Examples of Supervisory Interaction with Route Optimizers

Oliver Turnbull*†, Arthur Richards*‡
*Department of Aerospace Engineering, University of Bristol
Queens Building, University Walk, Bristol, BS8 1TR, UK
†Research Assistant, Email: oliver.turnbull@bristol.ac.uk
‡Lecturer, Email: arthur.richards@bristol.ac.uk

***Abstract*—As part of the SUPEROPT SESAR WP-E research project we are interested in providing ways of interacting with trajectory optimizers. In particular, this paper will develop a 3-D aircraft performance model and then concentrate on giving a supervisor the ability to select their desired "sense" of conflict resolution between multiple aircraft; time is included in the model to allow one aircraft to pass "ahead" or "behind" another. A nonlinear model with equivalent sense constraints is also developed to facilitate the inclusion of fuel use or potentially noise and emissions in future developments.**

The linear model is applied to a large scale problem and a tool is presented to facilitate exploration of the solution space created by the available sense constraints and additionally by different cost/objective functions before "committing" to a specific solution.

Foreword

This paper describes a project that is part of SESAR Work Package E, which is addressing long-term and innovative research.

I. Introduction

There has been a large volume of research into optimal trajectory generation [1], [2], collision avoidance [3], [4], and aircraft flow modelling [5], [6]. However, the SESAR Concept of Operations [7] calls for "Extensive use of automation support to reduce operator task load, but in which controllers remain in control as managers". The "SUPErvision of Route OPTimizers" (SUPEROPT) project aims to create optimizers that facilitate human interaction.

Air Traffic Control (ATC) encompasses a range of requirements: some are hard constraints, e.g. aircraft performance and collision avoidance, while others may be soft constraints that are influenced by user preferences, e.g. minimum cost or minimum time trajectories. One of the themes of the SUPEROPT project is developing methods to allow a human supervisor to alter the behaviour of a trajectory optimizer to include a supervisor's preferences based on their perception of the current state of the airspace.

Previous work [3], [5] has addressed the fundamental problem of optimal trajectories and collision avoidance. In this paper, the model and ideas introduced in [8] regarding the "sense" of a conflict are extended to the 4-D case and, in particular, we consider what is meant when we state that we wish to resolve a conflict vertically (by altitude) or temporally (one aircraft to pass ahead/behind another). Furthermore, we investigate how a controller may explore the solution space of such constraints.

The first part of this paper adopts Mixed-Integer Linear Programming (MILP) [3]–[5] to solve the global, non-convex conflict resolution optimization. MILP captures the discrete decision making within the problem, such as collision avoidance, with binary decision variables. It has been chosen here as it is extensible and, with CPLEX software [9], reliable to solve. An aircraft performance model is developed before additional constraints are defined such that we can control the sense of conflict resolutions, e.g. flight A passes above flight B.

Nonlinear optimization has also been proposed for trajectory generation [10], [11], although convergence can be a challenge and global optimality is not guaranteed. The second part of this paper takes the collocation method proposed in [12] and the obstacle avoidance method of Patel and Goulart based on polar sets [10] to develop sense constraints with a non-linear model. Furthermore, polar set based obstacle avoidance is shown to generalize well to the concept of 4-D obstacles such as a temporarily closed sector.

A key contribution is the development of constraint forms that capture high-level behaviours, such as the sense of a conflict resolution, without overly constraining low-level behaviours, e.g. the full 4-D trajectory. Thus the approach aims to allow intuitive human input in terms of high-level decision making while still enabling the optimizer to do what it does best: designing efficient 4-D trajectories subject to avoidance constraints.

The remainder of this paper is organised as follows: Section II introduces the nomenclature used throughout the remainder of the paper; Section III provides an introduction to the details of the problem addressed by the paper; Section IV develops a linear approximation to the 3-D aircraft model of BADA [13] and proposes a scheme to constrain the sense of conflict resolution within the model; Section V outlines an equivalent scheme for a non-linear dynamics model and introduces obstacle avoidance in 4-D; Section VI applies the earlier results to a large scale air traffic problem and presents a tool to explore the problem space and multiple cost/ojective functions; finally some concluding remarks are made in Section VII.

II. Nomenclature

The nomenclature used in this paper is defined in Table I.

TABLE I
NOMENCLATURE

Symbol	Definition
N_a	Number of aircraft
a	Index of aircraft
$\mathbf{r}(t,a)$	Position of a at time t (decision variable)
$r_d(t,a)$	Position of a at time t in dimension d
$\mathbf{r}^R(t,a)$	Reference trajectory of a at time t (fixed parameter)
$\ddot{\mathbf{r}}(t,a)$	Acceleration of a at time t
k	Index of time step
t_k	Sample time k
N_t	Number of times-steps
$t_f^R(a)$	Final/exit time of a on reference trajectory (fixed parameter)
$t_f(a)$	Final/exit time of a (decision variable)
$b_L(a,k,m)$	Binary variable used to identify current altitude "band", m, of aircraft a at time k
$\lambda_L(a,k,m)$	Proportional distance from breakpoint m in PWA Function for aircraft a at time k

III. Problem Description

For the purpose of SUPEROPT, we define an air-traffic role of Multi-Sector Controller (MSC) responsible for the detailed 4-D trajectories of each aircraft and maintaining a safe separation between all aircraft at all times in a Multi-Sector Area (MSA). SUPEROPT aims to design a tool to support such a role and to show that a tool/model can be devised with sufficient flexibility to be relevant to multiple problems (on different scales).

First we define a cost function that aims to capture mathematically the metrics of performance that the MSC wishes to "minimise", but without any specific upper limit on their acceptable values. The cost should be made as small as possible subject to the constraints.

Fig. 1 illustrates a typical problem instance. We define N_a aircraft in the MSA and assume that each aircraft a has Reference Business Trajectory (RBT) $\mathbf{r^R}(t,a)$ running from $t = t_0$, the current time, to $t = t_f^R(a)$, the reference time at which the RBT ends; the immediate destination of aircraft a is defined by $\mathbf{r^R}(t_f^R(a),a)$. This would typically refer to the pre-determined point, e.g. in the Shared Business Trajectory (SBT), at which the aircraft is expected to exit the MSA. The optimizer designs for each aircraft a trajectory $\mathbf{r}(t,a)$ from time t_0, i.e the current time, to $t_f(a)$, the new chosen time at which a exits the MSA. Finally, since a numerical optimizer can only have a finite number of constraints, define discrete time step variable k to index a set of N_t sampling times between t_0 and $t_f(a)$. Constraint and cost evaluation will be performed at these points.

With the scenario defined we define our cost function as:

$$J = \sum_{a=1}^{N_a}\sum_{d=1}^{N_d}\left[\begin{array}{c}\alpha t_f(a)+\\ \beta\big(r_d(t_f(a),a) - r_d^R(t_f(a),a)\big)^2+\\ \gamma\ddot{r_d}(t_k,a)+\\ \delta\sum_{k=1}^{N_t}\left|r_d(t_k),a) - r_d^R(t_k,a)\right|\end{array}\right] \quad (1)$$

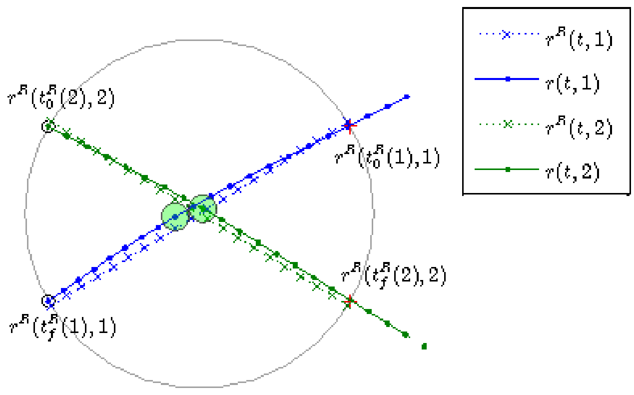


Fig. 1. Example of trajectory definitions

where: the first term with a weighting on the final time reflects the desire to avoid delay; the second term penalizes coordinations with the adjoining MSA and long term deviations from the RBT with a weighting on deviation from the exit point; the third time is included to reduce manoeuvring and increase passenger comfort with a weighting on acceleration $\ddot{r}$; and the final term reflects the desire to stay close to the RBT throughout the MSA.

The relative importance of the different terms is adjusted via the weights $(\alpha, \beta, \gamma, \delta)$. Section VI presents initial results on the exploration of different weightings.

IV. Linear 3-D Aircraft Trajectory Model with Sense Constraints

A. Aircraft Performance Model

In order to extend the idea of 2-D sense constraints [8] to 3-D while maintaining realistic aircraft dynamics, it is necessary to introduce a performance model to the trajectory generator/optimization.

The EUROCONTROL Base of Aircraft Data (BADA) provides both an analytical model and a database of aircraft performance for typical commercial aircraft. The BADA User Manual [13] states that the longitudinal and normal acceleration for civil airliners is limited to 2 and $5fps^2$ respectively.

One approach persued by the SUPEROPT project is to adopt Mixed Integer Linear Programming (MILP) to solve for a globally optimal set of trajectories while enforcing additional logical constraints. Trajectory generation using this method requires the use of a global frame of reference. Consequently, the longitudinal and normal accelerations have been approximated as horizontal and vertical respectively:

$$\ddot{r}_1(k,a)\cos\left(i\frac{2\pi}{N_c}\right) + \ddot{r}_2(k,a)\sin\left(i\frac{2\pi}{N_c}\right) \leq A_v \quad (2)$$

$$\ddot{r}_3(k,a) \leq A_h \quad (3)$$

$$\forall\ k \in \{1,\ldots,N_t-1\}, a \in \{1,\ldots,N_a\}, i \in \{1,\ldots,N_c\}$$

where N_a is the total number of aircraft; N_t is the number of timesteps; N_c is the number of constraints used to approximate the acceleration magnitude; a_w is the acceleration in direction w; A_h and A_v are the horizontal and vertical acceleration limits respectively.

This approximation is reasonable given the relatively small angle of attack and nominal bank angles of civil airliners.

To model individual aircraft dynamics more precisely, the Rate Of Climb/Descent (ROCD) has been limited according to BADA. Taking the data for a typical aircraft (Airbus A319), operating at its nominal weight, it is clear that the permitted ROCD is dependent on flight regime and level (Figure 2 [13]) and that this data can be approximated by suitable Piecewise Affine functions (PWA) for climb and descent.

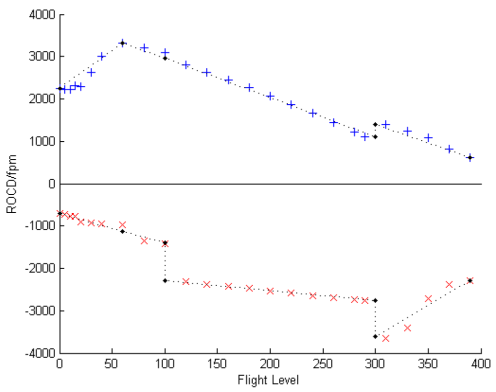


Fig. 2. Piecewise affine function to approximate the ROCD of an A319

The PWA can be simply implemented as a MILP as follows:

$$\sum_{m=1}^{N_L-1} b_L(a,k,m) = 1 \quad (4)$$

$$\sum_{m=1}^{N_L} \lambda_L(a,k,m) \geq 0 \quad (5)$$

$$\forall \ a \in \{1,\ldots,N_a\}, k \in \{0,\ldots,(N_t-1)\},$$

$$\sum_m \lambda_L(a,k,m) = \begin{cases} b_L(a,k,m), \\ \quad for\ m = 1 \\ b_L(a,k,m-1) + b_L(a,k,m), \\ \quad for\ m \in 2,\ldots,N_L(a) \\ b_L(a,k,N_L(i)-1), \\ \quad for\ m = N_L(a) \end{cases} \quad (6)$$

$$\sum_m \lambda_L(a,k,m) B_L(a,m) = r_z \quad (7)$$

$$v_z(a,k) \leq \sum_m \lambda_L(a,k,m) A_C(a,m) \quad (8)$$

$$v_z(a,k) \geq \sum_m \lambda_L(a,k,m) A_D(a,m) \quad (9)$$

where N_L is the number of breakpoints in the ROCD function; $B_L(a)$ is a vector of length N_L where each value is a flight level at which 'the rate of climb or descent function changes' and $A_C(a,m)$ and $A_D(a,m)$ are the maximum rates of climb and descent respectively at the altitudes specified in B_L.

Using a PWA to capture the performance of different aircraft, as shown above, allows us to investigate different constraints to control the sense of a conflict resolution.

B. Spatial Separation

Previous work [8] introduced the idea of constraining the sense of aircraft conflict resolution in 2-D. We could loosely refer to sense as the choice of "side", where the chosen side is indicated by setting a corresponding binary variable to 1, but during some manoeuvres two conflicting aircraft will spend some time on both sides of the other. Instead, [8] observed that the *total* perceived angle change provides a unique differentiator between the two cases. In one case, the line joining the two aircraft moves anticlockwise; in the other clockwise. This idea is also developed in the theory of robot motion planning, in which it is further observed that there are an infinite number of distinct classes of path [14]. The extra paths are achieved by adding multiples of 2π to the angle change, resulting in one aircraft looping around another.

In 2-D it is clear that conflicts between aircraft can be divided into distinct classes of solution, e.g. "left" or "right", or "above" or "below". These classes are referred to here as the sense of the solution.

When we consider the 3-D case, it is apparent that the differences between classes are not distinct as the extra dimension allows for a continuum of paths around an obstacle (assuming it is finite). Consequently, we first ensure that our problem is well defined by restricting the degrees of freedom to ensure distinct classes of solution when we discuss the sense of a conflict. Once we have ensured the problem is well defined, we show how to ensure that the solution belongs to a given class of problem such as "above/below" or "ahead/behind".

Forcing resolution in the desired direction (plane) can be achieved through fixing the trajectory to the initial plan (RBT) in the other directions. For example, to force a horizontal resolution we would fix the aircraft's altitude:

$$r_3(k,a) = r_3^R(k,a) \quad \forall \ : \ k \in \{1,\ldots,N_t\}, a \in \{1,\ldots,N_a\} \quad (10)$$

where $r_3(k,a)$ is the altitude of aircraft a at time-step k. By fixing the RBT in this manner we are effectively reducing the dimensionality of the planning problem, ensuring the presence of distinct solution classes. However, it should be noted that if the supervisor does not require a particular sense of conflict resolution then the model will determine an optimal solution across all dimensions.

Conversely, to enforce vertical resolution we would fix the longitude and latitude of the trajectory:

$$r_d(k,a) = r_d^R(k,a) \quad \forall \ d \in \{1,2\}, k \in \{1,\ldots,N_t\}, a \in \{1,\ldots,N_a\} \quad (11)$$

Fig. 3 shows an example scenario of two crossing aircraft, with one aircraft (F002), climbing up through the flightpath of the other. Each sub-figure shows the trajectories of both aircraft (travelling West to East) projected in the horizontal and vertical planes. The aircrafts' closest approach is highlighted with a cylinder representing half of the conflict distance in each direction, i.e. if the cylinders intersect there is a conflict (in which case they would be shaded red).

Fig. 3(a) shows the case with no additional constraints, i.e. no sense constraints. Separation is achieved through vertical separation (with F001 passing below F002) and the cost is reduced by increasing the velocity of both aircraft so that they arrive at the destination earlier.

Fig. 3(b) shows an alternative option but with the latitude and longitude of the trajectory fixed to force vertical resolution of the conflict and the sense constrained such that F001 passes over F002. It should be noted that in addition to the desired result, neither aircraft accelerates to reach its destination earlier which is a consequence of fixing the horizontal position of the aircraft.

Fig. 3(c) shows the same scenario where we have forced a horizontal resolution. It should be noted that in this case, despite fixing one dimension of the trajectory, the resulting solution has allowed the aircraft to accelerate and reach the target earlier; this occurs as the vertical distance to the destination at the closest approach is sufficiently small that the cost of arriving earlier is lower than the cost of not achieving the precise altitude of the destination at the arrival time-step.

It should be noted that in order to emphasise the different behaviours, the avoidance criteria in all the examples shown in Fig. 3 were defined as 3nm and approximately 2000ft.

Having restricted the class of the problem, the question of resolving the sense of the conflict is now well defined. Perhaps a more intuitive approach to enforcing a particular conflict resolution than the methods proposed in [8] is to fix some of the avoidance binaries in the problem.

We do not wish to force avoidance in a particular direction at all times as this would be too restrictive: consider the case in Fig. 4(a) where there is a square, 2-D obstacle which can be avoided by being: "before", "after", 'above' or 'under' it, i.e. with any sense. If we were to require the aircraft to be "above" or "below" the obstacle at *all* times then the problem would become infeasible (from the current initial position). However, as we also do not know at which time we wish to enforce our desired condition we must find an alternative formulation.

Consider Fig. 4(b), the binary allowing the trajectory to pass "under" the object has been fixed to prohibit this possibility; binaries 1 and 2 have not been fixed as we do not wish to enforce that the path is "above" the obstacle at all times. This leaves us with only trajectories that pass "over" the obstacle. In dense traffic situations, sense might become more complicated to constrain, and then the methods presented in [8] must be employed.

Returning to the case of 3-D, vertical conflict resolution, if we require F001 to go over F002 then we constrain the binaries as follows:

$$b_a(2,1,k,6) = 0 \;\forall\; k \in \{1,\dots,N_t\} \tag{12}$$

where $b_a(2,1,k,6)$ is the avoidance binary between aircraft F002 and F001 at all times, k, in direction "6" (F002 above F001). Setting this binary to zero forces the separation between the two aircraft to be achieved in a different direction at all times, i.e. it is not sufficient to achieve separation only through F002 being above F001 regardless of the distance between them. By fixing the horizontal components of the flightpath as before, then the only class of solution remaining is for F001 to pass over F002 as shown in Fig. 3(b).

C. Temporal Separation

An alternative representation of resolution sense is for A to pass ahead of B (or vice versa). This equates to a constraint that B cannot occupy any point in space that A currently occupies or will occupy in the future. If it did, B could reach the crossing point of the two trajectories before A, violating the sense constraint. Hence, for sense constraints in temporal form, avoidance must be enforced between a pair of vehicles at pairs of different time steps. When combined with the previous method of fixing the vertical or horizontal aspects to the RBT, this provides a powerful formulation to enforce flight A to pass ahead of (or behind) flight B.

Suppose that aircraft a_1 is required to pass ahead of aircraft a_2. It is necessary to introduce avoidance binaries between the pair of aircraft at all pairs of time-steps such that we can then ensure the aircraft remain separated by at least distance D_d in dimension d either side of the obstacle:

$$r_d(a_1,k_1) \le r_d(a_2,k_2) - D_d + M\left(1 - b_a(a_1,a_2,k_1,k_2,d)\right) \tag{13}$$

$$r_d(a_1,k_1) \le r_d(a_2,k_2) + D_d - M\left(1 - b_a(a_1,a_2,k_1,k_2,d+3)\right) \tag{14}$$

$$\forall\; a_1 \in \{1,\dots,N_a\}, a_2 \in \{1,\dots,N_a\}, k_1 \in \{2,\dots,N_t\}, k_2 \in \{2,\dots,N_t\}, d \in \{1,2\} : (k_1 \ge k_2)$$

where the notation is as before except: we now consider avoidance between a_1 and a_2 at time k_1 and k_2 respectively.

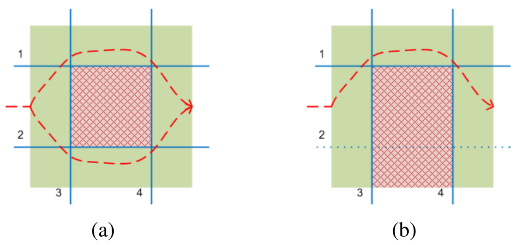
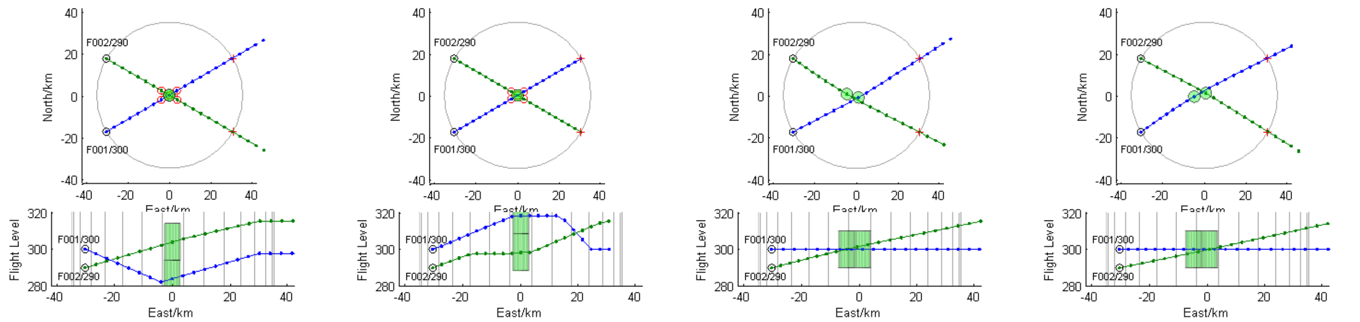


Fig. 4. Example of fixing an avoidance binary



(a) Unconstrained conflict resolution (resolved vertically)

(b) Vertical conflict resolution with constrained sense (F001 over F002)

(c) Horizontal conflict resolution

(d) Horizontal conflict resolution with constrained sense (F002 ahead of F001)

Fig. 3. Example of planar conflict resolution

It should be noted that except for indexing and the number of avoidance binaries, the principal difference between the above equations and the standard MILP avoidance constraints (without the sense conditions) is the extra time-steps at which the constraints are applied, i.e. when $k_1 > k_2$, this forces the avoidance of all future positions of a_1 by a_2.

Finally we must ensure that each pair of aircraft are separated in at least one direction for each time step before they reach their destination:

$$\sum_{m=1}^{6} b_a(a_1, a_2, k_1, k_2, m) = 1 - \sum_{k_3=1}^{k_1-1} b_f(a_1, k_a) - \sum_{k_4=1}^{k_2-1} b_f(a_2, k_4) \quad (15)$$

$$\forall\ a_1 \in \{1, \dots, N_a\}, a_2 \in \{1, \dots, N_a\}, \\ k_1 \in \{2, \dots, N_t\}, k_2 \in \{2, \dots, N_t\} : (k_1 \geq k_2)$$

where the final two summations ensure that we do not plan for aircraft after they reach their destination or exit the MSA.

Fig. 3(d) shows an example where the sense of the conflict resolution has been forced such that F002 passes ahead of F001 compared to the sense free case in Fig. 3(c)

V. Generalization to a Nonlinear Model

Generalization to a nonlinear model is a logical step, enabling more realistic aircraft modelling and allowing us to test the validity of the small angle approximations made in (2) and (3). Developing a nonlinear model also facilitates the inclusion of fuel use in the model and possible emissions and noise modelling as in [15].

Collocation methods approximate the state of an optimal control problem by a basis of polynomials [16] and are an active area of research for problems with hard nonlinear dynamics [17]. The method solves for the coefficients of the Lagrange interpolating polynomial coefficients to the aircraft dynamics. The coefficients can be used to give the planned velocity and position of the vehicle at any time between the start and goal. For the MSC role this is particularly useful as it enables us to derive accurate points for diverting around aircraft/obstacles. The disadvantage compared to MILP is that it does not guarantee a globally optimal solution.

This paper will use the collocation model of [11] to develop a nonlinear sense constraint model.

A. The 4-D Obstacle

We will develop the concept of a "4-D obstacle", drawing on recent work in nonlinear optimizing [10] to define an obstacle both in terms of the space it occupies and the time for which it does so. We will see two applications of this approach: optimizing subject to temporary closure of airspace, and efficient conflict resolution with variable time-scales.

Patel and Goulart [10] advanced the idea of using polar sets for obstacle avoidance. The advantage to representing an obstacle in its polar form is that it transforms the constraints into a differentiable function which allows the use of fast, gradient based methods to solve the optimization, thus reducing solution times. The basic method is reviewed here briefly and extended to include the concept of temporal obstacles: in addition to requiring that all aircraft avoid conflicts with each other, there are times where a controller may wish to enforce all aircraft to avoid a region of airspace, e.g. closed sectors or sectors approaching their capacity limits. Closure of airspace is a temporal as well as spatial event which motivates the idea of 4-D obstacle avoidance.

This section advances the novel idea of applying avoidance in four dimensions, i.e. three spatial dimensions plus time. Since the polar set form provides a flexible mechanism for constraining a vector to be outside a convex set, we apply it in four dimensions to avoid an obstacle that occupies a convex region in space for a defined interval in time.

Here we consider obstacle avoidance using polar sets for a single aircraft planning. First we define a set of points, $\mathbf{y_e}(t) \in \mathbb{R}^4$, that lies within the polar set of the obstacle defined by the (Cartesian + time) vertices, $\mathbf{h}(v)$:

$$\mathbf{h}(v)\mathbf{y_e} \leq 1\ \forall\ v \in \{1, \dots, N_v\}, t \in \{1, \dots, N_t\} \quad (16)$$

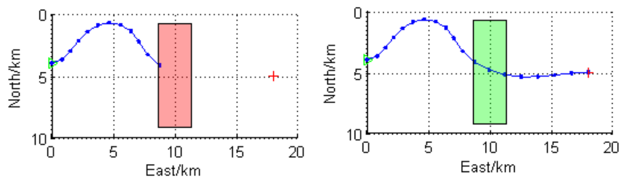
where: N_v is the number of vertices of the obstacle; $\mathbf{h}(v)$ is row v of matrix H^T which defines all the vertices; and N_t is the number of time-points along the trajectory where we enforce the constraints.

Next we ensure that the aircraft remains outside of the obstacle (within the polar set) at each evaluation time point:

$$\mathbf{y_e}(t)^T \begin{pmatrix} \mathbf{r}(t) - \mathbf{r}_{obs}(t) \\ t - t_{obs} \end{pmatrix} \geq 1 \;\forall\; t \in \{2, \ldots, N_t\} \qquad (17)$$

where r_{obs} is the centroid of the obstacle region and t_{obs} is the time in the middle of the interval during which the region is closed.

Applying the above formulation for a single aircraft to a single obstacle we can see the possible effects of the obstacles temporal nature on the aircraft trajectories. Fig. 3 shows an examples of 4-D obstacle avoidance trajectories. The trajectory is shown projected into the X-Y plane at two time points: Fig. 5(a) shows the trajectory during the period that the obstacle must be avoided and Fig. 5(b) shows the complete trajectory including the period where the obstacle no longer needs to be avoided. The trajectory shows a case where the required deviation around the obstacle is so large that it is "cheaper" to wait until such a time that the trajectory can pass straight through the obstacle; the curve in the path prior to entering the obstacle (Fig. 5(a)) is due to the weightings of the cost function favouring a minimum time solution, by constantly accelerating whilst waiting for the obstacle to disappear the time required after that point can then be minimized.

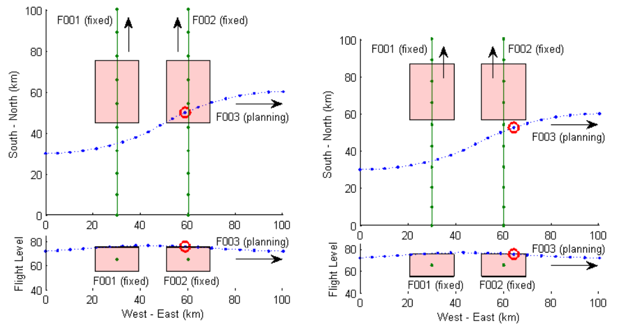


(a) (b)

Fig. 5. Example of a trajectory avoiding a 4-D obstacle

One of the advantages of the collocation method is that the length of the trajectory (finish time) can be expressed as a decision variable. However, a disadvantage is that when considering multiple vehicles, care must be taken to ensure that the constraints are evaluated at temporally congruent points. The introduction of 4-D obstacles provides an alternative to this restriction. If we plan for one vehicle at a time, then fix the aircraft trajectories we can model an aircraft trajectory as a series of spatially fixed, temporal obstacles. If we define an obstacle between each pair of time-steps along a trajectory and extend the obstacle the appropriate distance in all directions (e.g. 5*nmi* and 1000*ft* then we can ensure that the planning aircraft remains a safe distance from the previously planned trajectories.

Fig. 6 shows some examples of a 3-aircraft problem where F001 and F002 have already been fixed and F003 (dotted line) is now planning; the figures are presented again in the horizontal and vertical planes; the current time step of F003 is circled and the obstacles representing F001 and F002 at the current time step are shown as a shaded region (it is noted that 5*nmi* is approximately 9*km*).



(a) F003 crossing "over" F001 and F002 (vertical separation)

(b) F003 crossing "over" F001 and F002 (horizontal separation)

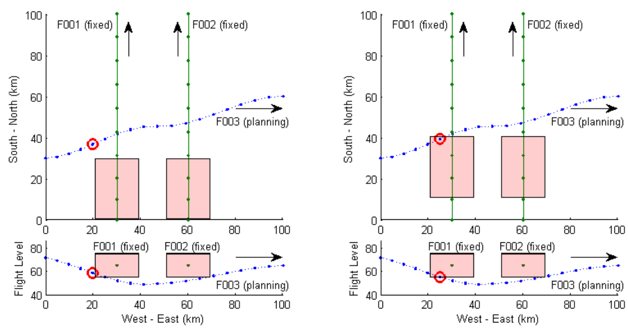
Fig. 6. Example of multi-vehicle conflict resolution using a nonlinear dynamics model

B. Sense Constraints in the Nonlinear Problem

Fixing the MILP avoidance binaries is equivalent to removing vertices from the polar set in the collocation formulation. As the polar set of the obstacle is defined as containing the origin, then if we restrict the location of the polar-point of the aircraft's location such that it must lie to one side of the origin, we force the vehicle to pass to that side of the obstacle. An example of constraining the sense of the conflict is presented in Fig. 7 where we have introduced an additional constraint restricting the location of the polar-point to be less than zero:

$$y(\tau_1, 3, a_1) \leq 0 \;\forall\; \tau_1 \in \{2, \ldots, N_t\} \qquad (18)$$

where $y(\tau_1, 3, a_1)$ is the coordinate of the point in the polar set that corresponds to the vehicles altitude relative to F001.



(a) HF003 crossing "under" F001 and F002 (horizontal separation)

(b) F003 crossing "under" F001 and F002 (vertical separation)

Fig. 7. Example of multi-vehicle conflict resolution with enforced sense constraints using a nonlinear dynamics model

Fig. 7 shows the same scenario as in Fig. 6 but with the additional constraint in (18). After being forced to go "under" F001, the shortest path is to continue also "under" F002.

VI. Multi-Objective

The ideas introduced in this paper lead to a variety of solutions to a given air traffic instance. If the cost function is parametrized to enable scalable weightings on different terms then the number of solutions can increase further still.

To enable a controller to explore the set of solutions it is proposed that a simple interface displaying the cost history of different stages during the planning process along with a "Back" button can be used to select the preferred option before the chosen solution is "Committed" and executed by the affected aircraft until such time that another interaction is required, e.g. unscheduled demand creates another conflict.

Fig. 9 shows a large scale problem based on 78 actual flights passing through the 3 sectors over Wales and the Irish Sea as shown in 8. The data was obtained from FlightRadar24.com for two separate 20 minute periods and superimposed onto each other in order to increase the traffic density and to generate some conflicts (identified in Fig. 9 with a red box around the flight identifier of both aircraft involved).

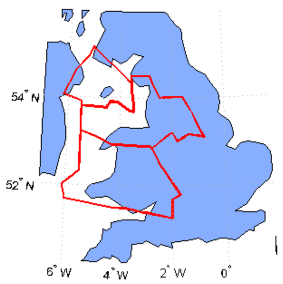


Fig. 8. Extent of the sectors over the UK included in large scale problem

Given a problem instance, we can now explore the solution space by solving the optimization and adding optional additional constraints. Fig. 10 shows "step" 4 of 5 in a planning session. The left half of the figure shows the 6 trajectories that represent the resolution of the previous conflicts (the other flights remain in the model but are not plotted in order to improve clarity of the conflict resolution). The graph in the upper-right of the figure shows the relative cost history for two differently weighted cost functions over a series of planning steps. One cost function penalises deviation from the RBT (input data) and the other is more flexible and broadly aims to minimize time and accelerations.

Table II defines the steps in the planning session of Fig. 10 and presents the solve time for the optimization using both cost functions. Ensuring there are distinct classes of solution, i.e. fixing the problem in one or more dimensions, significantly reduces the solution time as expected. Furthermore, from these limited results, adding the sense constraints appears to have limited impact on solution time; again, this is to be expected as it only affects a small number of binary variables.

VII. Conclusions

A flexible tool for supervised trajectory generation has been presented including a mechanism to enable a supervisor to select the sense of aircraft conflict resolution in 3-D. The sense constraints have been extended to account for time, providing the flexibility to require one aircraft to pass "ahead" or "behind" another. Furthermore, a simple tool for exploring the solution space provided by these constraints has been demonstrated to facilitate supervisor interaction with the optimizer.

Future work on SUPEROPT will consider constraint prioritization in the Air Traffic Flow Management problem as well as performing more post-processing of the optimization output in order to communicate rationale to the supervisor.

TABLE II
Example optimization solve times

	Solve Time (s)	
Planning Step & Constraints	**Min. Deviation**	**Time and Acc.**
1: Free	18.81	73.41
2: Resolve horizontally	0.36	0.75
3: Resolve Vertically	0.36	0.48
4: Vertical + 1 sense constraint	0.34	0.58
5: Vertical + 2 sense constraints	0.34	0.53

References

[1] Y. Kuwata and J. How, "Three dimensional receding horizon control for UAVs," in *AIAA Guidance, Navigation, and Control Conference*, 2004.
[2] F. Borrelli, D. Subramanian, A. U. Raghunathan, and L. T. Biegler, "Milp and nlp techniques for centralized trajectory planning of multiple unmanned air vehicles," in *Proceedings of the 2006 American Control Conference*, IEEE, 2006.
[3] A. Richards and J. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," in *American Control Conference, 2002. Proceedings of the*, pp. 1936–1941, 2002.
[4] A. Richards and J. How, "Decentralized model predictive control of cooperating UAVs," in *43rd IEEE Conference on Decision and Control*, IEEE, 2004.
[5] D. Bertsimas, G. Lulli, and a. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Operations Research*, vol. 59, pp. 211–227, Mar. 2011.
[6] G. Clare and A. Richards, "Air traffic flow management under uncertainty: Application of chance constraints," in *Application and Theory of Automation in Command and Control Systems*, 2012.
[7] Eurocontrol, "Sesar concept of operations,"
[8] A. Richards, "Constraining the sense of conflict resolution: Supervision of route optimization," in *SESAR Innovation Days*, 2011.
[9] IBM, "IBM ILOG CPLEX website," March 2012.
[10] R. B. Patel and P. J. Goulart, "Trajectory generation for aircraft avoidance maneuvers using online optimization," *Journal of Guidance, Control, and Dynamics*, vol. 34, pp. 218–230, JanuaryFebruary 2011.
[11] A. Eele and A. Richards, "Path-planning with avoidance using nonlinear branch-and-bound optimization," *Journal of Guidance, Control, and Dynamics*, vol. 32, pp. 384–394, Mar. 2009.
[12] F. Fahroo and M. I. Ross, "Direct trajectory optimization by a chebyshev pseudospectral method," in *Proceedings of the American Control Conference*, 2000.
[13] Eurocontrol, "Base of aircraft data," 2010.
[14] J. Latombe, *Robot Motion Planning*. Kluwer international series in engineering and computer science: Robotics, Springer, 1990.
[15] R. F. de Oliveira and C. Büskens, "Benefits of optimal flight planning on noise and emissions abatement at the frankfurt airport," in *AIAA Guidance, Navigation, and Control Conference*, 2012.
[16] B. Fornberg, *A practical guide to pseudospectral methods,*, vol. 1 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, 1996.
[17] Q. Gong, F. Fahroo, and M. Ross, "Spectral algorithm and for pseudospectral and methods and in optimal and control," in *Journal of Guidance, Control, and Dynamics*, vol. 31, May–June 2008.

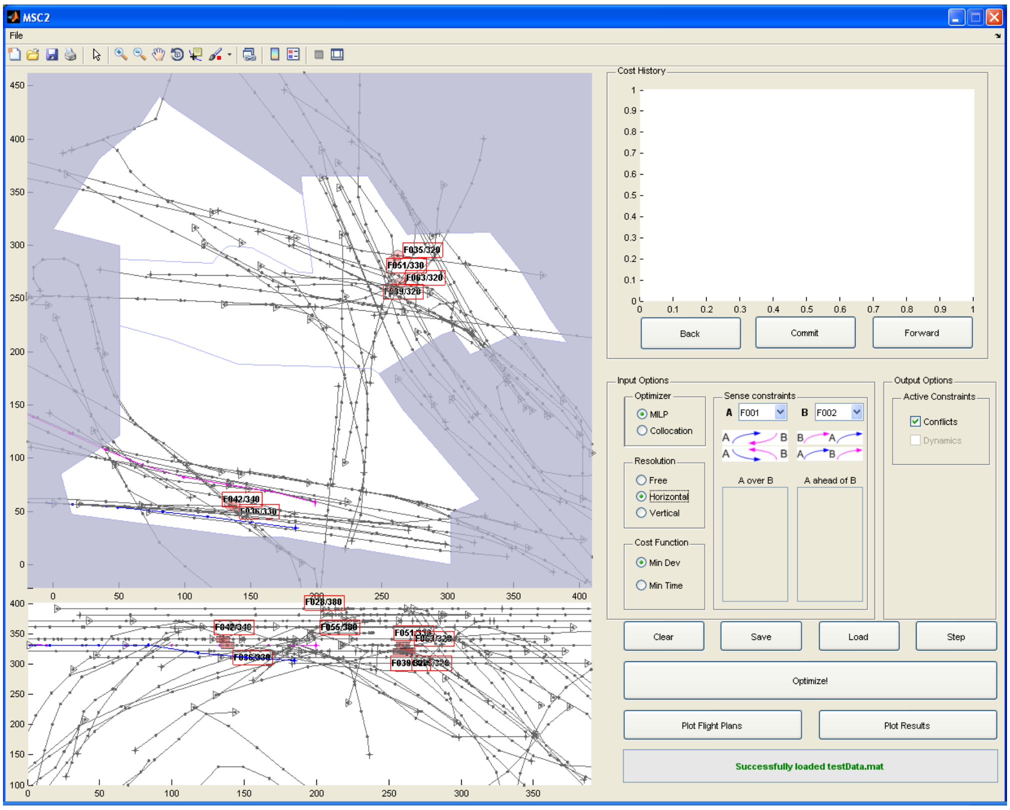


Fig. 9. Example traffic through 3 UK sectors with increased traffic density

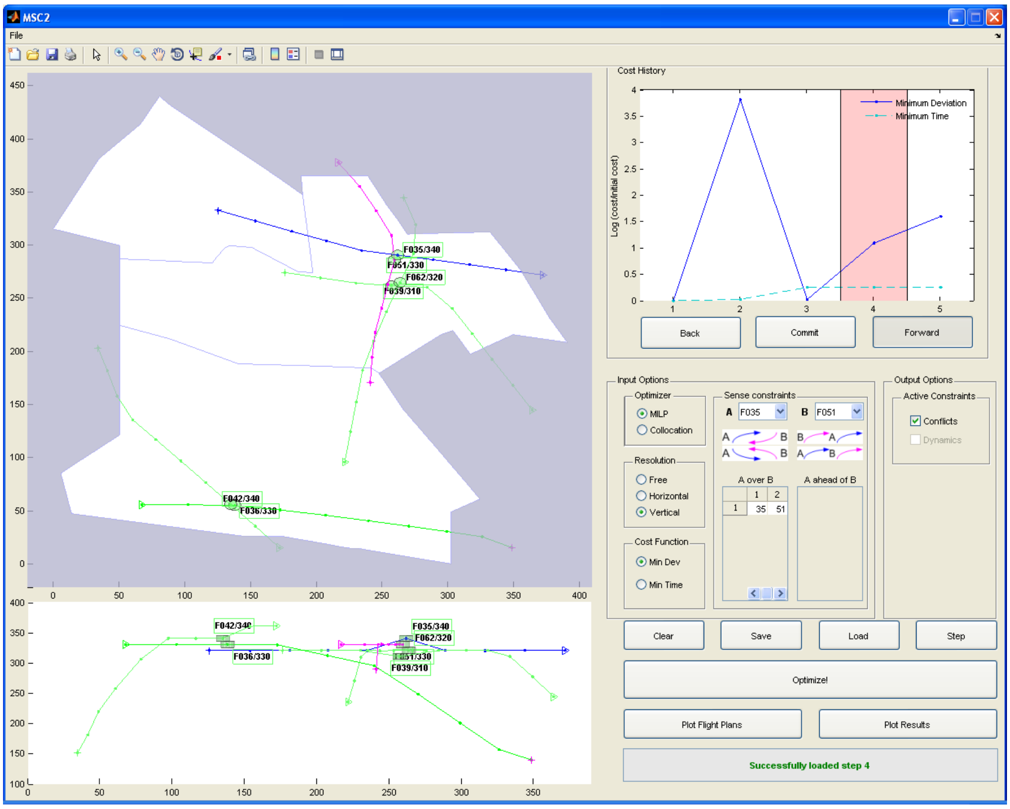


Fig. 10. Example of cost history to explore conflict resolution in previous example