \epsilon_i(t+2) \quad \dots \quad \epsilon_i(t+N)] \right\|_1}_{\text{relaxation of constraints}} + \beta \underbrace{\sum_{i=1}^I 2^{i-1} \delta_i}_{\text{priorities}}, \end{aligned} \quad (93)$$

where  $\beta$  is a positive scalar given by  $\beta = NM_\epsilon + 1$ . This choice of  $\beta$  ensures that the priorities part of the cost dominates the relaxation of constraints part. Moreover, given the specific structure of weighting, the various binary variables ensure that the satisfaction of higher priority constraints always results in a larger cost reduction than any possible combination of the lower priority constraints. In our 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.

For the separation constraints, we need to take into consideration the fact that as we advance in the horizon steps, aircraft may have deviated more from their nominal flight plan. Thus, using (90), we get:

$$\bar{\Delta}_{i,j}(t+k|t, l) = 6\bar{\sigma}_{i,j}(t, kL + l), \quad (94)$$

which will be subsequently used in the MPC formulation.

Upon substituting the dynamics (80) into the constraints (91), and by utilizing (94), we obtain the following set of separation constraints along the the optimization horizon  $N$ :

$$\begin{aligned} \tilde{e}_{i,j}^x(t+k, l) &= \tilde{x}_i(t) - \tilde{x}_j(t) + \sum_{\tau=t}^{t+k-1} (u_i^x(\tau) - u_j^x(\tau)) \\ &\quad + \frac{lh}{L} (u_i^x(t+k) - u_j^x(t+k)) \\ \tilde{e}_{i,j}^y(t+k, l) &= \tilde{y}_i(t) - \tilde{y}_j(t) + \sum_{\tau=t}^{t+k-1} (u_i^y(\tau) - u_j^y(\tau)) \\ &\quad + \frac{lh}{L} (u_i^y(t+k) - u_j^y(t+k)) \\ \tilde{e}_{i,j}^x(t+k, l) &\geq +\Delta + \bar{\Delta}_{i,j}(t+k|t, l) - M_{\bar{p}} d_{i,j}^1(t) \\ \tilde{e}_{i,j}^x(t+k, l) &\leq -\Delta - \bar{\Delta}_{i,j}(t+k|t, l) + M_{\bar{p}} d_{i,j}^2(t) \\ \tilde{e}_{i,j}^y(t+k, l) &\geq +\Delta + \bar{\Delta}_{i,j}(t+k|t, l) - M_{\bar{p}} d_{i,j}^3(t) \\ \tilde{e}_{i,j}^y(t+k, l) &\leq -\Delta - \bar{\Delta}_{i,j}(t+k|t, l) + M_{\bar{p}} d_{i,j}^4(t) \\ \sum_{\nu=1}^4 d_{i,j}^\nu(t) &\leq 3, \quad d_{i,j}^\nu(t) \in \{0, 1\}, \end{aligned} \quad (95)$$

for all  $k \in \{1, \dots, N\}$ . Having defined all the required constraints, the main finite-horizon optimization problem to be solved at each time  $t \in \{0, 1, 2, \dots\}$  is given by

$$\min_{\substack{u_i(t) \\ i \in \{1, \dots, I\} \\ t \in \{0, \dots, N-1\}}} \left\{ \mathcal{J}(t) \mid (80), (82), (89), (90), (92), (95) \right\}. \quad (96)$$

Problem (96) is an MILP, and hence can be solved effectively for a reasonable traffic scenario.

### D.2.2 Lower level controller - autopilot

Once the optimization problem (96) is solved, the resulting control inputs for each aircraft  $u_i(t), \dots, u_i(t + N - 1)$  are pushed down to the autopilot. Consequently, the autopilot generates for the first sampling period  $h$  the following thrust and bank angle inputs

$$T_i = \begin{cases} C_T T_{\max} & \text{if } \|u_i(t)\|_2 + \delta_{\text{tol}} > v_i \\ 0.95 T_{\max} & \text{if } \|u_i(t)\|_2 - \delta_{\text{tol}} < v_i \\ \frac{C_D S \rho}{2} \|u_i(t)\|_2^2 & \text{else} \end{cases} \quad (97)$$

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

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

where  $T_{\max}$  and  $C_T$  are parameters depending on the aircraft type and flight phase of the aircraft (see [? ]),  $\delta_{\text{tol}}$  a small tolerance to avoid chattering around the nominal airspeed and  $k_1, k_2$  design parameters of the bank angle controller. As the linear controller  $\phi_i^1$  may command unrealistically large bank angles, we introduce the following saturation at a given angle  $\frac{\pi}{6}$ :

$$\phi_i^2 = \min\{\max\{\phi_i^1, -\frac{\pi}{6}\}, \frac{\pi}{6}\}. \quad (98)$$

Despite the saturation, aircraft may travel in circles in case they deviate too far from their reference path. To prevent this, a further limit, dependent on the heading error is introduced, leading to the final setting for the bank angle:

$$\phi_i = \begin{cases} \min\{\phi_i^2, 0\}, & \pi/2 \geq \psi \geq \frac{\pi}{3} \\ \max\{\phi_i^2, 0\}, & \pi/2 \geq -\psi \geq \frac{\pi}{3} \end{cases} \quad (99)$$

The hybrid controller described in (97) through (99) stabilizes the simplified continuous-time dynamics (78) as shown in [27].

### D.2.3 Overall Hierarchical MPC formulation

The overall proposed scheme is summarized in Algorithm 3.

---

**Algorithm 3** Prioritized Hierarchical MPC Algorithm

---

**Require:**  $p_i(t), t = 0$  and  $p_i^d, \forall i \in \{1, \dots, I\}$

- 1: **while**  $\exists i$  s.t.  $\|p_i(t) - p_i^d\|_2 > \Delta$  **do**
  - 2:   Solve the MPC problem (96)
  - 3:   Evolve the system according to the autopilot and aircraft dynamics (97), (98), and (99) in the interval  $[th, (t + 1)h[$
  - 4:   Set  $t = t + 1$
  - 5:   Measure new aircraft position  $p_i(t)$
  - 6: **end while**
-