# Trajectory Optimization in Convex Underapproximations of Safe Regions

Jerry Ding and Claire J. Tomlin

*Abstract*— **This paper discusses a computationally efficient method for optimizing aircraft trajectories in a two-aircraft conflict scenario, under a noncooperative setting. It is assumed that the future trajectory of the uncontrolled aircraft is unknown, but that deterministic input bounds are given. Unsafe reachable sets are computed in a game theoretic framework to account for worst case behaviors. Overapproximations of the reachable sets are used as constraints in a convex trajectory optimization program at each time step. We prove the safety property of the optimization program, and address the errors introduced by model linearization through a robustness analysis. Simulation results demonstrate that the algorithm generates collision free conflict resolution trajectories even as the adversary aircraft inputs are drawn randomly from within input bounds.**

## I. Introduction

In the current air transportation system, conservative restrictions are imposed on the operational procedure and routing of aircraft to ensure safety under uncertainties such as pilot behavior, weather, and changes in air traffic density [1]. However, this often results in trajectories that are indirect, not wind optimal, and inefficient in the use of airspace. As the demand for air travel increases, there is a growing need for more efficient routing of aircraft. This could lead to a situation in which aircraft have the freedom to choose optimal trajectories for their intended flight plan, under the constraint of safe separation from other aircraft. Under these new operating conditions, automated conflict detection and resolution algorithms could be used to alleviate the workload of human operators while ensuring the safe operation of such a system.

In this work, we consider the problem of synthesizing pairwise horizontal conflict resolution maneuvers, which are commonly employed for en route air traffic away from busy airports. In the spirit of noncooperative collision avoidance, we will assume that the future trajectory of one of the aircraft (termed the pursuer in the game theoretic framework) is unknown, and that we do not have access to its control inputs. However, its inputs are assumed to lie within certain known bounds. The only information available to the controlled aircraft (termed the evader) is the position of the pursuer at discrete time instants. The goal is then to compute an efficient conflict-free resolution trajectory over a finite time horizon while optimizing for progress towards a given goal point. We ensure the safety condition through the computation of Hamilton-Jacobi reachable sets [2], and the optimal control inputs for the evader are computed by solving a convex optimization problem at each discretized time instant.

This work is supported by the "MURI - Frameworks and Tools for High Confidence Design of Adaptive, Distributed Embedded Control Systems" project administered by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-06-1-0312.

J. Ding and C.J. Tomlin are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720, USA {jding,tomlin}@EECS.Berkeley.Edu

## II. Related Works

For an introduction to the diverse literature in aircraft conflict detection and resolution, one may refer to [3]. Significant effort has been devoted to studying the problem of cooperative conflict resolution. In this formulation of the problem, it is assumed that the intent of the aircraft involved in a conflict scenario is known perfectly, either to a centralized controller, or through the exchange of information through data link between aircraft. In centralized schemes, the central planner chooses the control input of all agents in the conflict scenario to satisfy safety constraints and minimize a global optimality criterion [4], [5].

The main drawback with centralized schemes is that it cannot be easily extended to an air traffic management system where part of the control authority is distributed down to individual aircraft. This is addressed through decentralized conflict resolution [6]–[9], where individual agents solve independent trajectory planning problems and broadcast their solutions to neighboring agents through a data link. While not yielding globally optimal trajectories, these algorithms ensure certain global properties such as collision avoidance through appropriate choices of constraints and cost functions.

Although cooperative conflict resolution may yield optimal solutions under ideal operating conditions, the uncertainties inherent in the air transportation system are not sufficiently addressed. This could include uncertainties in aircraft behavior due to wind turbulence or adverse weather patterns and possible dropouts in communication between aircraft. Noncooperative conflict resolution [10]–[12] has sought to address these issues by using game theoretic techniques to prove safety properties of flight maneuvers under bounded uncertainties. The disadvantages of this approach include conservative solutions as well as difficulties in efficiently synthesizing resolution maneuvers online.

One may note that the problem of conflict resolution is akin to trajectory planning in the presence of a moving obstacle. This problem is well studied in the robotics literature when the trajectory of the moving obstacle is known [13]–[17]. In that case, the problem can be reduced to a static obstacle avoidance problem by projecting the trajectory of the moving obstacle in space-time. A feasible trajectory can then be computed through visibility graph methods [13], [14] or cell decomposition of the free space [16], [17]. Computationally efficient methods such as probabilistic roadmaps [18]–[20] and artificial potential fields [21]–[23] have been recently developed. These methods focus upon finding a feasible path and are shown experimentally to handle small changes in the environment. However, robust guarantees under uncertainties, especially in the presence of fast moving unknown obstacles, are not available.

The main contribution of our work is in formulating an efficient online trajectory optimization problem in the presence of an adversary with unknown future trajectory, but bounded inputs. As compared with cooperative trajectory planning schemes, we seek to address the uncertainties in the air traffic system by using reachable sets to predict the worst case trajectories of the adversary aircraft. As compared with existing work on verifying safety in noncooperative conflict scenarios, this work proposes a method to efficiently synthesize conflict resolution maneuvers in an online setting. Finally, with respect to global planning schemes in the robotics literature, the proposed algorithm make use of over-approximations of reachable sets to allow for reactive online planning, which may yield less conservative trajectories depending on the behavior of the adversary.

## III. System Model

To develop a more precise formulation of the problem, we first present a point mass kinematics model for two aircraft moving in 2D plane. An example of a conflict scenario is shown in Figure 1. Let the state of aircraft $i$ be denoted as $x_i$, we have from the kinematic equations in 2D plane

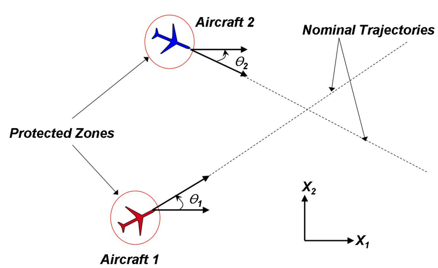


Fig. 1. Two aircraft encounter scenario in horizontal plane.

$$\dot{x}_i = f(x_i, u_{1i}, u_{2i}) = \frac{d}{dt}\begin{bmatrix} x_{1i} \\ x_{2i} \\ \theta_i \end{bmatrix} = \begin{bmatrix} u_{1i}\cos\theta_i \\ u_{1i}\sin\theta_i \\ u_{2i} \end{bmatrix} \tag{1}$$

The state variables in this equation are defined as follows:

- $x_{1i}$ = longitudinal coordinate of aircraft$_i$
- $x_{2i}$ = latitudinal coordinate of aircraft$_i$
- $\theta_i$ = heading of aircraft$_i$

The inputs in this equation are defined as follows:

- $u_{1i}$ = translational speed of aircraft$_i$
- $u_{2i}$ = turning rate of aircraft$_i$

For the optimization problem, we will only apply constant control input on each discrete time interval. As such, we discretize the above continuous time model by integrating the dynamics of a single aircraft over $[t_k, t_{k+1}]$ assuming constant control inputs [8],

$$x_{1i}(t) = \begin{cases} x_1(t_k) + \frac{u_1}{u_2}\left(\sin(\theta(t_k) + u_2(t-t_k))\right. \\ \left.-\sin\theta(t_k)\right),\ u_2 \neq 0 \\ x_1(t_k) + u_1\cos(\theta(t_k))(t-t_k),\ u_2 = 0 \end{cases} \tag{2}$$

$$x_{2i}(t) = \begin{cases} x_2(t_k) - \frac{u_1}{u_2}\left(\cos(\theta(t_k) + u_2(t-t_k))\right. \\ \left.-\cos\theta(t_k)\right),\ u_2 \neq 0 \\ x_2(t_k) + u_1\sin(\theta(t_k))(t-t_k),\ u_2 = 0 \end{cases} \tag{3}$$

Assuming that the time interval $[t_k, t_{k+1}]$ is chosen to be short enough, we can apply small angle approximation and model linearization to arrive at the following approximate linear system dynamics

$$p_1(t,u) \approx p_1(t_k) + B(t)u,\ t \in [t_k, t_{k+1}] \tag{4}$$

where the matrix $B(t)$ is given by

$$B(t) = \begin{bmatrix} \cos\theta(t_k)(t-t_k) & -\sin\theta(t_k)\frac{(t-t_k)^2}{2}\tilde{u}_1 \\ \sin\theta(t_k)(t-t_k) & \cos\theta(t_k)\frac{(t-t_k)^2}{2}\tilde{u}_1 \end{bmatrix} \tag{5}$$

Here $\tilde{u}_1$ is defined as the choice of $u_1$ in the previous time interval $[t_{k-1}, t_k]$.

## IV. Problem Formulation

Now we present a precise formulation of the constrained trajectory optimization problem. At some initial time $t_0$, a controllable aircraft, which will be referred to as the evader in the game theoretic framework, starts at a location in 2D space $p_1(t_0)$, while an adversary aircraft, which will be referred to as the pursuer, starts at a location $p_2(t_0)$. Let $p_{final}$ be a desired target location that the evader would like to reach. A conflict occurs comes when one aircraft enters a circular protected zone of radius $r_0$ centered on the other aircraft. In current en route airspace, this distance is set to be 5 nmi.

A fixed terminal time $t_f$ is chosen, and the finite time horizon $[t_0, t_f]$ is discretized into $N$ time intervals $[t_k, t_{k+1}], k = 0, 1, ..., N-1$, where $t_{k+1} - t_k = T$ for some fixed sampling period $T$, and $t_N = t_f$. At the start of each time interval, the evader receives a reading of the pursuer position $p_2(t_k)$. It then chooses a constant control input $u_{[t_k, t_{k+1}]}$ within the bounds $[\underline{u}, \overline{u}]$ that ensures a collision free trajectory and optimizes for progress towards the desired target location. At the same time, the pursuer applies a disturbance input $d_{[t_k, t_{k+1}]}$ that is unknown and can be time varying, but is assumed to lie within certain input bounds $[\underline{d}, \overline{d}]$. In this formulation of the problem, we are primarily concerned with generating a collision free trajectory under uncertain pursuer behavior. No guarantees will be made with regards to reachability to the desired target point. This latter problem can be handled with additional machinery from reachable set computation [2]. However, it will be left for a future publication.

We denote an unsafe backward reachable set over a finite time horizon $\tau$ by $G_\tau$. For the particular case under consideration, it is the subset of the relative coordinate space for which there exists some pursuer speed and turn rate input that can force the evader to enter the protected zone of the pursuer aircraft over some finite time horizon, regardless of the choice of evader speed and turn rate input. As seen in [24], this set grows converges to a fixed point under the condition that the evader is more maneuverable than the pursuer, i.e. $\underline{u} \leq \underline{d}$ and $\overline{d} \leq \overline{u}$. Thus, one may define this set to be the infinite horizon unsafe set $G_\infty$. By constraining the trajectory of the evader aircraft to lie outside this set, i.e. $p_1(t) \notin G_\infty$, $\forall t \in [t_0, t_f]$, we ensure that there always exists some feasible evader control input that can safely avoid a collision with the pursuer aircraft, even at the termination of the finite horizon planning algorithm. One may also choose to replace $G_\infty$ by any compact region $S$ centered on the pursuer aircraft. In that case, the algorithm described here can ensure that the evader aircraft remains outside of $S$ over the time horizon of interest.

The trajectory planning algorithm begins by generating offline the set $G_\infty$. The input bounds are chosen, based upon realistic aircraft data [25], as $[\underline{u}_1, \overline{u}_1] = [\underline{d}_1, \overline{d}_1] = [400\ kts, 500\ kts]$, $[\underline{u}_2, \overline{u}_2] = [-2\ deg/s, 2\ deg/s]$, $[\underline{d}_2, \overline{d}_2] = [-1\ deg/s, 1\ deg/s]$. With this, the set $G_\infty$ is shown in Figure 2(a) in relative coordinates. The reachable set is generated using a numerical level set toolbox [26]. Here the vertical direction is the relative heading between the aircraft, while the horizontal plane is the relative 2D position of the aircraft. To eliminate the heading dependence, we project this set onto the horizontal plane and overbound it by a disc of radius $r_0$ centered on the position of the pursuer, call it $D_{r_0}(p_2)$. For the given input bounds, the projected set is shown in Figure 2(b). As can be seen, we can choose $r_0 = 13\ nmi$ to overbound this set.

To compute a constraint set to be used for the optimization program, we need to introduce the notion of a forward reachable set. A forward reachable set $F_\tau$ is defined as the set of states that can be reached from a set of initial conditions $X_0$ within time horizon $\tau$. With this definition in mind, the constraint set for the optimization program is then the forward reachable set computed from the over approximation of the unsafe set $G_\infty$. Specifically, the initial condition is set to be $X_0 = D_{r_0}$, and $F_\tau$ is computed for $\tau \in [t_0, t_f]$ using the dynamics of the pursuer with the assumed disturbance input bounds. It can be shown that projection of the forward reachable set onto the relative x-y space after time $\tau$ is given by a disc of radius $r_0 + \overline{d}_1\tau$, where $\overline{d}_1$ is the maximum translational velocity of the pursuer, and so $F_\tau = D_{r_0+\overline{d}_1\tau}(p_2)$. An example forward reachable set computed from the projected $G_\infty$ set is shown in Figure 3. The problem then becomes one where, in each discrete time interval $[t_k, t_{k+1}]$, we choose a constant input $u_{[t_k, t_{k+1}]}$ within the input bounds, so that the trajectory $p_1(t), t \in [t_k, t_{k+1}]$ lies outside the set $F_{t_f - t_k} = D_{r_0+\overline{d}_1(t_f-t_k)}(p_2(t_k))$, while minimizing the distance to the target location at the end of the time interval $\left\| p_1(t_{k+1}) - p_{final} \right\|_2$.

It is important to note the reactive nature of this algorithm. Although we overapproximate the forward reachable set, it is with respect to the updated position of the pursuer aircraft at each time step. Specifically, we do not preclude

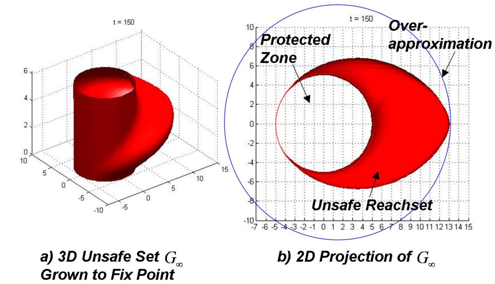


Fig. 2. (a)Backward reachable set for specified input bounds; (b)projection of backward reachable set onto horizontal plane.

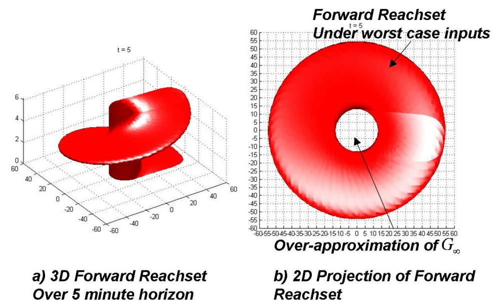


Fig. 3. (a)Forward reachable set computed from unsafe set $G_\infty$ over time horizon of 5 minutes; (b)projection of forward reachable set onto horizontal plane.

the evader aircraft from entering the forward reachable set over the entire time horizon, $F_{t_f - t_0}$, as one would have to do with a global planning algorithm. Consequently, the online planning algorithm proposed here may yield less conservative trajectories as compared to a global planning algorithm, depending on the actions taken by the pursuer aircraft.

## V. Optimization Program

The main result of this section is that an optimization program can be formulated with reachable set constraints for the nonlinear system dynamics described by equations (2) and (3) such that the trajectory $p_1(t), t \in [t_k, t_{k+1}]$ lies outside the set $F_{t_f - t_k}$ in each time step. By properties of the forward reachable set, this guarantees that the evader never enters into a conflict with the pursuer and that the trajectory terminates in a safe condition over the finite optimization interval.

In order to prove this result, we first develop some notations. Denote the heading vector of the evader at time $t$ by $h_t$ and the corresponding heading angle by $\theta_1(t)$. Following the previous notation, denote a disk of radius $r$ by $D_r$. Let $q$ be any point outside $D_r$, then we can calculate tangent lines from $q$ to $D_r$, denote these by $L_1(q, D_r)$ and $L_2(q, D_r)$. We can represent points on $L_1$ and $L_2$ by $L_i(q, D_r) = \left\{x : a_i^T x = b_i\right\}, i = 1, 2$, where $a_i$ and $b_i$ are chosen so that $\forall x \in D_r,\ a_i^T x \geq b_i,\ i = 1, 2$. Denote the half planes containing $D_r$ by $H_1(q, D_r)$ and $H_2(q, D_r)$, where $H_i(q, D_r) = \left\{x : a_i^T x \geq b_i\right\},\ i = 1, 2$, so that $D_r \subset H_1(q, D_r) \cap H_2(q, D_r)$.

Finally, for a given subset $A$ in the state space $X$, we denote its complement by $A^C = X \backslash A$. Now we can formally state our result.

### A. Provably Safe Optimization Program

Assume the model for $p_1(t)$ in terms of the input $u$, given by equations (2) and (3) in section III. Consider an initial condition satisfying $p_1(t_0) + \alpha h_{t_0} \in F^C_{t_f - t_0}, \forall \alpha \geq 0$, and the set $G_\infty$ is known. Note that this condition is equivalent to $p_1(t_0) + \alpha h_{t_0} \in H_1(p(t_0), F_{t_f - t_0})^C \cup H_2(p(t_0), F_{t_f - t_0})^C, \forall \alpha \geq 0$. We will discuss later how the system can be initialized to satisfy this condition.

Now define an optimization algorithm as follows. At time $t_k < t_N$, if $p_1(t_k) + \alpha h_{t_k} \in F^C_{t_f - t_k}, \forall \alpha \geq 0$, then calculate $L_i(p_1(t_k), F_{t_f - t_k}), i = 1, 2$. Note that the condition on $p_1(t_k)$ is equivalent to $p_1(t_k) + \alpha h_{t_k} \in H_1(p(t_k), F_{t_f - t_k})^C \cup H_2(p(t_k), F_{t_f - t_k})^C, \forall \alpha \geq 0$. Assume without loss of generality $p_1(t_k) + \alpha h_{t_k} \in H_1(p(t_k), F_{t_f - t_k})^C, \forall \alpha \geq 0$. Let $a$, $b$, be the corresponding parameters describing $L_1$. Denote the rotation of $h_{t_k}$ by angle $\phi$ as $h_{t_k, \phi}$. Let $\psi_1(t_k) \geq 0$ be the smallest clockwise rotation of $h_{t_k}$ such that $a^T h_{t_k, -\psi_1} \geq b$. Let $\psi_2(t_k) \geq 0$ be the smallest counter-clockwise rotation of $h_{t_k}$ such that $a^T h_{t_k, \psi_2} \geq b$. The geometric interpretation of these parameters is given in Figure 4.

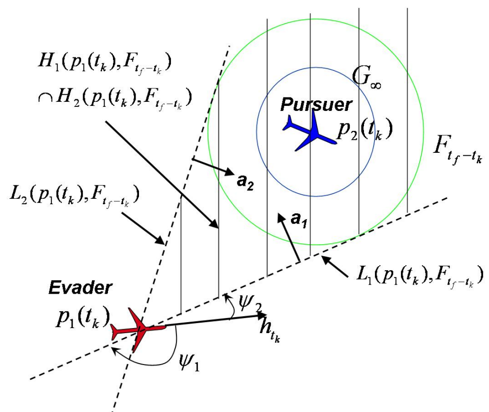


Fig. 4. Geometric interpretation of optimization program parameters.

After calculating these parameters, perform the following optimization program:

*Trajectory Optimization Problem at Time $t_k$*

$$\begin{array}{ll} \text{minimize} & \left\| p_1(t_{k+1}) - p_{final} \right\|_2^2 \\ \text{subject to} & \sup_{t \in [t_k, t_{k+1}]} a^T p_1(t) \leq b \\ & u_2 T \in [-\psi_1(t_k), \psi_2(t_k)] \\ & u_1 \in [\underline{u}_1, \overline{u}_1] \\ & u_2 \in [\underline{u}_2, \overline{u}_2] \end{array}$$

If the optimization is infeasible, then we terminate the optimization program. Otherwise, we use the optimal input $u^*$ calculated by the above program on the interval $[t_k, t_{k+1}]$.

If the assumption on $p_1(t_k)$ is not satisfied, i.e. $\exists \alpha \geq 0$, such that $p_1(t_k) + \alpha h_{t_k} \in F_{t_f - t_k}$, then we terminate the optimization program.

At time $t_N$, we terminate the optimization program, if it has not already.

*Proposition 1:* Assume the model for $p_1(t)$ given by equations (2) and (3). Then under the condition $p_1(t_0) + \alpha h \in F^C_{t_f - t_0}, \forall \alpha \geq 0$, the optimization algorithm given previously does not terminate until time $t_N$ and generates a solution trajectory $p_1$ satisfying $p_1(t) \in G^C_\infty, \forall t \in [t_0, t_N]$.

*Proof:* For $k = 0$, the condition on $p_1(t_0)$ is satisfied by assumption. For the inductive step, assume $p_1(t_k) + \alpha h_{t_k} \in F^C_{t_f - t_k}, \forall \alpha \geq 0$ for $k \geq 0$. Performing the optimization program at time $t_k$, we note that $u = [\overline{u}_1 \ 0]^T$ satisfies the conditions on the inputs, and generates the trajectory $p_1(t) = p(t_k) + \overline{u}_1(t - t_k) h_{t_k}$ for $t \in [t_k, t_{k+1}]$. Noting that the speed input for an aircraft is nonnegative, we have $\overline{u}_1 \geq \underline{u}_1 \geq 0$, and so $p_1(t) \in H_1(p(t_k), F_{t_f - t_k})^C, \forall t \in [t_k, t_{k+1}]$. Thus, the optimization program at time $t_k$ is feasible. Let $u^*$ be the optimal input, and $p_1^*$ be the corresponding trajectory on $[t_k, t_{k+1}]$. Then $p_1^*(t) \in H_1^C \subset F^C_{t_f - t_k} \subset F^C_{t_f - t_{k+1}} \subset G^C_\infty, \forall t \in [t_k, t_{k+1}]$. In particular, $p_1^*(t_{k+1}) \in H_1^C$. Furthermore, due to the constraint $u_2^* T \in [-\psi_1(t_k), \psi_2(t_k)]$, the heading at $t_{k+1}$ satisfies $p_1(t_{k+1}) + \alpha h_{t_{k+1}} \in H_1^C \subset F^C_{t_f - t_k} \subset F^C_{t_f - t_{k+1}}, \forall \alpha \geq 0$. Then by induction, the result follows. ■

### B. Convex Optimization Formulation Using Linear Dynamics

We note that the optimization program given previously is non-convex, due to the nonlinear dynamics of $p_1(t, u)$. However, using the approximate linear dynamics given by equations (4) and (5), and after some simplifications, we can formulate the optimization program at each time step as a quadratic optimization program with linear constraints

*Quadratic Program at Time $t_k$*

$$\begin{array}{ll} \text{minimize} & \left\| p_1(t_{k+1}) - p_{final} \right\|_2^2 \\ \text{subject to} & a^T B(t_{k+1}) u \leq 0 \\ & u_2 T \in [-\psi_1(t_k), \psi_2(t_k)] \\ & u_1 \in [\underline{u}_1, \overline{u}_1] \\ & u_2 \in [\underline{u}_2, \overline{u}_2] \end{array}$$

Details on the calculation of parameters $a$, $b$, $\psi_1$, and $\psi_2$ have been omitted for the sake of space, but they follow in a straightforward manner from geometric argument. The important point to note here is the computational efficiency of the above optimization program. By overapproximating the unsafe set at each time step using a simple geometric object, one arrives at a slightly conservative but highly efficient online trajectory optimization problem, in the presence of an unknown moving obstacle.

### C. Initialization Condition and Robust Halfspace Constraint

As seen previously, the result of Proposition 1 requires that the initial condition $p_1(t_0)$ satisfy $p_1(t_0) + \alpha h \in F^C_{t_f - t_0}, \forall \alpha \geq 0$, or equivalently that the heading direction lies outside the tangent lines to the disc $F_{t_f - t_0}$. This condition can be ensured by first performing a turn maneuver either to the left or right, until the heading direction is outside the tangent lines. Note that at most one has to turn by 90 degrees or $\pi/2$ radians. For the kinematics of a single vehicle given in equation (1), the number of iterations for this initialization procedure can be upper bounded by $N_{max} = \left\lceil \frac{\pi}{2\bar{u}_2 T} \right\rceil$. Since $F_{t_f - t_0}$ is fixed for given pursuer-evader input bounds, one can perform backward reachability analysis to give condition on the minimum separation at which the optimization procedure can be initialized safely.

We note that the quadratic program given in the previous section uses a linearized system model, and so the model errors with respect to the nonlinear system dynamics given in equations (2) and (3) have to be sufficiently accounted for in the constraints of the optimization program to ensure safety for the nonlinear system. As we make use of standard procedures for producing robust halfspace constraints, some details have been omitted. The interested reader may refer to [27] for further information on this subject. The important points specific to the current problem are sketched out below.

It can be observed that as long as $p_1(t_k)$ does not lie exactly on $F_{t_f - t_k}$ at time $t_k$, a family of tangent lines can be used for the optimization program given here. Consider the relative position vector $\Delta p(t_k)$ between the position of the pursuer and the evader at time $t_k$. Instead of calculating the tangent lines from the position of the evader $p_1(t_k)$, one can choose to calculate a separate set of tangent lines from a point closer to the pursuer but still outside the constraint set $F_{t_f - t_k}$. Specifically, one may consider the point $\tilde{p} = p_1(t_k) + \beta \frac{\Delta p(t_k)}{\|\Delta p(t_k)\|_2}$, where $\beta \in [0, \|\Delta p(t_k)\|_2 - r_k]$. Recall that $r_k$ is the radius of the constraint set $F_{t_f - t_k}$. From this, one obtains a new set of parameters $\tilde{a}$, $\tilde{b}$, $\tilde{\psi}_1$, and $\tilde{\psi}_2$, which can be used in a robust formulation of the optimization program.

In fact one can show that when the position of the evader at time $k$ is far enough from the constraint set $F_{t_f - t_k}$, specifically when $\|\Delta p(t_k)\|_2 - r_k \geq E_{max}$, where $E_{max}$ is a parameter depending on the model error in each time interval, we can choose

$$\beta = \frac{\|\Delta p(t_k)\|_2 E_{max}}{r_k + E_{max}} \qquad (6)$$

Then, the resulting tangent lines calculated from the corresponding $\tilde{p}$ satisfy the safety constraints under the nonlinear model. We note that for the input bounds chosen in section IV, and sampling interval $T = 10s$, $E_{max} \approx 0.248$ from a straightforward Taylor series analysis.

For the case where $\|\Delta p(t_k)\|_2 - r_k < E_{max}$, i.e. when the position of the evader is too close to the constraint set at time $k$, one can select any input of the form $u = [v \quad 0]^T$, $v \in [\underline{u}_1, \bar{u}_1]$. Then by the assumptions given in section V-A, the trajectory generated by the nonlinear dynamics remains safe in the interval $[t_k, t_{k+1}]$.

## VI. Results

Here we show some results obtained from applying the optimization procedure to simulated aircraft conflict scenarios. For all scenarios described here, the input bounds assume the values given in section IV, which are chosen based upon realistic air traffic data. The destination point in all cases is $p_{final} = [100nmi \ 0]^T$, while the sampling period is given by $T = 10s$.

First, we performed the optimization procedure on a series of encounter scenarios where the evader and pursuer use straight line nominal trajectories with a fixed path crossing angle $\phi_0$ that would result in an exact collision. The optimization procedure is performed for a finite time horizon of 10 minutes, which for our sampling period correspond to the discrete extent $N = 60$. The results are shown in Figure 5 for the cases $\phi_0 = \pi/2$, $\phi_0 = -3\pi/4$, and $\phi_0 = \pi$. Clearly, in each case, the resulting trajectory remains outside the unsafe backward reachable set $G_\infty$ throughout the optimization time horizon. In Figure 5(b), we zoom in on the trajectory obtained for $\phi_0 = \pi/2$. It can be seen that the trajectory follows the constraint set tightly up to the robustness bound characterized in section V-C. The same behavior can be observed for the case where $\phi_0 = -3\pi/4$. However, as can be seen in the case where $\phi_0 = \pi$, conservative trajectories maybe produced to account for unexpected turns by the adversary. Given that the simulations are performed using nonlinear system dynamics, the robust halfspace constraints are shown in these examples to sufficiently account for errors incurred from model linearization.

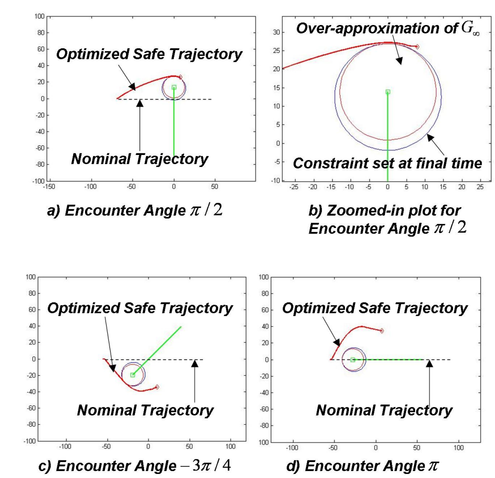


Fig. 5. Results of trajectory optimization in straight line encounter scenarios for path crossing angles (a)$\phi_0 = \pi/2$, (b)$\phi_0 = \pi/2$(zoomed-in), (c)$\phi_0 = -3\pi/4$, (d)$\phi_0 = \pi$.

For known pursuer trajectories or ones that can be reasonably predicted, many existing trajectory optimization procedures can provide less conservative results. However, the advantage of our proposed algorithm lies in the fact that it can guarantee safety regardless of the trajectory of the pursuer, with respect to the assumed system model and input bounds. Some scenarios involving random inputs by the adversary aircraft are shown in Figure 6. In these simulations, the nominal trajectory results in a head-on collision between the evader and the pursuer, and the pursuer inputs are drawn randomly from within the input bounds at each time step. It can be observed that applying the optimization procedure results in a safe trajectory even under random disturbance inputs. The perturbed trajectories of the pursuer aircraft shown in these simulations justify the generation of what appears to be conservative trajectories in Figure 5(d).

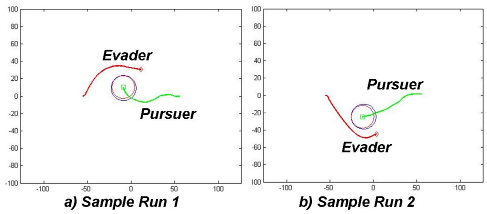


Fig. 6. Trajectory optimization results for random disturbance inputs at nominal path crossing angle $\phi_0 = \pi$.

As can be seen from the formulation of the convex optimization program, the computation time is expected to be very fast. In the example scenarios given above, the computation time for each step of the optimization algorithm was on the order of 1 second, as implemented in the MATLAB environment using the CVX toolbox [28].

## VII. Future Work

To achieve guaranteed safety under uncertainty, this optimization algorithm uses a worst case approach towards conflict resolution. As such, the resulting conflicting trajectories are more conservative as compared with those generated by algorithms in cooperative scenarios. Future work on this project could include relaxing the worst case constraint set if the intent of the adversary aircraft is partially known. Furthermore, one may consider extensions to the multiple adversary aircraft case, where we modify the constraint of the problem by forming intersections of half space constraints. Finally, one may also wish to formulate three dimensional conflict resolution problems where one allows vertical evasive maneuvers.

## References


[1] J. Jackson and S. Green, "Control applications and challenges in air traffic management," *American Control Conference, 1998. Proceedings of the 1998*, vol. 3, pp. 1772–1788 vol.3, Jun 1998.

[2] C. Tomlin, I. Mitchell, A. Bayen, and M. Oishi, "Computational techniques for the verification of hybrid systems," *Proceedings of the IEEE*, vol. 91, no. 7, pp. 986–1001, July 2003.

[3] J. Kuchar and L. Yang, "A review of conflict detection and resolution modeling methods," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 1, no. 4, pp. 179–189, Dec 2000.

[4] J. Hu, M. Prandini, and S. Sastry, "Optimal coordinated maneuvers for three dimensional aircraft conflict resolution," *AIAA J. Guidance, Control and Dynamics*, vol. 25, no. 5, pp. 888–900, 2002.

[5] A. Richards, J. Bellingham, M. Tillerson, and J. How, "Coordination and control of multiple UAVs," in *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Monterey, California, August 2002.

[6] T. Schouwenaars, J. How, and E. Feron, "Decentralized cooperative trajectory planning of multiple aircraft with hard safety guarantees," in *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Providence, Rhode Island, August 2004.

[7] A. Bicchi and L. Pallottino, "On optimal cooperative conflict resolution for air traffic management systems," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 1, no. 4, pp. 221–231, Dec 2000.

[8] J. Mejia and D. Stipanovic, "Safe trajectory tracking for the two-aircraft system," *Electro/Information Technology, 2007 IEEE International Conference on*, pp. 362–367, May 2007.

[9] J. Kosecka, C. Tomlin, G. Pappas, and S. Sastry, "2-1/2D conflict resolution maneuvers for ATMS," *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, vol. 3, pp. 2650–2655 vol.3, 1998.

[10] I. Mitchell, A. Bayen, and C. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *Automatic Control, IEEE Transactions on*, vol. 50, no. 7, pp. 947–957, July 2005.

[11] C. Tomlin, I. Mitchell, and R. Ghosh, "Safety verification of conflict resolution manoeuvres," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 2, no. 2, pp. 110–120, Jun 2001.

[12] R. Teo and C. J. Tomlin, "Computing danger zones for provably safe closely spaced parallel approaches," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 3, pp. 434–443, May-June 2003.

[13] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 72–89, 1986.

[14] M. Erdmann and T. Lozano-Perez, "On multiple moving objects," *Algorithmica*, vol. 2, no. 4, p. 477521, 1987.

[15] J. Reif and M. Sharir, "Motion planning in the presence of moving obstacles," *Foundations of Computer Science, 1984., 26th Annual Symposium on*, pp. 144–154, Oct. 1985.

[16] K. Fujimura and H. Samet, "A hierarchical strategy for path planning among moving obstacles [mobile robot]," *Robotics and Automation, IEEE Transactions on*, vol. 5, no. 1, pp. 61–69, Feb 1989.

[17] C. Shih, T. Lee, and W. Gruver, "Motion planning with time-varying polyhedral obstacles based on graph search and mathematical programming," *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pp. 331–337 vol.1, May 1990.

[18] D. Hsu, R. Kindel, J.-C. Latombe, and S. Rock, "Randomized kinodynamic motion planning with moving obstacles," *The International Journal of Robotics Research*, vol. 21, no. 3, pp. 233–255, 2002.

[19] S. M. Lavalle and J. J. Kuffner, "Randomized kinodynamic motion planning," *The International Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.

[20] F. Lamiraux and J. Laumond, "On the expected complexity of random path planning," *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 4, pp. 3014–3019 vol.4, Apr 1996.

[21] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2, pp. 500–505, Mar 1985.

[22] B. Krogh and C. Thorpe, "Integrated path planning and dynamic steering control for autonomous vehicles," *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, vol. 3, pp. 1664–1669, Apr 1986.

[23] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fact mobile robots," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 19, no. 5, pp. 1179–1187, Sep/Oct 1989.

[24] I. Mitchell, "Games of two identical vehicles," Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, Tech. Rep.

[25] A. M. Bayen, "Computational control of networks of dynamical systems: Application to the national airspace system," Ph.D. dissertation, Stanford University, Stanford, CA, 2003.

[26] I. M. Mitchell and J. A. Templeton, "A toolbox of Hamilton-Jacobi solvers for analysis of nondeterministic continuous and hybrid systems," *Hybrid Systems Computation and Control*, vol. 3414/2005, pp. 480–494, Feb. 2005.

[27] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[28] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming," 2008. [Online]. Available: http://http://www.stanford.edu/ boyd/cvx/