Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

UAV collision avoidance exploitation for noncooperative trajectory modification ☆

Pietro Pierpaoli [a], Amir Rahmani [b]

[a] Mechanical and Aerospace Engineering, University of Miami, FL 33146, United States
[b] Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, United States

ARTICLE INFO

Article history:
Received 28 March 2017
Received in revised form 4 December 2017
Accepted 6 December 2017
Available online xxxx

Keywords:
Next-gen trustworthiness
Autonomous flight
ADS-B
Collision avoidance
Model predictive control
Autonomous system vulnerability

ABSTRACT

Distributed collision-free trajectories are generally obtained through a continuous sharing of information between vehicles. With the intent of investigating possible sources of vulnerability in autonomous frameworks, we formalize a procedure malicious players can follow to influence other. In this paper we propose a strategy for steering a UAV towards predetermined targets. The strategy described here relies on the existence of a flight information sharing protocol (i.e. ADS-B) and predictable collision avoidance algorithms. A model predictive controller is applied to the switching system representing a pair of UAVs coupled by the presence of an imminent collision. As showed by means of numerical simulations and robot experiments, the result is a loss of autonomy on the UAV. Our results suggest the need to include the subject of our study in the discussion on safe automated airspace.

1. Introduction

Autonomous guidance and navigation have become major subjects in aerospace science during the past two decades. Today, the family of unmanned aerial vehicles (UAV) and air systems (UAS) includes a surprisingly high variety of different kinds of vehicles, such as fixed wing aircraft, multicopters and aerobots. Fields of application are extremely vast; e.g. environmental surveillance [1], demonstrators for extraterrestrial missions or aerial manipulators for physical interaction with the environment [2] [3].

The gap between remotely controlled and fully autonomous aircraft is progressively reducing thanks to recent improvements in guidance, navigation and control techniques. In addition, the large diffusion of low cost hardware also contributes to the diffusion of privately owned UAVs. Therefore, the commitment from both scientific and regulatory communities is twofold: meet the extraordinary diversification in user requests while reducing threats and hazards.

Finally, in order to accommodate the increasing air traffic volume, a satellite-based network is under realization and will require aircraft to share navigation information with ground bases and neighboring aircraft. The well known Automatic Dependent Surveillance – Broadcast (ADS-B) [4] represents a first step in this direction.

1.1. Safety in autonomous navigation

Due to the increasing degree of automation in ATM, *cyber-physical* security has become a major topic of interest. Both unintentional malfunctions and adversarial attacks must be investigated. For example, a performance optimization technique aimed to reduce interference between radar and ADS-B interference is proposed in [5].

Malicious attacks can be targeted to both ground stations and aircraft themselves. Three general classes of attacks have been identified by [6]: *interceptions of transmission*, *jamming* and *injection of messages* into a data link. In addition, several other inherent GPS vulnerabilities have been investigated over the years [7] [8]. Jamming of wireless networks is considered a *denial-of-service* attack caused by the presence of a high power transmitting device which prevents connectivity with other lower power devices [9]. Cross-verification of ADS-B data through sensor fusion have been proposed to improve trustworthiness of broadcast messages and GPS data disruptions [10] [11] [12].

1.2. Collision detection and resolution

When dealing with safety in autonomous systems, a major class of threats is represented by *losses of minimum separation* with

☆ This research was partially supported by the University of Miami graduate research grant.

E-mail address: p.pierpaoli@umiami.edu (P. Pierpaoli).

https://doi.org/10.1016/j.ast.2017.12.008
1270-9638/© 2017 Published by Elsevier Masson SAS.

surrounding objects or vehicles. Conflict detection and resolution (CDR) techniques for autonomous vehicles have been largely investigated in the last few decades. Although many solutions have been proposed, dynamic environments and online real-time collision resolution protocols are still active areas of research.

Though computationally expansive, several methods for optimal definition of conflict-free trajectories have been proposed. In [13], a linear mixed integer programming minimizes the total distance traveled by the aircraft. Although applicable to relative crowded scenarios, the proposed methods assumes complete cooperation between aircraft and is solved in a centralized fashion. In [14], collision-free, piece-wise polynomial trajectories are produced through a centralized mixed integer quadratic programming for formations of agile quadrotors. In [15], single artificial potential fields are associated to each obstacle (high potentials) and goal (low potentials); a control law is then defined as a smooth navigation function that follows the descending gradient of the total resulting potential. Other results on the subject include Rapidly-Exploring Random Tree [16], control barrier functions [17] [18] [19], multi-agent hybrid dynamics [20] and reachability analysis [21]. A detailed survey on relevant collision avoidance techniques for UAV applications can be found in [22].

When large number of vehicles are considered, decentralized solutions are required. Several geometric reactive CDR methods are available in the literature and can be implemented in decentralized fashion. In our work, we employ an approach derived from the Velocity Obstacle (VO) method, first presented in [23]. VO has similar features with the Collision Cone method [24], where collision configurations are mapped into agents' absolute velocity space, and with the Forbidden Velocity Map by [25]. These techniques reduce the collision problem to the identification of a set of velocity vectors, that results in a collision in finite time. The VO method has also been integrated with dynamic obstacles and reciprocal configurations in [26],[27]. Successful applications of VO include marine vehicle sea traffic rule implementation [28], UAV [29,30], human-robot interaction [31] and highway obstacle detection [32].

1.3. A novel class of vulnerability

In the context of CDR techniques design 1) collision events and 2) deviation from the original path are usually considered to be the main sources of penalty for an algorithm. Another class of vulnerability will be defined here. As introduced by the authors in [33] and [34], the presence of a collision avoidance protocol in the aircraft control algorithm introduces a fundamental dependency from the external environment (Fig. 1). Upon detection of collisions, original navigation plans must be revised and trajectories modified in accordance with some predefined rules. In this sense, as it will be made clear later, the event of a collision represents a constraint on the vehicle's dynamics. In this work we show that when deterministic collision resolution policies are engaged, avoiding maneuvers can be anticipated. Consequentially, the dependency from the external environment can be exploited with the consequent loss of the vehicle's autonomy. The resulting threat is emphasized when the exact location of an aircraft is known (i.e. ADS-B). Although strictly connected with the *intercept of transmissions*, our contribution introduces a fundamental new class of vulnerability.

On one side, an autonomous vehicle employing a collision avoidance algorithm (*evader*) tries to reach a fixed destinations. On the other side, an opponent agent (*pursuer*), without adopting any kind of *sense and avoid*, aims to steer the evader towards a predefined target set. The purpose of our work is twofold: first, we investigate the dynamics of two autonomous vehicles coupled by a uni-lateral collision avoidance algorithm; then, we define a strategy the pursuer agent can adopt in order to steer the evader towards the target set. A full understanding of the problem is the primary objective of this work, which tries to promote future developments of collision avoiding strategies capable of mitigating the highlighted vulnerability.

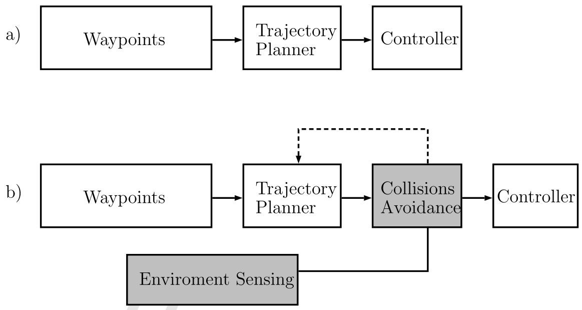


Fig. 1. Autonomous mobile agent layered structure for motion planning without (a) and with (b) sense and avoid layer.

In section 2, we introduce main assumptions and derive the equations for our model. In section 3, we describe a strategy composed by two sequential maneuvers that can be used to drive an evader towards a desired capture set. Finally, in section 4, we present the results from numerical simulations and experiments conducted on differential drive robots, demonstrating the application of our strategy.

2. Model definition

In this section we introduce the model describing a pair of vehicles coupled by a collision avoidance algorithm acting on only one of them. Planar motions within a two dimensional space are considered. When a layered airspace structure does not allow for significant changes in altitude, this assumption reflects a real constraint. In addition, aerodynamic wakes produced by rotor-craft, such as helicopters or multi-copters, prevent the vehicles from being vertically overlapped. However, for more general frameworks, this simplification is introduced to limit the complexity of the problem and it represents a first step towards a complete understanding of the problem.

In a global reference frame $\mathcal{G}$ attached to the earth, vehicle configurations are denoted by $(q,\phi)^T$; agent positions and angular orientations (or headings) are dented by $q=(x,y)^T\in\mathbb{R}^2$ and $\phi\in[0,2\pi)=\mathbb{S}$ respectively. Given the planar domain and considering small variations in the velocity vector, UAV dynamics can reasonably be approximated by unicycle dynamics:

$$\begin{aligned}\dot{x} &= v\cos\phi\\ \dot{y} &= v\sin\phi\\ \dot{\phi} &= \omega.\end{aligned}\tag{1}$$

Communication delays or interruptions are neglected, and the sharing of information takes place in real time. At each time, the information set for each agent is represented by the current neighbor configurations. This complete information pattern can be obtained in different ways, e.g. appropriate on-board sensors, network based sharing protocols (i.e. ADS-B), overhead tracking systems or a combination of them.

Finally, we assume avoiding maneuvers to be composed of heading maneuvers only; no variation in speed occur while performing avoiding maneuvers. This assumption is introduced in order to simplify the analysis of the problem. However, for fuel economy, a purely turning maneuver could in fact be the preferred choice when compared with speed or altitude changes.

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

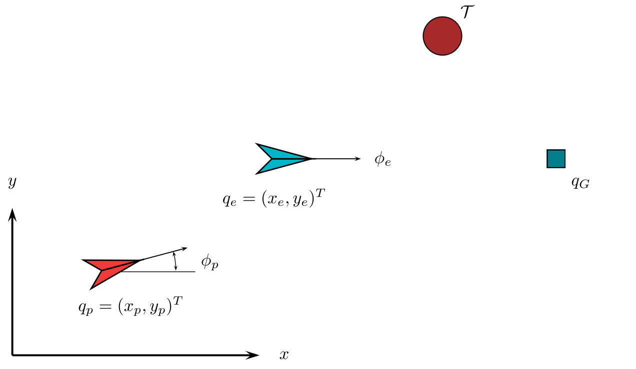


Fig. 2. The problem involves two types of agents; an *evader* agent (blue in figure), whose objective is to reach a point q_G, and a *pursuer* agent (red in figure), whose objective is to steer the evader into the capture set $\mathcal{T}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

We define $\mathcal{T} \subset \mathbb{R}^2 \times \mathbb{S}$ to be the set of target poses toward which the purser aims to steer the evader; $q_G \in \mathbb{R}^2$ is the evader's fixed goal (Fig. 2).

2.1. Revised collision cone method

In cooperative navigation frameworks, the communication of intents between agents and the agreement on traffic rules allows a sharing of the avoiding effort between neighbors. Conversely, in non-cooperative scenarios, decisions are based on frequently updated measurements. In this case, online reactive CDR algorithms are best suited. Reactive CDR algorithms rely on the projection of the current neighbors' configuration within a finite horizon future. Three different kinds of projections are possible: 1) linear, 2) worst case and 3) probabilistic ([22]). In this work, we introduce our reactive CDR algorithm based on a revised concept of *Velocity Obstacle* (VO) method. The idea behind VO is to define a set of velocity vectors for which, given the current neighbors' poses and assuming no variation in their velocities, a collision will occur within a finite time window. VO assumes linear trajectory projections, i.e. obstacles are assumed to move at constant velocities between consecutive iterations of the algorithm.

Assuming identical, circular protected zones of radius R_{pz}, and denoting $\|\cdot\|$ the Euclidean norm, we introduce the following definition:

Definition 1. A *collision* event occurs when the distance between two agents is less than the protected zone radius, i.e. agents E and P experience a collision if and only if $\|q_e - q_p\| < R_{pz}$.

Note that the definition above does not differ from considering collisions as the intersection of two protected zones. In fact, R_{pz} can be assumed to be the sum of both agents radii; without loss of generality, this case can be reduced to a point-circle collision problem, where the circle has radius equal to $2R_{pz}$.

With reference to Fig. 3(a), consider the collision between a point E and a circle with radius R_{pz}, centered in P; let the points be separated by the vector $d = q_e - q_p$. The relative velocity (red in Fig. 3(b)) is defined as:

$$v_{ep} = v_e - v_p = \begin{bmatrix} v_e \cos\phi_e - v_p \cos\phi_p \\ v_e \sin\phi_e - v_p \sin\phi_p \end{bmatrix}. \tag{2}$$

Denoting by λ_f and λ_r the acute angles formed by the two tangents from E to P's protected zone, we can write:

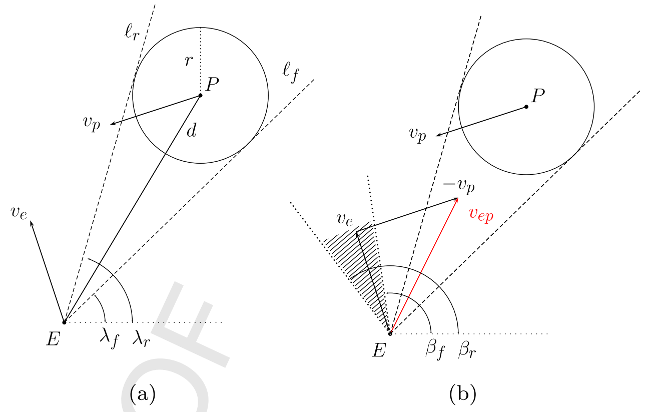


Fig. 3. Collision avoidance technique geometry and nomenclature relative to the vehicle pair E and P separated by distance d. Agent P has circular protected area of radius r. (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

$$\begin{aligned} \angle\ell_f = \lambda_f &= \arctan\left(\frac{y_p - y_e}{x_p - x_e}\right) - \arcsin\left(\frac{R_{pz}}{\|q_e - q_p\|}\right) \\ \angle\ell_r = \lambda_r &= \arctan\left(\frac{y_p - y_e}{x_p - x_e}\right) + \arcsin\left(\frac{R_{pz}}{\|q_e - q_p\|}\right). \end{aligned} \tag{3}$$

Theorem 1. *At time t_0, consider two agents E and P with fixed velocity and heading. A collision between E and P occurs at time t, with $t_0 < t < \infty$, if and only if the relative velocity vector v_{ep} belongs to the planar sector formed by the two tangents to the protected area of agent P having origin in E, that is, if:*

$$\lambda_f < \arctan\frac{\sin\phi_{ep}}{\cos\phi_{ep}} < \lambda_r. \tag{4}$$

See [30] for a complete proof of this theorem. A revised version of the same proof is applied to the following theorem.

Theorem 2. *When the relative velocity vector v_{ep} is oriented as λ_f, i.e.:*

$$\phi_{ep} = \arctan\left(\frac{y_p - y_e}{x_p - x_e}\right) + \arcsin\left(\frac{R_{pz}}{\|q_e - q_p\|}\right)$$

then, $\min\|d(t)\| = R_{pz}$.

Proof. From the definition of v_{ep} in (2), the distance between E and P is $d(t) = d_0 - v_{ep}\,t$. The minimum separation between agents corresponds to the following:

$$\begin{aligned} \frac{d}{dt}\left(\frac{1}{2}\|d(t)\|^2\right) &= \|d(t)\|\,\frac{\frac{d}{dt}d(t)^T\,d(t)}{\|d(t)\|} \\ &= -v_{ep}^T(d_0 - v_{ep}\,t) = 0 \end{aligned}$$

where v_{ep} is assumed to be constant between iterations of the VO algorithm. Rearranging and solving for t_m, the time at which the two vehicles experience the minimum separation:

$$0 = v_{ep}^T\,d_0 - v_{ep}^T\,v_{ep}\,t_m$$

from which:

$$t_m = \frac{v_{ep}^T\,d_0}{\|v_{ep}\|^2}. \tag{5}$$

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

Referencing Fig. 3(b), assuming $\phi_{ep} = \lambda_f$ and (3), we have:

$$\begin{aligned} v_{ep}^T d_0 &= \|v_{ep}\|\,\|d_0\| \cos(\angle v_{ep} - \angle d_0) \\ &= \|v_{ep}\|\,\|d_0\| \cos\left(\phi_{ep} - \arctan\frac{y_p - y_e}{x_p - x_e}\right) \\ &= \|v_{ep}\|\,\|d_0\| \cos\left(\arcsin\frac{R_{pz}}{\|d_0\|}\right). \end{aligned} \tag{6}$$

When $\phi_{ep} = \lambda_f$, the time to the minimum distance defined in (5) reduces to:

$$\begin{aligned} t_m &= \frac{\|v_{ep}\|\,\|d_0\|}{\|v_{ep}\|^2} \cos\left(\arcsin\frac{R_{pz}}{\|d_0\|}\right) \\ &= \frac{\|d_0\|}{\|v_{ep}\|} \cos\left(\arcsin\frac{R_{pz}}{\|d_0\|}\right). \end{aligned} \tag{7}$$

and the distance can be expressed as:

$$\begin{aligned} \|d(t)\| &= \sqrt{(d_0 - v_{ep}\,t)^T (d_0 - v_{ep}\,t)} \\ &= \sqrt{\|d_0\|^2 - 2\,v_{ep}^T\,d_0\,t + v_{ep}^T\,v_{ep} t^2}. \end{aligned}$$

Finally, at the time of minimum separation, assuming again $\phi_{ep} = \lambda_f$, the last result is obtained by substituting (5) for $v_{ep}^T d_0$ and (7) for t:

$$\begin{aligned} \|d_m(t)\| &= \sqrt{\|d_0\|^2 - 2\,\|v_{ep}\|^2\,t_m^2 + \|v_{ep}\|^2 t_m^2} \\ &= \sqrt{\|d_0\|^2 - \|v_{ep}\|^2\,t_m^2} \\ &= \sqrt{\|d_0\|^2 - \frac{\|v_{ep}\|^2\,\|d_0\|}{\|v_{ep}\|} \cos^2\left(\arcsin\frac{R_{pz}}{\|d_0\|}\right)} \\ &= \sqrt{\|d_0\|^2 \left(1 - \cos^2\left(\arcsin\frac{R_{pz}}{\|d_0\|}\right)\right)} \\ &= \sqrt{\|d_0\|^2 \left(\sin^2\left(\arcsin\frac{R_{pz}}{\|d_0\|}\right)\right)} = R_{pz}. \end{aligned} \tag{8}$$

The extension to $\phi_{ep} = \lambda_r$ is obtain similarly. From the result in Theorem 2 we know the heading value vehicle E should choose in order to have the minimum allowed separation from E; this result is relevant for the avoiding policy introduced in section 2.2.

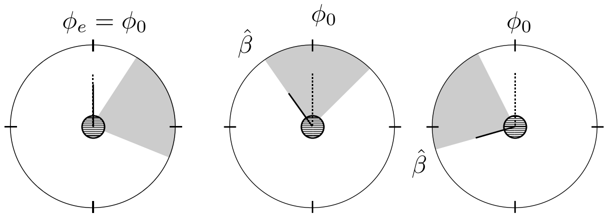
Equation (2) describe the mapping from relative to agents velocity. We denote by β_f and β_r the evader's heading corresponding to ϕ_{ep} oriented as λ_f or λ_r respectively. Hence, when $\phi_{ep} = \lambda_f$, using equation (2):

$$\frac{v_e \sin\beta_f - v_p \sin\phi_p}{v_e \cos\beta_f - v_p \cos\phi_p} = \frac{v_{ep} \sin\lambda_f}{v_{ep} \cos\lambda_f} = \tan\lambda_f. \tag{9}$$

Rearranging:

$$v_e(\sin\beta_f - \tan\lambda_f \cos\beta_f) + v_p(\tan\lambda_f \cos\phi_p - \sin\phi_p) = 0. \tag{10}$$

Equation (10) defines the mapping from the relative velocity space into agents' absolute headings. Letting ϕ_{ep} span the set of angles between λ_f and λ_r, it is possible to define the complete set of *collision headings* B_{ep} for the agent E. At time t, the set of heading angles $B_{ep}(t) \subset [0, 2\pi)$ for which E will eventually experience a loss of minimum separation is defined as[1]:



(a) Available ϕ_0 (b) Obscured and not available ϕ_0 (c) Not obscured but not available ϕ_0

Fig. 4. Examples of deconfliction policy outcomes; whenever the desired heading ϕ_0 belongs to the set of collision angles B_{ep} (shaded area) or B_{ep} is between current and desired heading, the evader switches its heading towards the closest boundary of B_{ep} aiming for a grazing maneuver.

$$B_{ep} = \{\beta \in [0, 2\pi) : \|q_e(\tau) - q_p(\tau)\| < R_{pz},\ t < \tau < \infty\}. \tag{11}$$

When n neighbors exist, each conflict is considered separately and the final set of infeasible headings B_{ep} can be defined as the union of the single collision sets B_{ep_i}, for $i = 1, \dots, n$:

$$B_{ep} = \bigcup_{i=1}^{n} B_{ep_i} = \{\beta_f^1, \dots, \beta_r^1\} \cup \dots \cup \{\beta_f^n, \dots, \beta_r^n\}. \tag{12}$$

2.2. Definition of the conflict resolution policy

In addition to a collision detection algorithm, autonomous navigation requires a *deconfliction policy* in order to produce appropriate evading maneuvers such that 1) the minimum separation from obstacles, 2) the least deviation from original trajectories and 3) arrival to the desired destinations are guaranteed.

In a cooperative environment, predefined engagements protocols or traffic rules are used to distribute the evading effort among the agents, e.g. available fuel, aircraft maneuverability or current phase of flight. In non-cooperative environments, agents must assume that no avoiding effort is taken by neighbors. We introduce here the deconfliction policy assumed in the rest of the paper. With reference to Fig. 4, assume that the evader is initially flying with heading $\phi_e = \phi_0$. When an incoming conflict is detected, the evader sets its desired heading to a new feasible heading $\phi_e \notin B_{ep}$.

Definition 2. An *evading maneuver* is a change of heading towards a new value $\hat{\beta}$, such that:

$$\hat{\beta} = \underset{\beta \in B_{ep}}{\arg\min}\ \delta(\phi_e, \beta) \tag{13}$$

where $\delta(\cdot,\cdot) : [0, 2\pi) \times [0, 2\pi) \to \mathbb{R}$ is a distance function between angles that takes into account the periodicity of the angular dimension.

We also assume that if $B_{ep} \neq \emptyset$ at time t_0, then $\phi_e \notin B_{EP}$ for all $t_0 < t < \infty$ (Fig. 4(c)); this means that the agent is never allowed to cross the set of collision headings in order to resolve an incoming conflict. Considering smaller UAVs, where size and power resources are more limited, it might not be possible to have an accurate estimation of the expected time to impact. For this reason, we do not consider resolution strategies that bring aircraft on a collision route as part of an evading maneuver.

2.3. Complete system

The evader's objective is to reach its final goal while avoiding collisions. Using the collision avoidance technique defined in the previous section, we introduce a suitable model for the dynamics of the evader. To this end, the evading policy, can be incorporated

[1] For convenience, we drop the time superscript on B_{ep}, acknowledging that the set is updated on each iteration of the collision avoidance algorithm.

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

in the evader dynamics by switching between behaviors. We index the two possible states of the system using a boolean collision variable $\nu = [0, 1]$. We refer to the state corresponding to $\nu = 0$ as *collision free state*, i.e. no imminent collisions are detected, and to $\nu = 1$ as *collision state*, i.e. when agents are heading for a collision. Transitions from one state to the other depends on current neighbors' states, which are processed online by the collision detection algorithm. Therefore, (1) can be updated as:

$$\begin{aligned} \dot{x}_e &= v_e \cos\phi_e \\ \dot{y}_e &= v_e \sin\phi_e \\ \dot{\phi}_e &= \omega_e(\phi_0, \phi_e, \hat{\beta}, \nu) \end{aligned} \qquad \nu = \begin{cases} 1 & \text{if } \phi_0 \in B_{EP} \\ 0 & \text{if } \phi_0 \notin B_{EP} \end{cases} \tag{14}$$

where the control function $\omega_e(\cdot) : \mathbb{S} \times \mathbb{S} \times \mathbb{S} \times \{0, 1\} \to \mathbb{R}$ computes the input angular velocity from $\{\phi_0, \phi_e, \hat{\beta}, \nu\}$. Note that the exact definition of the aircraft heading control function is beyond the scope of this paper; for our simulations and experiments, a simple proportional controller was used; e.g. $\dot{\phi}_e = \kappa(\phi_0(1-\nu) + \hat{\beta}\nu - \phi_e)$, with $\kappa \in \mathbb{R}_+$ being an appropriate controller gain. The set of equations (14) reveals the fundamental dependency between evader trajectory and its neighbors' states.

Using the same unicycle model introduced in (1), pursuer dynamics, which are not affected by collision avoidance are given by:

$$\begin{aligned} \dot{x}_p &= v_p \cos\phi_p \\ \dot{y}_p &= v_p \sin\phi_p \\ \dot{\phi}_p &= \omega_p \end{aligned} \tag{15}$$

Finally, including the collision avoidance constraints (10) and (3), the complete system can be represented as:

$$\begin{aligned} &\dot{x}_e = v_e \cos\phi_e \\ &\dot{y}_e = v_e \sin\phi_e \\ &\dot{\phi}_e = \omega_e(\phi_0, \phi_e, \hat{\beta}, \nu) \qquad \nu = \begin{cases} 1 & \text{if } \phi_0 \in B_{ep} \\ 0 & \text{if } \phi_0 \notin B_{ep} \end{cases} \\ &\dot{x}_p = v_p \cos\phi_p \\ &\dot{y}_p = v_p \sin\phi_p \\ &\dot{\phi}_p = \omega_p \\ &v_e\left(\tan\lambda \cos\hat{\beta} - \sin\hat{\beta}\right) + v_p\left(\tan\lambda \cos\phi_p + \sin\phi_p\right) = 0 \\ &\lambda - \arctan\left(\frac{y_p - y_e}{x_p - x_e}\right) \mp \arcsin\left(\frac{R_{pz}}{\|q_p - q_e\|_2}\right) = 0, \end{aligned} \tag{16}$$

where $\hat{\beta}$ is the closest boundary of the unfeasible heading set B_{ep}, defined in (13).

The switched system defined in (16) is a set of differential-algebraic equations (DAE) representing the dynamics of two vehicles E and P coupled by the *unilateral collision avoidance* framework described in this section. The switching behavior of the system is governed by the variable ν; when $\nu = 1$, the evader trajectory is constrained by the presence of an incoming conflict.

2.4. Connection with game theory

Beside the nomenclature we used, the problem presented in this section has *game theoretic* features. To this extent, a noticeably similar problem was studied in [21], where a two-vehicle game problem is formulated in order to define a collision reachability set.

In game theory, a *game* ([35]) is defined as an interactive decision-making problem where each player chooses its action in order to maximize a certain gain (payoff). However, given the deconfliction policy defined in this section, once a collision is detected, the evading action is uniquely defined. In order for our problem to be considered a game, more relaxed assumptions for the evading maneuver should be considered (i.e. evader's variable velocity). We discuss possible extensions in the concluding section.

In the context of pure game theory, it is possible to discern between (1) the degree of information each player has on the state of the system (i.e. perfect or non-perfect information games) and (2) the relationship between players' payoffs (i.e. cooperative or non-cooperative games). It is important to note that neither of these two features implies the other. The problem studied here, although not formally a game, could be classified as a perfect information non-cooperative game. That is, excluding the trivial case where the evader's final destination is a subset of pursuer's target $\mathcal{T}$, an increase in the pursuer's payoff necessarily corresponds to a decrease in the evader's payoff.

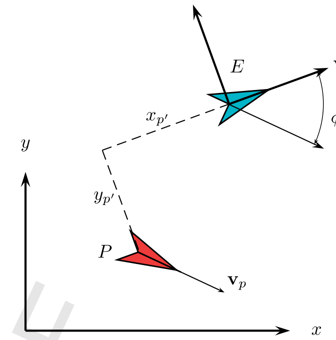


Fig. 5. Relative frame of reference centered in the evader and pursuer's location $(x_{p'}, y_{p'})$ with respect to it.

3. Trajectory modification design

In the previous section we derived the equations describing the dynamics of a system composed by of a pair of vehicles, coupled by a unilateral collision avoidance algorithm. In this section we describe how inputs v_p and ω_p can be computed, in order to steer a component of the state, namely $\{q_e^T, \phi_e\}^T$, from its original trajectory towards a given point $\{q_T^T, \phi_T\}^T \in \mathcal{T}$.

As described in section 2.2, when $\nu = 1$ the evader switches its original heading ϕ_0 towards one boundary of the collision set, namely $\hat{\beta}$. This suggests the idea that the evader's heading can be strategically influenced by producing an appropriate collision set B_{ep}. In this section we present a technique that can be used to produce desired $\hat{\beta}$ and consequently steer the evader towards $\mathcal{T}$. Given the switching behavior of the model described in (16) we focus our attention on the two states $\nu \in \{0, 1\}$ separately.

3.1. Switching guard conditions

Before introducing the equations relative to each state, we proceed first to define under what conditions the system switches from $\nu = 0$ to $\nu = 1$. In order to do so, let us introduce the evader's reference frame centered in q_e, having x-axis oriented as ϕ_e; we define $q_{p'} = \{x_{p'}, y_{p'}\}^T$ and $\phi_{p'}$ as the pursuer's state in this new system (Fig. 5).

The system switches to the collision state $\nu = 1$ when the condition $\phi_0 \in B_{ep}$ is satisfied; thus, we are interested in finding the set of pursuer's states for which the switching condition is met. Without loss of generality, we assume $\hat{\beta} = \beta_f$; what follows can easily be extended to the case $\hat{\beta} = \beta_r$.

In the new frame of reference, it is possible to rewrite our collision avoidance constitutive equations (3) and (10) as:

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

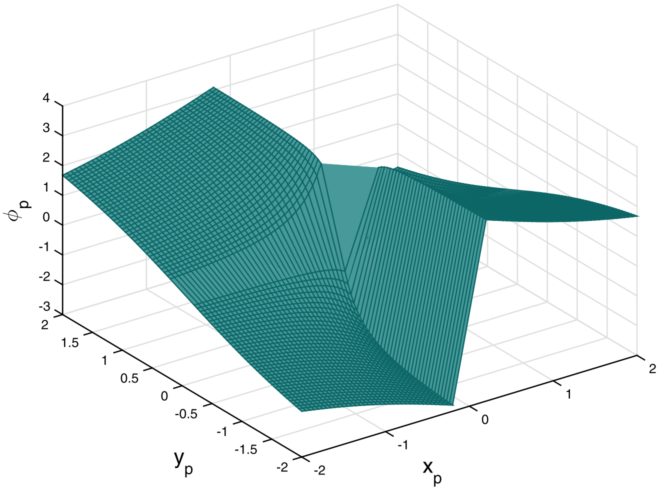


Fig. 6. Set $\mathcal{C}$ of points belonging to the switching guard.

$$\arctan\left(\frac{y_{p'}}{x_{p'}}\right) + \arcsin\left(\frac{R_{pz}}{\|q_{p'}\|}\right) = \lambda' \tag{17}$$

$$\tan\lambda'\left(v_e - v_p\cos\phi_{p'}\right) + v_p\sin\phi_{p'} = 0. \tag{18}$$

Assuming equal linear velocities $v_e = v_p$, equation (18) simplifies to:

$$\tan\lambda' = \frac{\sin\phi_{p'}}{\cos\phi_{p'} - 1}. \tag{19}$$

Substituting (19) into (17) yields:

$$\tan\left(\arctan\left(\frac{y_{p'}}{x_{p'}}\right) + \arcsin\left(\frac{R_{pz}}{\|q_{p'}\|}\right)\right) = \frac{\sin\phi_{p'}}{\cos\phi_{p'} - 1}. \tag{20}$$

Every pose $\{q_{p'}^T, \phi_{p'}\}^T$ that satisfies equation (20) represents a point in the evader's relative frame, for which the switching between collision and collision-free states occurs. We denote the set of points satisfying equation (20) as *Switching Surface* $\mathcal{C}$:

$$\mathcal{C} = \{\{q_{p'}^T, \phi_{p'}\}^T \in \mathbb{R}^2 \times \mathbb{S} \mid \phi_e = \hat{\beta}\}. \tag{21}$$

In Fig. 6 $\mathcal{C}$ is represented for an arbitrary hyper-rectangle $[x_{min}, x_{max}] \times [y_{min}, y_{max}] \times [\phi_{min}, \phi_{max}]$.

Finally, poses in the evader's frame can easily be converted to global coordinates using the following transformation:

$$\begin{bmatrix} x_p \\ y_p \\ \phi_p \end{bmatrix} = \begin{bmatrix} \cos\phi_e & -\sin\phi_e & 0 \\ \sin\phi_e & \cos\phi_e & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{p'} \\ y_{p'} \\ \phi_{p'} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \phi_e \end{bmatrix}. \tag{22}$$

Agents' configurations satisfying equation (20) represent switching conditions between the collision and collision-free states of the system. Hence, when the relative state $\{q_{p'}^T, \phi_{p'}\}^T \in \mathcal{C}$, it is possible to consider the system switching from $\nu = 0$ to $\nu = 1$.

3.2. Collision and collision-free state

Assuming the relative state $\{q_{p'}^T, \phi_{p'}\}^T \in \mathcal{C}$ and considering $\nu = 1$, the set of equations represented in (16) reduces to:

$$\begin{aligned}
&\dot{x}_e = v_e\cos\phi_e && \dot{x}_p = v_p\cos\phi_p \\
&\dot{y}_e = v_e\sin\phi_e && \dot{y}_p = v_p\sin\phi_p \\
&\dot{\phi}_e = \omega_e(\phi_e - \hat{\beta}) && \dot{\phi}_p = \omega_p \\
&v_e\left(\tan\lambda\cos\hat{\beta} - \sin\hat{\beta}\right) + v_p\left(\tan\lambda\cos\phi_p + \sin\phi_p\right) = 0 \\
&\lambda - \arctan\left(\frac{y_p - y_e}{x_p - x_e}\right) \mp \arcsin\left(\frac{R_{pz}}{\|q_e - q_p\|}\right) = 0.
\end{aligned} \tag{23}$$

Once $\nu = 1$, our goal is to define the pursuer's control inputs, namely v_p and ω_p, such that the sub-component $\{q_e^T, \phi_e\}^T$ of the system is driven towards a point in the target set $\{q_T^T, \phi_T\}^T \in \mathcal{T}$.

Before describing a solution for the control problem just introduced, we introduce the modeling equations for the system corresponding to the collision free state $\nu = 0$:

$$\begin{aligned}
&\dot{x}_e = v_e\cos\phi_e && \dot{x}_p = v_p\cos\phi_p \\
&\dot{y}_e = v_e\sin\phi_e && \dot{y}_p = v_p\sin\phi_p \\
&\dot{\phi}_e = \omega_e(\phi_0 - \phi_e) && \dot{\phi}_p = \omega_p.
\end{aligned} \tag{24}$$

3.3. Model predictive control formulation

In this section we describe the model predictive control (MPC) used by the pursuer to steer the evader towards the desired capture set. In MPC ([36]), a dynamic programming approach is used to minimize a cost function over a given finite number of steps N. The purpose of the optimization problem is to find the pursuer's control inputs $u_p = \{v_p, \omega_p\}^T$ such that the distance from the evader's state and the target $\{q_T^T, \phi_T\}^T$ is minimized over time. In order to do so, the complete strategy will be composed by two sequential maneuvers, which we will study separately.

Definition 3. (Approaching maneuver): Assuming the system in $\nu = 0$ at time t_0, find u_p such that $\{q_{p'}^T, \phi_{p'}\}^T \in \mathcal{C}$ for some time $t_s < \infty$.

Definition 4. (Forcing maneuver): Assuming the system in $\nu = 1$ at time t_s, find u_p such that $\{q_e^T, \phi_e\}^T \in \mathcal{T}$, while maintaining $\nu = 1$ at all times.

3.3.1. Approaching maneuver predictive model

We introduce a discrete time domain indexed by k and let Δt be the corresponding constant time step. By definition, during the *approaching maneuver* a relative state between pursuer and evader must be reached. For this reason, we introduce a two-vehicle finite difference model, where the pursuer's dynamics are expressed in the evader's reference frame:

$$\begin{aligned}
x_{p'}^{k+1} &= x_{p'}^k + (\omega_p y_{p'}^k - v_p + v_e\cos\phi_{p'}^k)\Delta t \\
y_{p'}^{k+1} &= y_{p'}^k - (\omega_p x_{p'}^k - v_e\sin\phi_{p'}^k)\Delta t \\
\phi_{p'}^{k+1} &= \phi_{p'}^k + (\omega_e - \omega_p)\Delta t.
\end{aligned} \tag{25}$$

Letting $\{\tilde{x}_{p'}, \tilde{y}_{p'}, \tilde{\phi}_{p'}\}^T$ be the target relative state for the approaching maneuver, we can define the measure y_a,

$$y_a = \begin{bmatrix} \tilde{x}_{p'} \\ \tilde{y}_{p'} \\ \tilde{\phi}_{p'} \end{bmatrix} - \begin{bmatrix} x_{p'} \\ y_{p'} \\ \phi_{p'} \end{bmatrix},$$

and the following quadratic cost function,

$$c_a(y_a, u_p) = y_a^T R y_a + \Delta u_p^T Q \Delta u_p, \tag{26}$$

where weight control matrices $R \in \mathbb{R}^{3\times3}$, $Q \in \mathbb{R}^{2\times2}$ scale the different variables' magnitude. The resulting nonlinear MPC problem is:

$$\begin{aligned}
\min_{y_a, u_p} \quad & \sum_{i=1}^{N} c_a^i(y_a^i, u_p^i) \\
\text{s. t.} \quad & x_{p'}^{i+1} = x_{p'}^i + (\omega_p^i y_{p'}^i - v_p^i + v_e\cos\phi_{p'}^i)\Delta t \\
& y_{p'}^{i+1} = y_{p'}^i - (\omega_p^i x_{p'}^i - v_e\sin\phi_{p'}^i)\Delta t \\
& \phi_{p'}^{i+1} = \phi_{p'}^i + (\omega_e - \omega_p^i)\Delta t \\
& u_{min} \le u_i \le u_{max}
\end{aligned} \tag{27}$$

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

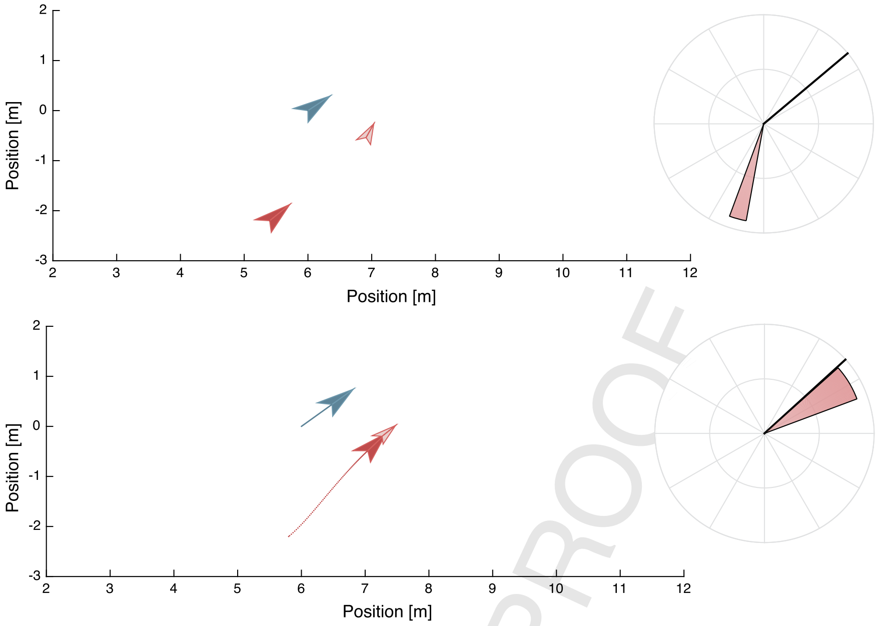


Fig. 7. Approaching maneuver configurations at initial and final time.

It is worth noting that, as common practice in MPC problems, the presence of the term Δu_p in the cost function has been introduced in order to bound possible oscillating behaviors on the control input.

3.3.2. *Forcing maneuver predictive model*

Let us now study the system for $\nu = 1$. In this case, since the final target is defined in global coordinates, the use of the relative dynamic model (25) does not introduce any simplifications. For this reason, the complete set of equations/constrains in (23) is used. However, given the additional non-linearities introduced by the algebraic constraints, we transform the continuous time DAE model into a discrete linear time varying (LTV) model.

In order to do so, we first rearrange state and control input as:

$$x = \{x_e, y_e, x_p, y_p, \phi_p\}^T \qquad u = \{\lambda, \phi_e, v_p, \phi_p\}^T$$

from which the complete model in (23) can be expressed as:

$$\begin{aligned} &\dot{x} = f(x, u) \\ &g(x, u) = 0. \end{aligned}$$

A linear prediction model is obtained using a first order Taylor approximation of the complete DAE set around the current state of the system $\{x_0^T, u_0^T\}$, i.e.:

$$\begin{aligned} &\dot{x}^k = f(x_0, u_0) + f_{x|x_0}(x^k - x_0) + f_{u|u_0}(u^k - u_0) + \mathcal{O}^2 \\ &g(x^{k+1}, u^{k+1}) = g_{x|x_0}(x^{k+1} - x^k) + g_{u|u_0}(u^{k+1} - u^k) + \mathcal{O}^2 = 0, \end{aligned}$$

where $g(x_0, u_0) = 0$ for consistency with the constraints. Neglecting higher order terms, we rewrite the approximate model as:

$$\begin{aligned} &\dot{x}^k = \tilde{A}_0 + \tilde{A}(x^k - x_0) + \tilde{B}(u^k - u_0) \\ &E\,(u^{k+1} - u^k) + F\,(x^{k+1} - x^k) = 0, \end{aligned} \tag{28}$$

where $\tilde{A}_0 \in \mathbb{R}^{5\times1}$,$\tilde{A} \in \mathbb{R}^{5\times5}$,$\tilde{B} \in \mathbb{R}^{5\times4}$, $F \in \mathbb{R}^{2\times5}$ and $E \in \mathbb{R}^{2\times4}$ are defined as:

$$\begin{aligned} &\tilde{A}_0 = \left[\, f(x_0, u_0) \,\right] \quad \tilde{A} = \left[\, f_{x|x_0} \,\right] \quad \tilde{B} = \left[\, f_{u|u_0} \,\right] \\ &E^k = \left[\, g_{x|x^k} \,\right] \qquad F^k = \left[\, g_{u|u^k} \,\right]. \end{aligned} \tag{29}$$

The complete expression for the terms in (29) is reported in the Appendix. Time derivatives in (28) can be discretized using a forward Euler approximation:

$$x^{k+1} = (I - \Delta t\tilde{A})x^k + \Delta t\tilde{B}u^k + \Delta t(\tilde{A}_0 - \tilde{A}x_0 - \tilde{B}u_0),$$

and rearranging,

$$\begin{aligned} &x^{k+1} = Ax^k + Bu^k + A_0 \\ &E\,(u^{k+1} - u^k) + F\,(x^{k+1} - x^k) = 0. \end{aligned} \tag{30}$$

The set of parameters $\{A, B, A_0, E, F\}$ contained in the LTV model (30) are updated at each iteration. Given the sparsity structure of the resulting model, such a task can be efficiently carried out numerically. Also, the possibility of writing adaptive models, where the problem is initialized only once, considerably reduces the computational burden. Finally, we introduce the following measure for the forcing maneuver y_f:

$$y_f = C\,x + D\,u - \begin{bmatrix} q_T \\ \phi_T \end{bmatrix}, \tag{31}$$

where appropriate matrix $C \in \mathbb{R}^{3\times5}$ and $D \in \mathbb{R}^{3\times4}$ select the variables in x and u to be steered towards the target set. From the cost function

$$c_f(y_f, u) = y_f^T\,R\,y_f + \Delta u^T\,Q\,\Delta u,$$

the final MPC relative to the forcing maneuver is:

$$\begin{aligned} \min_{y_f, u} \quad & \sum_{i=1}^{N} c_f^i(y_f^i, u^i) \\ \text{s. t.} \quad & x^{i+1} = Ax^i + Bu^i + A_0 \\ & E\,(u^{i+1} - u^i) + F\,(x^{i+1} - x^i) = 0 \\ & u_{min} \le u_i \le u_{max}. \end{aligned}$$

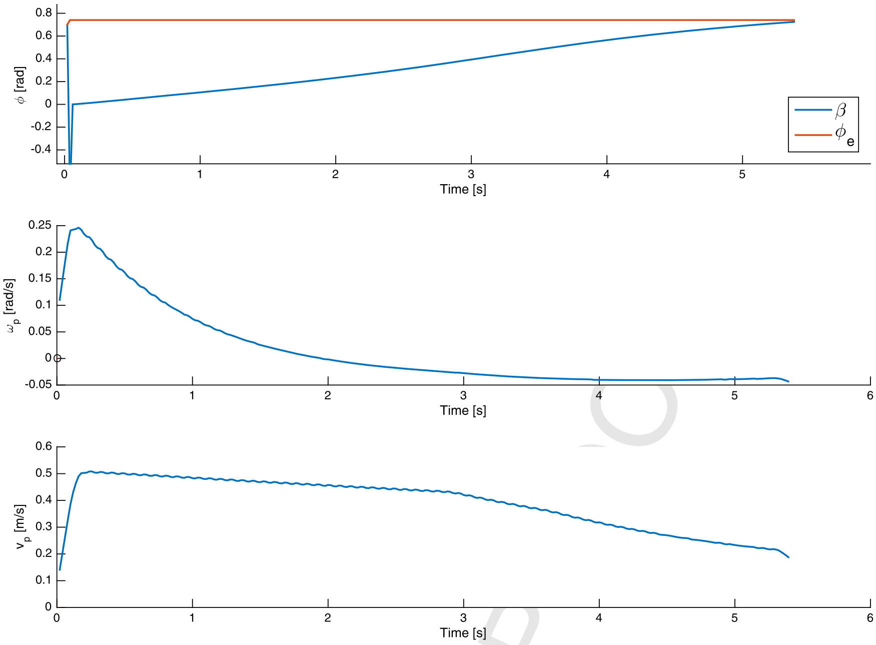


Fig. 8. Approaching maneuver results. In the top figure we can see the value of $\hat{\beta}$ approaching the evader's current heading. The bottom two figures represent the pursuer inputs associated with the approaching maneuver. (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

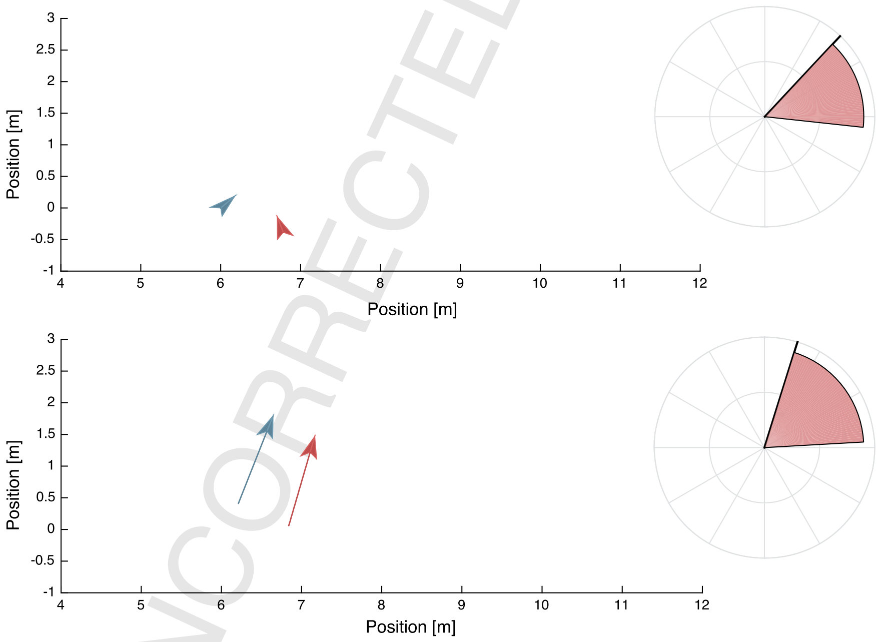


Fig. 9. Forcing maneuver at initial and final time.

4. Results

4.1. Numerical simulations

Results presented in this section were conducted on MATLAB 2015a running on a 3.2 GHz, 16 Gb iOS desktop. MPC problems were formulated using YALMIP [37] and solved using GUROBI solver.

As stated in Section 2, complete knowledge of neighbors' positions and velocities is assumed. In the context of the results presented here, we assume this information to be obtained by mean of a sharing protocol (i.e. ADS-B). In the first example, we provide results from the approaching maneuver. In Fig. 7, configurations for both evader and pursuer are represented at initial and switching times. In Fig. 8, we observe values of $\hat{\beta}$ (in blue) approaching the evader's heading ϕ_e (red) until the switching condition is reached.

Once the approaching maneuver is completed and the system's state belongs to the switching guard, the forcing maneuver can be initiated. We present results for a forcing maneuver whose objective is to steer evader's heading towards a new desired one; therefore, $\mathcal{T} \subset \mathbb{S}$. As shown in Fig. 9, the evader's heading moves from the initial value, $\phi_e = 0.7$ rad towards the target heading $\phi_T = 1.25$ rad. In Fig. 10, pursuer's control inputs for the maneuver are reported. Note that the new heading ϕ_T can be maintained as the pursuer continuously adjusts its input.

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

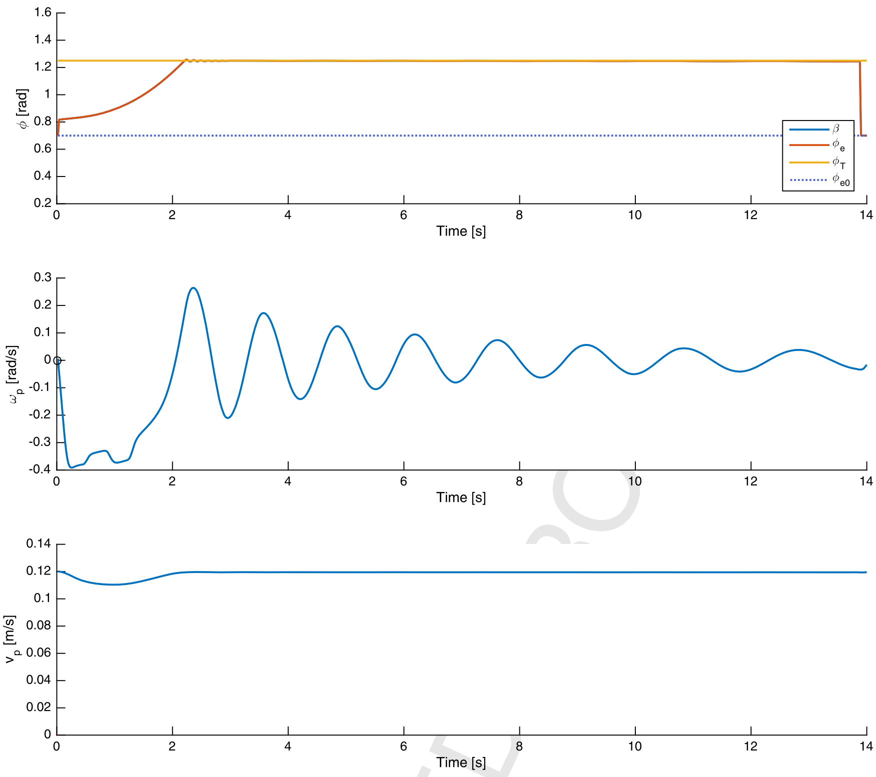


Fig. 10. Forcing maneuver results. In the top figure we can see the evader's heading ϕ_e been forced to the desired value ϕ_T and then maintained over time. Bottom two figures represent pursuer control input required for the maneuver.

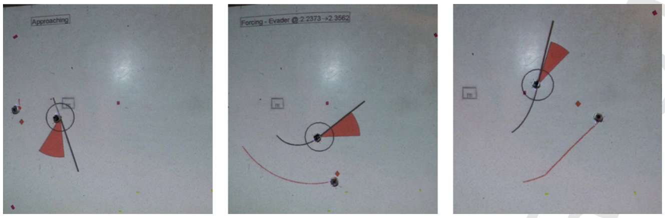


Fig. 11. Results from robot experiments. Three stages from realization of approaching and forcing maneuvers. (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

4.2. Robot experiments

Experiments conducted with real ground robots are presented here. Given the modeling assumptions introduced in section 2, that is, planar motion and unicycle kinematics, the complete dynamics of the problem can be fully represented by differential drive robots. To this end, it is possible to transform unicycle linear and angular velocity into wheel velocities required by the differential drive robots. In Fig. 11 we present screen-shots from a one-evader/one-pursuer scenario – snapshots are taken from a camera mounted on the ceiling of the room. Graphic elements on the floor are real-time simulation data obtained projecting a plot window on the ground. The reference system used in the experiments has x-axis pointing down and y-axis pointing right. The robot initially in the right position is the evader, whose protected zone is defined by the black circle around it. The red sector represents the obstacle velocities associated with the pursuer, initially located behind the evader. The evader goal is located straight respect to its initial heading, at a virtual distance of 300 meters. The pursuer aims to impose a new heading on the evader equal to $3/4\pi$ rad. In accordance with the numerical simulations, the pursuer employs its strategy by means of two sequential maneuvers. First, it reaches the switching positions $\{x_{p'}, y_{p'}\}$ (red diamond in figures); this control problem is solved using the online MPC problem presented earlier. Successively, the pursuer starts a predefined circular maneuver at constant turning radius. In this case, the input parameters $\{u_p, \omega_p\}$ were computed offline and transmitted real time to the robot. As shown, the evader initially changes its heading (marked by a black line) as a consequence of the expected collision. After that, the pursuer maintains its final direction, preventing the evader from heading back to the original goal.

5. Conclusion

The increasing demand for autonomous aerial systems necessitates strategies and design considerations to ensure safe navigation and trust-worthiness of the craft and its algorithms. In this paper, a novel concept of vulnerability for autonomous vehicles was introduced; this new class should be considered along with the existing set of vulnerabilities, composed by *interceptions of transmission*, *jamming* and *injection of message*.

The model described in this work shows that when predictable CDR algorithms are employed, avoidance maneuvers can be anticipated and exploited. This work intends to address a first understating of the introduced vulnerability, in order to proceed with the development of countermeasures and mitigating solutions. To this end, a possible direction of research could be the effect of a stochastic component in the collision avoidance maneuver.

Possible future developments will address the definition of a single optimization problem instead of the two stages strategy described here. A mixed integer programming formulation of the problem could serve the purpose. In addition, future developments include the partial relaxation of modeling assumptions introduced, i.e. extension to a three dimensional domain, evading maneuvers with varying speed and the introduction of noisy measurements.

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

Conflict of interest statement

None declared.

Appendix. First order Taylor approximation

We derive here the expressions for the first order Taylor approximation relative to the system dynamics and constraints introduced by the collision avoidance. Considering the two unicycle dynamics contained in (28) we have:

$$f(x_0,u_0)=\begin{bmatrix} v_e\cos\phi_e \\ v_e\sin\phi_e \\ v_p\cos\phi_p \\ v_p\sin\phi_p \\ \omega_p \end{bmatrix} \qquad f_{x|x_0}=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -v_p\sin\phi_p \\ 0 & 0 & 0 & 0 & v_p\cos\phi_p \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$f_{u|u_0}=\begin{bmatrix} 0 & -v_e\sin\phi_e & 0 & 0 & 0 \\ 0 & v_e\cos\phi_e & 0 & 0 & 0 \\ 0 & 0 & \cos\phi_p & 0 & 0 \\ 0 & 0 & \sin\phi_p & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{32}$$

The two constraints in (28) are reported and expanded here for convenience:

$$g_1^{k+1}:\tan\lambda^{k+1}\left(v_e\cos\phi_e^{k+1}-v_p^{k+1}\cos\phi_e^{k+1}\right) - v_e\sin\phi_e^{k+1}+v_p^{k+1}\sin\phi_p^{k+1}=0 \tag{33}$$

$$g_2^{k+1}:\lambda^{k+1}-\arctan\left(\frac{y_p^{k+1}-y_e^{k+1}}{x_p^{k+1}-x_e^{k+1}}\right) - \arcsin\left(\frac{R_{pz}}{\|q_p^{k+1}-q_e^{k+1}\|_2}\right)=0. \tag{34}$$

Equations (33) and (34) are now derived with respect to their variables, namely λ, v_p, ϕ_p and ϕ_e. Considering first constraint (33), we can derive:

$$\begin{aligned} g_{1,\lambda} &= \sec^2\lambda(v_e\cos\phi_e - v_p\cos\phi_p) \\ g_{1,v_p} &= -\tan\lambda\cos\phi_p+\sin\phi_p \\ g_{1,\phi_p} &= v_p\tan\lambda\sin\phi_p+v_p\cos\phi_p \\ g_{1,\phi_e} &= -v_e\tan\lambda\sin\phi_e-\cos\phi_e. \end{aligned}$$

Considering the constraint (34), we first need to make the dependency from the control inputs explicit. In order to do so, we introduce the discrete dynamics of the system into the constraints, that is:

$$0=\lambda^{k+1}-\arctan\left(\frac{\delta_y^{k+1}}{\delta_x^{k+1}}\right)-\arcsin\left(\frac{R_{pz}}{\sqrt{(\delta_x^{k+1})^2+(\delta_y^{k+1})^2}}\right), \tag{35}$$

where

$$\begin{aligned} \delta_x^{k+1} &= x_p^k+v_p^k\cos\phi_p^k-x_e^k-v_e\cos\phi_e^k\Delta t \\ \delta_y^{k+1} &= y_p^k+v_p^k\sin\phi_p^k-y_e^k-v_e\sin\phi_e^k\Delta t. \end{aligned}$$

Computing the derivatives of g_2 with respect to its arguments leads to:

$$g_{2,\lambda}=1$$

$$g_{2,v_p}=-\frac{1}{1+\frac{\delta_y^2}{\delta_x^2}}\left(\frac{\sin\phi_p\Delta t\,\delta_x-\cos\phi_p\Delta t\,\delta_y}{\delta_x^2}\right) - \frac{1}{\sqrt{1-\xi^2}}\left(-\frac{R_{pz}K_1}{\delta_x^2+\delta_y^2}\right)$$

$$g_{2,\phi_p}=-\frac{1}{1+\frac{\delta_y^2}{\delta_x^2}}\left(\frac{v_p\cos\phi_p\Delta t\,\delta_x+v_p\sin\phi_p\Delta t\,\delta_y}{\delta_x^2}\right) - \frac{1}{\sqrt{1-\xi^2}}\left(-\frac{R_{pz}K_2}{\delta_x^2+\delta_y^2}\right)$$

$$g_{2,\phi_e}=-\frac{1}{1+\frac{\delta_y^2}{\delta_x^2}}\left(\frac{-v_e\cos\phi_e\Delta t\,\delta_x-v_e\sin\phi_e\Delta t\,\delta_y}{\delta_x^2}\right) - \frac{1}{\sqrt{1-\xi^2}}\left(-\frac{R_{pz}K_3}{\delta_x^2+\delta_y^2}\right),$$

where the following quantities were defined to be

$$\xi=\frac{R_{pz}}{\sqrt{\delta_x^2+\delta_y^2}}$$

$$K_1=\frac{\partial\sqrt{\delta_x^2+\delta_y^2}}{\partial v_p}= \frac{1}{2}\left(\delta_x^2+\delta_y^2\right)^{-\frac{1}{2}}\left(2\delta_x\cos\phi_p\Delta t+2\delta_y\sin\phi_p\Delta t\right)$$

$$K_2=\frac{\partial\sqrt{\delta_x^2+\delta_y^2}}{\partial \phi_p}= \frac{1}{2}\left(\delta_x^2+\delta_y^2\right)^{-\frac{1}{2}}\left(-2\delta_x v_p\sin\phi_p\Delta t+2\delta_y v_p\cos\phi_p\Delta t\right)$$

$$K_3=\frac{\partial\sqrt{\delta_x^2+\delta_y^2}}{\partial \phi_e}= \frac{1}{2}\left(\delta_x^2+\delta_y^2\right)^{-\frac{1}{2}}\left(2\delta_x v_e\sin\phi_p\Delta t-2\delta_y v_e\cos\phi_p\Delta t\right).$$

These quantities can be written in the compact form:

$$E=\begin{bmatrix} g_{1,\lambda} & g_{1,\phi_e} & g_{1,v_p} & 0 \\ g_{2,\lambda} & g_{2,\phi_e} & g_{2,v_p} & 0 \end{bmatrix} \qquad F=\begin{bmatrix} 0 & 0 & 0 & 0 & g_{1,\phi_p} \\ 0 & 0 & 0 & 0 & g_{2,\phi_p} \end{bmatrix}. \tag{36}$$

References

[1] P.M. Dames, M. Schwager, D. Rus, V. Kumar, Active magnetic anomaly detection using multiple micro aerial vehicles, IEEE Robot. Autom. Lett. 1 (1) (2016) 153–160.
[2] M. Fumagalli, R. Naldi, A. Macchelli, F. Forte, A.Q. Keemink, S. Stramigioli, R. Carloni, L. Marconi, Developing an aerial manipulator prototype: physical interaction with the environment, IEEE Robot. Autom. Mag. 21 (3) (2014) 41–50.
[3] Q. Lindsey, D. Mellinger, V. Kumar, Construction with quadrotor teams, Auton. Robots 33 (3) (2012) 323–336.
[4] P. Drouilhet, G. Knittel, V. Orlando, Automatic dependent surveillance air navigation system, US Patent 5,570,095, https://www.google.com/patents/US5570095, 29 Oct. 1996.
[5] P. Park, C. Tomlin, Performance evaluation and optimization of communication infrastructure for the next generation air transportation system, IEEE Trans. Parallel Distrib. Syst. 26 (4) (2015) 1106–1116.

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008

[6] D. McCallie, J. Butts, R. Mills, Security analysis of the ADS-B implementation in the next generation air transportation system, Int. J. Crit. Infrastruct. Prot. 4 (2) (2011) 78–87.

[7] J.V. Carroll, Vulnerability assessment of the US transportation infrastructure that relies on the global positioning system, J. Navig. 56 (2) (2003) 185–193.

[8] T. Humphreys, Statement on the Vulnerability of Civil Unmanned Aerial Vehicles and Other Systems to Civil GPS Spoofing, University of Texas at Austin, 18 July 2012.

[9] C.J. Giannatto, G. Markowsky, Potential vulnerabilities of the nextgen air traffic control system, in: World Congress in Computer Science, Computer Engineering and Applied Computing, Las Vegas, 2014.

[10] P. Park, H. Khadilkar, H. Balakrishnan, C.J. Tomlin, High confidence networked control for next generation air transportation systems, IEEE Trans. Autom. Control 59 (12) (2014) 3357–3372.

[11] N. Ghose, L. Lazos, Verifying ADS-B navigation information through Doppler shift measurements, in: Digital Avionics Systems Conference (DASC), 2015 IEEE/AIAA 34th, IEEE, 2015, pp. 4A2–1.

[12] M. Monteiro, A. Barreto, T. Kacem, J. Carvalho, D. Wijesekera, P. Costa, Detecting malicious ADS-B broadcasts using wide area multilateration, in: Digital Avionics Systems Conference (DASC), 2015 IEEE/AIAA 34th, IEEE, 2015, pp. 4A3–1.

[13] L. Pallottino, E.M. Feron, A. Bicchi, Conflict resolution problems for air traffic management systems solved with mixed integer programming, IEEE Trans. Intell. Transp. Syst. 3 (1) (2002) 3–11.

[14] A. Kushleyev, D. Mellinger, C. Powers, V. Kumar, Towards a swarm of agile micro quadrotors, Auton. Robots 35 (4) (2013) 287–300.

[15] P. Panyakeow, M. Mesbahi, Deconfliction algorithms for a pair of constant speed unmanned aerial vehicles, IEEE Trans. Aerosp. Electron. Syst. 50 (1) (2014) 456–476.

[16] Y. Lin, S. Saripalli, Path planning using 3D dubins curve for unmanned aerial vehicles, in: Unmanned Aircraft Systems (ICUAS), 2014 International Conference on, IEEE, 2014, pp. 296–304.

[17] A.D. Ames, J.W. Grizzle, P. Tabuada, Control barrier function based quadratic programs with application to adaptive cruise control, in: Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on, IEEE, 2014, pp. 6271–6278.

[18] U. Borrmann, L. Wang, A.D. Ames, M. Egerstedt, Control barrier certificates for safe swarm behavior, IFAC-PapersOnLine 48 (27) (2015) 68–73.

[19] L. Wang, A.D. Ames, M. Egerstedt, Safe certificate-based maneuvers for teams of quadrotors using differential flatness, in: IEEE International Conference on Robotics and Automation, 2017, pp. 3293–3298.

[20] C. Tomlin, G.J. Pappas, S. Sastry, Conflict resolution for air traffic management: a study in multiagent hybrid systems, IEEE Trans. Autom. Control 43 (4) (1998) 509–521.

[21] I.M. Mitchell, A.M. Bayen, C.J. Tomlin, A time-dependent Hamilton–Jacobi formulation of reachable sets for continuous dynamic games, IEEE Trans. Autom. Control 50 (7) (2005) 947–957.

[22] B. Albaker, N. Rahim, A survey of collision avoidance approaches for unmanned aerial vehicles, in: Technical Postgraduates (TECHPOS), 2009 International Conference for, IEEE, 2009, pp. 1–7.

[23] P. Fiorini, Z. Shiller, Motion planning in dynamic environments using velocity obstacles, Int. J. Robot. Res. 17 (7) (1998) 760–772.

[24] A. Chakravarthy, D. Ghose, Obstacle avoidance in a dynamic environment: a collision cone approach, IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum. 28 (5) (1998) 562–574.

[25] B. Damas, J.S. Victor, Avoiding moving obstacles: the forbidden velocity map, in: Intelligent Robots and Systems, 2009, IEEE/RSJ International Conference on, IROS 2009, IEEE, 2009, pp. 4393–4398.

[26] J. Van den Berg, M. Lin, D. Manocha, Reciprocal velocity obstacles for real-time multi-agent navigation, in: Robotics and Automation, 2008, IEEE International Conference on, ICRA 2008, IEEE, 2008, pp. 1928–1935.

[27] J. Snape, J. van den Berg, S.J. Guy, D. Manocha, The hybrid reciprocal velocity obstacle, IEEE Trans. Robot. 27 (4) (2011) 696–706.

[28] Y. Kuwata, M.T. Wolf, D. Zarzhitsky, T.L. Huntsberger, Safe maritime navigation with COLREGS using velocity obstacles, in: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, IEEE, 2011, pp. 4728–4734.

[29] Y.I. Jenie, E.-J. Van Kampen, C.C. de Visser, Q.-P. Chu, Velocity obstacle method for non-cooperative autonomous collision avoidance system for UAVs, in: AIAA Guidance, Navigation, and Control Conf., 2014.

[30] E. Lalish, K. Morgansen, et al., Decentralized reactive collision avoidance for multivehicle systems, in: Decision and Control, 2008, 47th IEEE Conference on, CDC 2008, IEEE, 2008, pp. 1218–1224.

[31] A. Levy, C. Keitel, S. Engel, J. McLurkin, The extended velocity obstacle and applying ORCA in the real world, in: Robotics and Automation (ICRA), 2015 IEEE International Conference on, IEEE, 2015, pp. 16–22.

[32] Z. Shiller, R. Prasanna, J. Salinger, A Unified Approach to Forward and Lane-Change Collision Warning for Driver Assistance and Situational Awareness, Tech. Rep., SAE Technical Paper, 2008.

[33] P. Pierpaoli, M. Egerstedt, A. Rahmani, Altering UAV flight path by threatening collision, in: Proc. Digital Avionics Systems Conference, Prague, Czech Republic, 2015.

[34] P. Pierpaoli, R. Zanforlin, A. Rahmani, Vulnerability of UAV sense and avoid to exploitations: non-cooperative trajectory modifications, in: AIAA Guidance, Navigation and Control Conference, Kissimmee, Florida, 2015.

[35] M.J. Osborne, A. Rubinstein, A Course in Game Theory, MIT Press, 1994.

[36] J.B. Rawlings, D.Q. Mayne, Model Predictive Control: Theory and Design, Nob Hill Pub., 2009.

[37] J. Lofberg, Yalmip: a toolbox for modeling and optimization in Matlab, in: Computer Aided Control Systems Design, 2004 IEEE International Symposium on, IEEE, 2005, pp. 284–289.

Please cite this article in press as: P. Pierpaoli, A. Rahmani, UAV collision avoidance exploitation for noncooperative trajectory modification, Aerosp. Sci. Technol. (2017), https://doi.org/10.1016/j.ast.2017.12.008