Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Real Time Trajectory Planning for UAVs Using MILP

Waseem Ahmed Kamal, Da-Wei Gu, Ian Postlethwaite
Control and Instrumentation Group, Department of
Engineering, University of Leicester, Leicester, UK
Email: {wak5, dag, ixp }@le.ac.uk

Abstract— **A probabilistic finite receding horizon approach for trajectory planning of autonomous air vehicles is presented. The approach is based on mixed integer linear programming (MILP) techniques. The risk areas are modelled by dynamic boundaries to direct the vehicle towards the target. Various constraints are formulated to avoid radar zones and collisions, etc. These constraints are extended to be both hard and soft so as to alleviate the infeasibility problem usually encountered. The finite receding horizon approach is numerically stable and can be applied to centralized trajectory planning of a fleet of UAVs in real time. The MILP problem is solved using commercially available software AMPL/CPLEX. Finally the approach is applied to different scenarios with upper and lower bounds on the speed and acceleration of each UAV.**

I. INTRODUCTION

The mixed integer linear programming (MILP) approach is for optimisation problems which have integer variables in the cost function and/or the constraints. A commercially available software package AMPL/CPLEX [1], [2] that uses the branch and bound algorithm is used to solve these problems. In this paper, a receding horizon strategy within the MILP framework is proposed which is based on the work of [3], [4], [5]. Two solution strategies were proposed in [3]: a receding horizon strategy and a fixed arrival time approach. It was shown that receding horizon strategies, while computationally more attractive than strategies aiming at computing complete trajectories a priori can lead the system to unsafe conditions where the MILP is no longer feasible. The approach proposed in [5] decomposes this large problem into assignment and trajectory problems. This allows the control architecture to solve an allocation problem first to determine a sequence of waypoints for each vehicle to visit and then concentrate on designing paths to visit these pre-assigned waypoints. The approach which will be presented here models the risk area with soft dynamical boundaries to alleviate the infeasibility problem usually encountered. The paper is organised as follows. Section II is concerned with the problem formulation, including UAV dynamics, risk modelling and vehicle movement constraints. Section III introduces the receding horizon computation and the use of MILP. Some example scenarios are used to demonstrate the algorithm in Section IV, with computational results and discussions. Conclusions are given in Section V.

II. PROBLEM FORMULATION

A. Risk Modelling

For each defence unit (radar and SAM) opposing a UAV, there is a hit probability. Within a given range, this probability depends on the position of the UAV. This can be calculated by a function of height (h) and distance (d) and is given in [6] by

$$p_k(h,d) = (1 - \mathtt{SS}(\mathtt{d}, \mathtt{R}_{\mathtt{s,m,l}}, \mathtt{s}_{\mathtt{k}_1}))\cdot \\ \mathtt{SS}(\mathtt{d}, \mathtt{0.1.R}_{\mathtt{s,m,l}}, \mathtt{s}_{\mathtt{k}_2}) \cdot \mathtt{SS}(\arcsin(\mathtt{h}/\mathtt{d}), \gamma, \mathtt{s}_{\mathtt{k}_3}) \tag{1}$$

where

$$\mathtt{SS}(\mathtt{x}, \mathtt{x}_0, \mathtt{s}) = \frac{1}{2}\left(1 + \frac{\mathtt{x} - \mathtt{x}_0}{\sqrt{\mathtt{s}^2 + (\mathtt{x} - \mathtt{x}_0)^2}}\right)$$

and γ is the lower coverage angle of the radar and s_{k_i} the softness of the step function *SS*. $R_{s,m,l}$ is the range of the missile which may be short, medium or long. The hit probability for a long range SAM ($R = 65$ km) is shown in Figure 1 for $s_{k_1} = 12$, $s_{k_2} = 2.4$, $s_{k_3} = 0.1$, where the numbers on the different contours indicate the hit probability. If a UAV is within the reach of M SAM sites, the hit probability is increased by possible cooperation such as alternating radar transmissions or the choice of launch site. This effect is modelled by evaluating the hit probability of all covering SAM sites $p_k^j(h,d)$ and using the relation

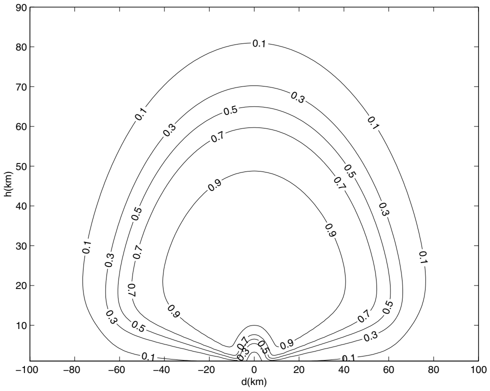

Fig. 1. Hit probability of long range SAM

$$p(x,y,z) = 1 - \prod_{j=1}^{M} (1 - p_k^j(h,d))$$

0-7803-9568-9/05/$20.00 ©2005 IEEE

Auxiliary cost functions and other tactics may be used to avoid radar sites but the risk at point will be calculated using the above formula.

B. Model of the Aircraft

The UAV dynamics is expressed as a simple point mass with position and velocity $[x, y, v_x, v_y]^T$ as state variables and acceleration $[u_x, u_y]^T$ as control inputs. The zero-order hold equivalent discrete time system is

$$\begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}_k + \begin{bmatrix} (\Delta t)^2/2 & 0 \\ 0 & (\Delta t)^2/2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}_k$$

or

$$\mathbf{s}_{k+1} = A\mathbf{s}_k + B\mathbf{u}_k \tag{2}$$

where k is a time step and Δt is the time interval between two steps. The control input $[u_x, u_y]_k^T$ stays constant over each time interval Δt under the zero-order hold assumption.

C. Modelling of the Risk Area with Dynamical boundaries

In [7] we modelled the risk area with soft rectangular boundaries parallel to the coordinate axis and the softness parameters were included into the cost function with fixed weight. The reason for using soft boundaries is described in section III. From simulation, it has been observed that sometimes near the imaginary boundary of the risk area a UAV will loiter for a long time. This is because of the bouncing of a UAV off the boundary again and again without finding a path to direct it towards the target due to the limited horizon and also due to the fixed weight on the softness parameters in the auxiliary cost function described in section II-F. This problem can be fixed by modelling the risk area with soft dynamic rectangular boundaries. At each simulation time step, two of the boundaries for each radar are taken parallel to the line of sight of the current point to the target while the other two boundaries are taken perpendicular to the line of sight as shown in Figure 2. Also the weights on the softness parameters in the auxiliary cost function are chosen dynamically according to the distance of the current point to the target so that when a UAV is near to the boundary of the detected SAM unit, it will be directed towards the target by sliding along the walls of the boundary due to these different weights. The range of the radar warning receivers (RWR) mounted onboard a UAV is assumed slightly higher than the range of the of the radar as defined in the design challenge of the GARTEUR Flight Mechanics Action Group 14 [6]. This assumption is very crude and does not reflect the real situation. So in this paper it is assumed that a SAM site is considered to be known in position and range when its range comes within a distance of $v_{max} * T$, where T is the time horizon and v_{max} is the maximum velocity of the UAV. Consider a radar situated at a point $P_r(x_r, y_r)$ having nominal range R_r enclosed in a rectangular boundary as shown in Figure 2. The position of the p^{th} UAV at the current simulation time step is $P_p^c(x_p^c, y_p^c)$ and $P_p^f(x_p^f, y_p^f)$ is the target point for this vehicle. Therefore the equations of the boundary lines, shown bold in Figure 2 for a radar situated at $P_r^{rad}(x_r^{rad}, y_r^{rad})$ which are parallel to the line of sight are

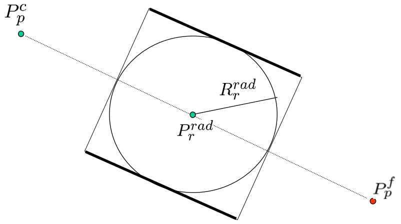

Fig. 2. Dynamical boundaries for risk area

$$(y_p^f - y_p^c)(x_p - x_r^{rad}) - (x_p^f - x_p^c)(y_p - y_r^{rad}) = \pm R_r^{rad} d_p^c \tag{3}$$

Accordingly, the equation of the boundary lines perpendicular to the line of sight are

$$(x_p^f - x_p^c)(x_p - x_r^{rad}) + (y_p^f - y_p^c)(y_p - y_r^{rad}) = \pm R_r^{rad} d_p^c \tag{4}$$

where $d_p^c = \sqrt{(x_p^c - x_p^f)^2 + (y_p^c - y_p^f)^2}$ is the distance of the current point of the UAV from the target. In the situation of overlapping obstacles, the receding horizon algorithm may not find a feasible solution because of such formulated hard constraints. For this consideration, the hard constraints can be transformed to soft constraints by the introduction of slack variables. At each time step i the position (x_i, y_i) of the vehicle must lie in the area outside of the risk area. If their are N_V vehicles and N_R radar sites, then the general mixed integer linear form of the radar avoidance constraints [7] can be written as: $\forall p \in [1, \ldots, N_V], \forall r \in [1, \ldots, N_R], \forall i \in [1, \ldots, N]$

$$\begin{aligned}
&(y_p^f - y_p^c)(x_{pi} - x_r^{rad}) - (x_p^f - x_p^c)(y_{pi} - y_r^{rad}) \\
&\leq -d_p^c R_r^{rad} m_{pci1}^{rad} + \Omega^{rad} b_{pgi1}^{rad} \\
&(y_p^f - y_p^c)(x_{pi} - x_r^{rad}) - (x_p^f - x_p^c)(y_{pi} - y_r^{rad}) \\
&\geq d_p^c R_r^{rad} m_{pci1}^{rad} - \Omega^{rad} b_{pgi2}^{rad} \\
&(x_p^f - x_p^c)(x_{pi} - x_r^{rad}) + (y_p^f - y_p^c)(y_{pi} - y_r^{rad}) \\
&\leq -d_p^c R_r^{rad} m_{pci2}^{rad} + \Omega^{rad} b_{pgi3}^{rad} \\
&(x_p^f - x_p^c)(x_{pi} - x_r^{rad}) + (y_p^f - y_p^c)(y_{pi} - y_r^{rad}) \\
&\geq d_p^c R_r^{rad} m_{pci2}^{rad} - \Omega^{rad} b_{pgi4}^{rad} \\
&\sum_{k=1}^{4} b_{pcik}^{rad} \leq 3 \\
&0 \leq m_{pci1}^{rad},\ m_{pci2}^{rad} \leq 1
\end{aligned} \tag{5}$$

where m_{pci1}^{rad}, m_{pci2}^{rad} are very small decision variables between 0 and 1. The idea is to reduce these variables to zero by incorporating them into the auxiliary objective (cost) function. The problem formulation returns to the original setting when these $m's$ are zero. If it is not possible to reduce them to zero, i.e. the original hard constraints cannot be satisfied, the algorithm will have the flexibility to generate solutions which violate these constraints as little as possible.

D. Collision Avoidance Constraints

In the case of a fleet of UAVs, it is necessary to consider collision avoidance in path planning. This consideration can also be formulated using mixed integer linear constraints [3]. At each time step, every pair of vehicles p and q must be a minimum distance apart from each other in the x and/or y directions. At the i^{th} time step, let (x_{pi}, y_{pi}) and (x_{qi}, y_{qi}) be the positions of the vehicles p and q, respectively, and d_x^{col} and d_y^{col} the safety distances in the x and y directions, then the collision avoidance constraints can be set as in [3] by $\forall i \in [1,\ldots,N], \forall p \in [1,\ldots,N_V], \forall q \in [p+1,\ldots,N_V]$

$$\begin{aligned} x_{pi} - x_{qi} &\geq d_x^{col} - \Omega^{col} b_{pqi1}^{col} \\ x_{qi} - x_{pi} &\geq d_x^{col} - \Omega^{col} b_{pqi42}^{col} \\ y_{pi} - y_{qi} &\geq d_y^{col} - \Omega^{col} b_{pqi3}^{col} \\ y_{qi} - y_{pi} &\geq d_y^{col} - \Omega^{col} b_{pqi4}^{col} \\ b_{pqik}^{col} &= 0 \texttt{ or } 1 \end{aligned} \tag{6}$$

Due to the probabilistic nature of the problem and also to increase the solvability of the optimisation problem, the constraints shown above can be written as soft constraints. $\forall i \in [1,\ldots,N], \forall p \in [1,\ldots,N_V], \forall q \in [p+1,\ldots,N_V]$

$$\begin{aligned} & x_{pi} - x_{qi} \geq d_x^{col}(1 - m_{pqi1}^{col}) - \Omega^{col} b_{pqi1}^{col} \\ & x_{qi} - x_{pi} \geq d_x^{col}(1 - m_{pqi1}^{col}) - \Omega^{col} b_{pqi2}^{col} \\ & y_{pi} - y_{qi} \geq d_y^{col}(1 - m_{pqi2}^{col}) - \Omega^{col} b_{pqi3}^{col} \\ & y_{qi} - y_{pi} \geq d_y^{col}(1 - m_{pqi2}^{col}) - \Omega^{col} b_{pqi4}^{col} \\ & \sum_{k=1}^{4} b_{pqik}^{col} \leq 3 \\ & b_{pqik}^{col} = 0 \texttt{ or } 1 \\ & 0 \leq m_{pqi1}^{col},\ m_{pqi2}^{col} \leq 1 \end{aligned} \tag{7}$$

But due to the importance of these constraints, a higher weight will be assigned to the these small parameters m's in the auxiliary cost function.

E. Speed and Acceleration Constraints

The maximum speed v_{max} is enforced by an approximation to a circular region in the velocity plane [4], [5]. For each vehicle, the velocity vector is projected to different directions to obtain $\forall m \in [1,\ldots,N_C^{v_{max}}],\ \forall i \in [1,\ldots,N]$

$$\dot{x}_i \cos(\frac{2\pi m}{N_C^{v_{max}}}) + \dot{y}_i \sin(\frac{2\pi m}{N_C^{v_{max}}}) \leq v_{max} \tag{8}$$

The above constraints require that the velocity vector be inside a regular polygon with N_C sides circumscribed about a circle of radius v_{max}. A constraint on the minimum speed can be expressed in a similar way. However, it is different from the maximum speed constraint in that at least one of the constraints must be active instead of all of them, $\exists m \in [1,\ldots,N_C^v],\ \forall i \in [1,\ldots,N]$

$$\dot{x}_i \cos(\frac{2\pi m}{N_C^v}) + \dot{y}_i \sin(\frac{2\pi m}{N_C^v}) \geq v_{min} \tag{9}$$

where N_C is the order of the discretization of the circle. Equation 9 is a non-convex constraint and can be written as a mixed integer linear constraint $\forall m \in [1,\ldots,N_C^v],\ \forall i \in [1,\ldots,N]$

$$\begin{aligned} \dot{x}_i \cos(\frac{2\pi m}{N_C^v}) + \dot{y}_i \sin(\frac{2\pi m}{N_C^v}) \quad &\geq \quad v_{min} - \Omega^v(1 - b_{im}^v) \\ \sum_{m=1}^{N_C^v} b_{im}^v \quad &\geq \quad 1 \end{aligned} \tag{10}$$

Similarly, the constraint for the upper bound on acceleration can be written as: $\forall m \in [1,\ldots,N_C^a],\ \forall i \in [1,\ldots,N]$

$$\ddot{x}_i \cos(\frac{2\pi m}{N_C^a}) + \ddot{y}_i \sin(\frac{2\pi m}{N_C^a}) \leq u_{max} \tag{11}$$

F. Auxiliary Cost Function

The actual risk function is given in section II-A but this risk function cannot be used directly in the MILP formulation due to highly complicated nonlinear characteristics. So an auxiliary objective function is defined. This objective function can be taken as the sum of two costs: a quadratic cost function and a cost to minimise the violation of constraints. We can minimise a quadratic function whose variables must satisfy the state space equation of the dynamic system (II-B) as follows

$$\min_{\mathbf{s},\mathbf{u}} J = \min_{\mathbf{s},\mathbf{u}} \int_0^\infty (\mathbf{s}^T Q \mathbf{s} + \mathbf{u}^T R \mathbf{u}) dt \tag{12}$$

subject to

$$\dot{\mathbf{s}} = A_c \mathbf{s} + B_c \mathbf{u} \tag{13}$$

where $\mathbf{s} \in \Re^n$ is the state vector and $\mathbf{u} \in \Re^{n_u}$ is the control. The system is assumed to start from some initial state $\mathbf{s}_0$ and end at a final (target) state $\mathbf{s}_f$. A linear discretised form of the quadratic cost is given by:

$$\begin{aligned} \min_{\mathbf{w}_{pi},\mathbf{v}_{pi}} \sum_{p=1}^{N_V} J_p = \min_{\mathbf{w}_{pi},\mathbf{v}_{pi}} \quad \Sigma_{p=1}^{N_V} \quad & (\sum_{i=1}^{N-1} \mathbf{q}_p^T \mathbf{w}_{pi} + \sum_{i=0}^{N-1} \mathbf{r}_p^T \mathbf{v}_{pi} \\ + \quad & \mathbf{p}_{\mathrm{p}}^{\mathrm{T}} \mathbf{w}_{\mathrm{pN}}) \end{aligned} \tag{14}$$

subject to

$$\begin{aligned} s_{pij} - s_{pfj} \quad &\leq \quad w_{pij} \\ -s_{pij} + s_{pfj} \quad &\leq \quad w_{pij} \\ u_{pik} \quad &\leq \quad v_{pik} \\ -u_{pik} \quad &\leq \quad v_{pik} \\ \mathbf{s}_{p,i+1} \quad &= \quad A_p \mathbf{s}_{p,i} + B_p \mathbf{u}_{pi} \end{aligned} \tag{15}$$

The weighting matrices Q and R of the quadratic formulation have been replaced by nonnegative weighting vectors $\mathbf{q}$ and $\mathbf{r}$, and s_{pfj} is the given final j^{th} state.

As discussed above, one way to increase solvability of the problem while keeping the violation of constraints to a minimum is to soften the constraints by including small slack variables ($m's$) in the cost function (14) as follows:

$$\min_{m^{rad}_{pcif}, m^{col}_{pqig}} \sum_{p=1}^{N_V} \tilde{J}_p = \min_{m^{rad}_{pcif}, m^{col}_{pqig}} \sum_{p=1}^{N_V} (w^{col} \sum_{q=p+1}^{N_V} \sum_{i=1}^{N} \sum_{g=1}^{2} m^{col}_{pqig} - \sum_{h=1}^{2} w^{rad}_h \sum_{c=1}^{N_R} \sum_{i=1}^{N} \sum_{f=1}^{4} m^{rad}_{pcif}) \tag{16}$$

So the overall objective function for the p^{th} vehicle is $J_p + \tilde{J}_p$. In equation (16), the weights ($w^{rad}_1 = \alpha \times d^c_p$, $w^{rad}_2 = \beta \times d^c_p$) are selected dynamically according to the distance of the current position to the target and the constant α is given smaller value than β. To avoid a collision, the weight w^{col} is given a relatively big value. The constraint list in (16) should be expanded to include other constraints considered earlier as the case in question to form the optimisation problem.

III. Receding Horizon Approach and MILP

A. Receding Horizon Control

A major difficulty in using MILP is the computational demand it requires. The computations increase dramatically with the number of time intervals (steps). On the other hand, big time steps could lead to inaccurate or non-implementable "solutions" . In order to make the algorithm workable in real time or near real time, a receding horizon approach has to be considered. In this approach, the path is computed online by solving the MILP over a limited horizon (in terms of time intervals) at each time step. The procedure is composed of a sequence of locally optimal segments. At each time step, the MILP is solved for T future time intervals, where the length T of the planning horizon is chosen as a function of the available computing resources as well as the individual problem. Solving this MILP provides the input commands for the T future time steps. The solution is of course only locally optimal. Only a subset of these T input commands is actually implemented. The process is then repeated and a new set of commands is generated for the next time window. Usually the applied subset is restricted to the first control input, such that a new set of input commands is calculated at each time step. Since the controller is designed at every sampling instant, disturbances can easily be dealt with. The concept is equally applicable to single-input, single-output (SISO) and multi-input, multi output (MIMO) systems, both linear and nonlinear.

B. Possible Infeasibility with Receding Horizon

When using receding horizon approaches, the non-existence of feasible solutions may occur during the MILP procedure, though in theory there are solutions to the whole problem. This is because the look ahead horizon is limited. The vehicle can be led to a critical state for which MILP has no solution in the next iteration. In other words, a feasible solution for T further time steps at current time step i does not guarantee a feasible MILP at the time step $i+1$. This can be further explained by the situation in which in the last time step of the planning horizon, the vehicle is moving at maximum speed, while its position is close to an obstacle that has not yet been spotted. The position of the vehicle satisfies the anti-collision constraints and so corresponds to a feasible solution of the MILP. At the next time step, the obstacle is identified and the vehicle needs to brake or turn quickly thereby exceeding the constraints on acceleration or on the available manoeuvre space; a solution will not therefore be found. Increasing the time horizon will ease this kind of situation, but will also raise the computational demand.

C. Safe Feasible Mechanism

In the radar/SAM exposure minimisation problem, there are no physical obstacles. Rather we have radars of various detection ranges. We may have, say, three types of radars: long range SAM units (65 km), medium range SAM units (25 km), and short range SAM units (7 km). We can approximately model these circular regions with squares of the same lengths as the radii of these circles. So in order to make the problem feasible some minimum violation of these constraints can be allowed. These radar ranges can overlap with one another. So if the UAV path is totally blocked by these overlapping radars or if the receding horizon approach with hard constraints and cost is used, then MILP leads to an infeasible solution. But by using the soft constraints (5), (7) and including auxiliary variables in the cost (16), we can always obtain a feasible solution. Violations are kept to a minimum by use of the small variables ($m's$) in the cost. If further reduction of violations is required, we may model the threats as squares of flexible size, slightly greater than the actual fitted square. The increment can be taken as 10% of the actual square. In this way, a vehicle can enter the radar zone but has to follow the safest possible path by optimising this flexibility.

IV. Scenarios and Simulation Results

The simulation for one vehicle case was performed for 100 randomly created scenarios consisting of short range (7 km) and medium range (25 km) SAM units distributed over a 200 $\mathtt{km} \times 200$ $\mathtt{km}$. The random parameters are the number, locations and strengths of the SAM units. The number of SAM units was limited between 5 and 10. Since the focus was to make the technique useful for real time computation so an upper limit of 1.5 seconds was set for the computation or optimisation process using CPLEX. The justification for this is that large and difficult problems may take hours or days to prove for optimality and the optimal solution might have been obtained at very early stages of calculation. The parameters used in all simulations are summarised in Table 1. The measured quantities are success rate, peak computation time (maximum time to compute a way point), peak risk, average risk and total flight length. The results are plotted in Figures 3, 4, 5, 6 and 7. It is clear from Figure 6 that the dynamic modelling of the risk zones helps a lot in reducing the flight times and this is achieved by directing the vehicle towards the target by moving along the rectangular boundaries that change at each simulation step. The horizon T is chosen as 5 seconds, i.e, the trajectory is evaluated

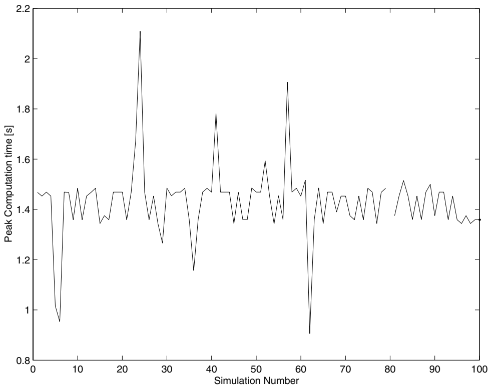

Fig. 3. Peak computation time for 100 randomly generated data sets

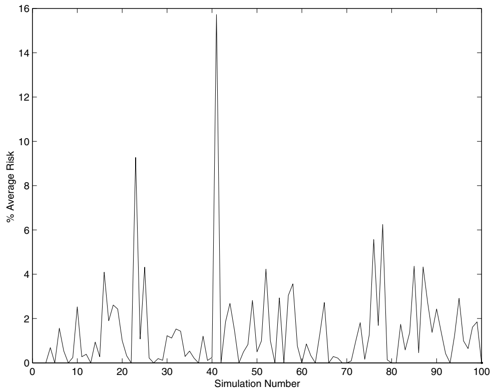

Fig. 5. Average risk for 100 randomly generated data sets

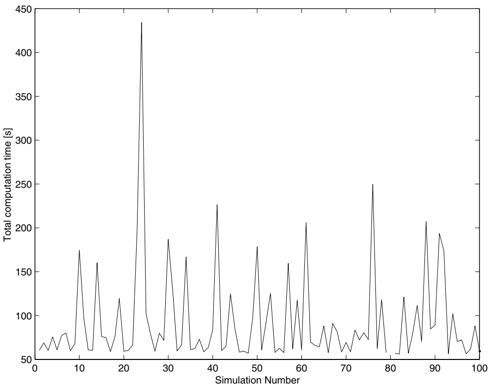

Fig. 4. Total computation time for 100 randomly generated data sets

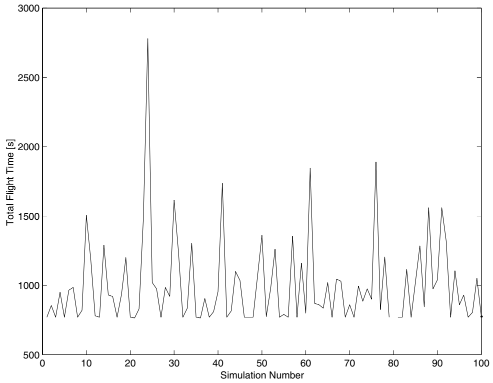

Fig. 6. Total flight time for 100 randomly generated data set

at five time steps of each 1 second duration. The success rate was 99% i.e, the solution was obtained for 99 random scenarios. For the 80^{th} scenario, no solution was obtained within the set time limit. Figure 3 shows a little violation of the set time limit for a few scenarios. The cause for these violations is not clear. This extra delay might be due to the internal setting of the commercially available solver CPLEX while communicating with the AMPL/CPLEX from within the Matlab environment. The risk was evaluated with the actual probabilistic model described in section II-A. As shown in Figure 5, the average risk for each trajectory while travelling to the destination for most of the scenarios is reasonable except for the 41^{st} and 23^{rd} scenarios. The average risk for the 41^{st} scenario is 16%. The highest values of the total distance travelled, total computation time, total flight time and peak risk occurs at the 24^{th} scenario. This is due to the very high risk experienced by the UAV as it moves towards the target and because of this it starts loitering in order to find a safe region and thus making the flight path relatively longer than other scenarios. A multi-vehicle scenario is shown in Figure 8. This example finds centralised trajectories for 3 UAVs which start from different positions but fly to the same destination. The operational region is $180\ \mathtt{km} \times 200\ \mathtt{km}$ and has 10 defence units (radar and SAM) shown as circles in the Figure 8. Five units are of medium range (25 km) centred at coordinates $(100,100)$, $(125,65)$, $(125,135)$, $(50,155)$, $(50,45)$, and the rest are of short range (7 km) centred at $(42,102)$, $(167,182)$, $(167,127)$, $(167,37)$, $(167,77)$. The initial positions of UAV1, UAV2, UAV3 are $(10,10)$, $(10,120)$, $(10,180)$, respectively. All three UAVs start at the same time and move towards a common goal at $(170,100)$. The UAVs fly at an initial speed of 200 m/s and with an initial heading angle of 40° with regard to the horizontal. It is required that the final speed of each vehicle when arriving at the destination is 100 m/s and the heading is along the x-axis. The maximum flight speed is 300 m/s (see Table 1). In this exercise the horizon T is chosen as 25 seconds, i.e, the trajectory is evaluated for five time steps each of 5 seconds duration but only the first control input is implemented. To use the formulation described earlier,

TABLE 1. PARAMETERS USED IN THE SIMULATION

Parameter	Value	Parameter	Value
$\mathbf{q}_p^T$	$[1,1,1,1]$	$\mathbf{r}_p^T$	$[1,1]$
$\mathbf{p}_p^T$	$[1,1,1,1]$	w_1^{rad}	$3 \times d_p^c$
w^{col}	10^{10}	d_x^{col}	1000
d_y^{col}	1000	Ω^{col}	8×10^5
Ω^{rad}	8×10^{11}	Ω^v	9000
Ω^u	50	N_C^v	30
N_C^v	30	N_C^u	30
u_{max}	10	w_2^{rad}	$6 \times d_p^c$
v_{min}	100	v_{max}	300

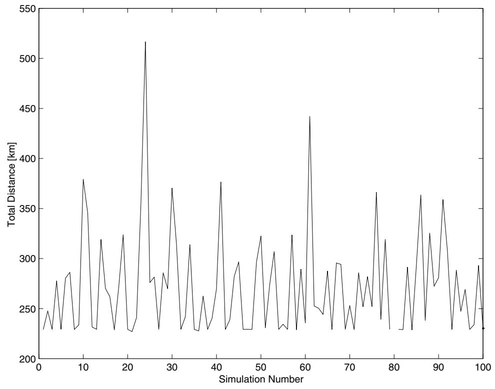

Fig. 7. Total flight length for 100 randomly generated data sets

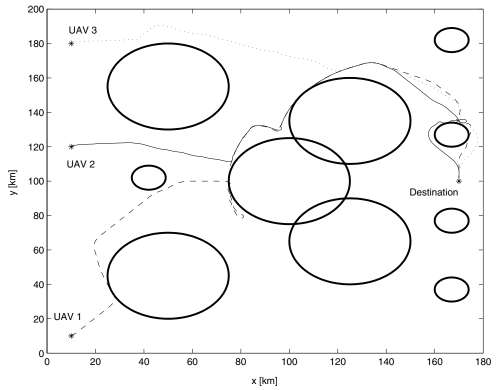

Fig. 8. A multi-vehicle scenario and trajectories for three UAVs

the defence units are modelled as squares containing the circular threat regions. Adoption of "soft"constraints in the finite horizon optimisation scheme alleviates the possibility of infeasibility outcomes. The vehicles may enter threat zones but will try to leave them as soon as possible to keep constraints violations at a minimum. It has been observed that maximum violations will occur when the vehicle enters a threat zone at maximum speed and is perpendicular to the square boundary at a tangent point with the circle. That makes it possible that the vehicle will actually enter the dangerous zone before it can make a turn. Hence, in order to reduce this exposure, an additional safety measure can be taken by expanding the squares by 10% as is done in this exercise. The computation is carried out using a PC machine with a CPU of 2.66 GHz and a RAM of 1.048 Gb. The computation time for the full simulation is 1970 seconds; that is, the time when the last vehicle (UAV1) reaches the destination. The simulation times for UAV2 and UAV3 are 1390 and 860 seconds, respectively. The collision avoidance can be seen in Figure 9 which shows the distances between the each two vehicles at different times.

V. Conclusion

In this work, we have shown that flight trajectory planning for multiple UAVs can be solved by a linearly constrained optimisation formulation with real and integer variables. To design a whole trajectory with a planning horizon fixed at the goal is very difficult to perform in real time because the computational effort required grows rapidly with problem size. It was shown that this limitation can be avoided and trajectories in real time can be obtained by using a finite receding horizon approach, in which MILP is used to form a shorter plan at each simulation step that extends towards the goal but does not necessarily reach it. Robustness of receding horizon control is guaranteed by modelling the constraints as soft while the flight time can be reduced by dynamical boundaries of the radar zones. The efficiency of the techniques depends upon proper modelling of the mixed linear constraints and also on the time horizon. Optimality can be increased by increasing the time window but by doing this the computational load will increase. There should be adequate compromise between optimality and computational load. Future work will be focused on extending the approach to three dimensions with an emphasis on calculating trajectories in real time. A comparison between the efficiency of the MILP approch and nonlinear optimisation will also be carried out.

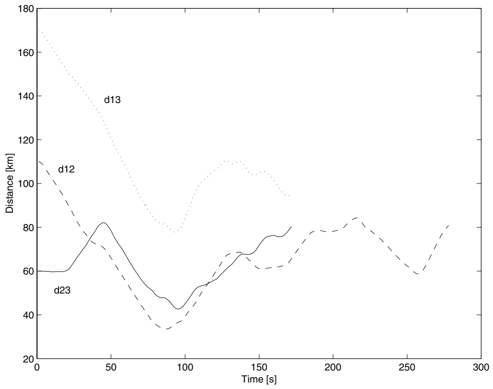

Fig. 9. Distances between the UAVs during flight

VI. Acknowledgments

The research work is supported by the TROSS scholarship scheme of the Government of Pakistan, the UK Engineering and Physical Sciences Research Council and BAE Systems.

References

[1] *ILOG CPLEX User's guide*, ILOG, 1999.

[2] R. Fourer, D. M. Gay, and B. W. Kernighar, *AMPL, A modeling language for mathematical programming*. The Scientific Press, 1993.

[3] T. Schouwenaars, B. DeMoor, E. Feron, and J. P. How, "Mixed integer programming for multi-vehicle path planning," in *Proc. of the ECC Conf.*, Porto, Portugal, September 2001, pp. 2603–2608.

[4] A. Richards and J. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," in *Proceedings of the American Control Conference*, May 2002.

[5] A. Richards, J. Bellingham, M. Tillerson, and J. P. How, "Coordination and control of multiple UAVs," in *Proc. of the AIAA GN&C Conf.*, Aug 2002.

[6] GARTEUR Action Group AG14, "Autonomy in UAVs: A design challenge," *http://www.nlr.nl/projects/garteur_wan/inde x2.html*, 2003.

[7] W. A. Kamal, D. W. Gu, and I. Postlethwaite, "MILP and its application in flight path planning," in *Proceedings of the 16th IFAC World Congress*, Prague, July 2005.