34

Autonomous Guidance of UAVs for Real-Time Target Tracking in Adversarial Environments

Ugur Zengin and Atilla Dogan
University of Texas at Arlington
USA

1. Introduction

While recent technological advances have enabled the development of unmanned vehicular systems and recent implementations have proven their benefits in both military and civilian applications, the full benefit of unmanned systems will be utilized when they can operate autonomously. The primary requirements of autonomy are the capabilities of detecting internal and external changes, and of reacting to them without human intervention in a safe and efficient manner. This can be achieved by developing and implementing autonomous guidance and control systems (AGCS) to "pilot" unmanned vehicles (Rathbun et al., 2002; Finke et al., 2003; Flint et al., 2002; Jun et al., 2002; Pongpunwattana & Rysdyk, 2004; Nikolas et al., 2003; Zhu et al., 2005; Waydo & Murray,2003). Tracking highly mobile targets is a type of mission that can significantly benefit from the use of UAVs (Unmanned Aerial Vehicles) with the capability of autonomy, especially when the pursuit is to take place in an environment where various sources of "threat", obstacles and restricted areas may exist. In a military scenario, multiple UAVs can be used to track enemy or escort a friendly convoy while avoiding no-fly zones and possible sites of SAMs (Surface-to-Air Missiles). In a border patrol application, UAVs can be employed to track intruders while staying within the border and avoiding high elevation. In a law-enforcement scenario, criminals might be pursued or a specific vehicle might be tracked for protection while avoiding high buildings or residential areas. As a wild-life protection effort, animals can be tracked while avoiding high elevation.

When threat exposure, obstacle and/or restricted region are not among the concerns of a tracking problem, UAV trajectories are commanded to fly directly over the moving target (Sengupta & Hedrick, 2003; Spry et al., 2005). When the target is evasive in an intelligent manner, the tracking problem is the subject of pursuit-evasion game theory (Jang & Tomlin, 2005; Antoniades et al., 2003; Vidal et al., 2002; Hespanha et al.,2000; Hespanha et al.,1999). Ground target tracking and required sensors are also particularly studied (Sengupta & Hedrick, 2003; Schumacher 2005; Sinha et al.,2004; Shea, 2000; Koch & Klem, 2001). For example, in (Sengupta & Hedrick, 2003), the tracking is performed by utilizing an offset vector. There are various new challenges when tracking needs to be done by mobile sensors in an area where there exist various threats, obstacles and/or restricted areas as well as other vehicles to avoid. There might be various and completely different types of "threats", obstacles and restricted areas, and the information regarding their presence and/or level

might contain some uncertainty (Jun & D`Andrea, 2003; Hespanha et al.,2001). "Threat" does not necessarily mean locations or objects that have the potential of inflicting damage on the pursuing vehicles. An area or object on which the proximity of the pursuing vehicles may pose some undesired risk is also referred to as threat.

The previous work by the authors (Zengin & Dogan, 2004; Dogan & Zengin, 2006) developed a rule-based (Negnevitsky, 2002) guidance algorithm for a UAV to follow a dynamic ground target while minimizing threat exposure level, eliminating the risk of flying into obstacles and avoiding no-fly zones. This was done by introducing and utilizing the probabilistic threat exposure map (PTEM), which quantifies threats, obstacles and restricted regions in single framework. This chapter presents a new and significantly improved approach that has resulted in a more systematic and analytic formulation of the guidance strategy, a computationally more efficient algorithm. Specifically, the algorithm described in this chapter uses a gradient search approach in minimizing threat exposure and avoiding restricted areas and systematically utilizes mathematical tools such as level curves, inertial and moving-rotating frames and rotation matrices, vector representations of directions and geometric relations between cones, circles and straight lines. Furthermore, strategy states of this algorithm are better organized, which makes the algorithm much easier for reading and programming. As compared to the previous algorithms, a computationally more feasible guidance algorithm is developed without compromising the performance of tracking, avoiding obstacles/restricted-areas and minimizing threat exposure.

2. Formulation of Adversarial Environment

Adversarial environment is an environment where threat exposure should be minimized, obstacles and restricted areas should be avoided. In this chapter, "threat" is used as a broad term to describe the risk or cost for a UAV to occupy a given location at a given time as well as obstacles and restricted regions in the area of operation. When a UAV is flying in an area with multiple threats, safety of the flight is characterized by the probability of the UAV becoming disabled at a certain location, specified by it's x -- and y -- coordinates relative to a frame of reference, (x, y) at a certain time t. To be able to construct the problem in a probabilistic framework, several events are defined and their probabilities are determined.

2.1 Formulation of Area of Operation

Let $\mathrm{E_i}(x, y, t)$ be the event that the UAV becomes disabled by $\mathrm{i^{th}}$ source of threat at the position of (x, y) at time t in the area of operation. $\mathrm{E}(x, y, t)$ is the event that the UAV becomes disabled by at least one of the threat sources at the position (x, y) at time t. Then, let $\mathrm{f}_{p,i}(x, y)$ and $\mathrm{f}_{t,i}(x, y)$ be probability density functions (*pdf*) such that the probability of the UAV becoming disabled by i^{th} threat source at the neighbourhood of (x, y) at time t is

$$\mathrm{p}_i(x, y) = \mathrm{f}_{p,i}(x, y)\, \mathrm{f}_{t,i}(t)\, \Delta x\, \Delta y\, \Delta t \tag{1}$$

where Δx and Δy are to define the area of a neighbourhood of (x, y) and Δt is to define a neighbourhood of t. Note that, $\mathrm{f}_{p,i}(x, y)$ models the dependency of becoming disabled on

position and $f_{t,i}(t)$ models the dependency of becoming disabled on time. As an example, $f_{p,i}(x,y)$ can be characterized by a Gaussian *pdf*, which specifies the concentration point (location) of the threat by the mean value and the level of penalty of flying close to it by the variance. Regarding the time dependency of becoming disabled, various possible *pdf*s can be used for $f_{t,i}(t)$. For example, a uniform *pdf* for $f_{t,i}(t)$ means that the threat exposure level of the UAV at a given position does not depend on time itself but the amount of elapsed time in the neighbourhood of that position. If the level of exposure of the UAV to threats increases as it stays longer in the area of operation, then an increasing probability density function of time should be defined for $f_{t,i}(t)$.

Now, let $S(x,y,t)$ be a certain event that the UAV follows trajectory S to reach (x,y) at time t. Then, the conditional probability of the event that the UAV becomes disabled by i^{th} source of threat at the position of (x,y) at time t under the condition that the UAV follows trajectory S is defined as

$$p_{S,i}(x,y,t) = P[E_i(x,y,t) \mid S(x,y,t)] = \int_t f_{p,i}(x,y)\, f_{t,i}(t)\, l_1(x,y,t) l_2(x,y,t) dt \tag{2}$$

where l_1 and l_2 are used to define the neighbourhood at a point on trajectory S (e.g. radar signature area of the vehicle). Also note in (2) (x,y) are functions of time and thus $f_{p,i}$ is also a function of time.

To better explain the dependency of the conditional probability on both position and time, (2) will be presented in a special case where the UAV enters the area of operation at time t_0, moves until time t_1, afterwards stops and hovers at the same position. In this case,

$$\begin{aligned} p_{S,i}(x,y,t) &= \int_{t_0}^{t_1} f_{p,i}(x,y) f_{t,i}(t) l_1(x,y,t) l_2(x,y,t) dt \\ &+ f_{p,i}(x(t_1),y(t_1)) \int_{t_1}^{t} f_{t,i}(t) l_1(x(t_1),y(t_1),t) l_2(x(t_1),y(t_1),t) dt \end{aligned} \tag{3}$$

where $f_{p,i}$ is constant after time t_1 because the UAV does not change its position. If l_1 and l_2 do not explicitly depend on time, then

$$\begin{aligned} p_{S,i}(x,y,t) &= \int_{t_0}^{t_1} f_{p,i}(x,y) f_{t,i}(t) l_1(x,y,t) l_2(x,y,t) dt \\ &+ f_{p,i}(x(t_1),y(t_1)) l_1(x(t_1),y(t_1)) l_2(x(t_1),y(t_1)) \int_{t_1}^{t} f_{t,i}(t) dt \end{aligned} \tag{4}$$

Note that in the above equation, the second term shows how the probability increases even when the position of the UAV does not change. Let us assume that $f_{t,i}(t)$ is a uniform *pdf*, i.e. it is constant in the interval when it is not zero. Let us further assume that during the time the UAV stays in the area of operation, $f_{t,i}(t)$ is nonzero. Then,

$$\begin{aligned} p_{S,i}(x,y,t) &= f_{t,i}(t_0)\int_{t_0}^{t_1} f_{p,i}(x,y)l_1(x,y,t)l_2(x,y,t)dt \\ &+ f_{p,i}(x(t_1),y(t_1))l_1(x(t_1),y(t_1))l_2(x(t_1),y(t_1))f_{t,i}(t_0)(t-t_1) \end{aligned} \tag{5}$$

where note in the first term that $f_{p,i}(x,y)$ is still implicitly a function of time since the UAV moves until t_1.

If there are N number of sources of threat in the area of operation, then the conditional probability of the UAV becoming disabled by at least any one of the sources of threat at the position of (x,y) at time t under the condition that it follows trajectory S is

$$p_S(x,y,t) = P[E(x,y,t) \mid S(x,y,t)] \tag{6}$$

Since

$$E(x,y,t) = \bigcup_{i=1}^{N} E_i(x,y,t) \tag{7}$$

and $E_i(x,y,t)$ are not necessarily disjoint events,

$$p_S(x,y,t) \le \sum_{i=1}^{N} p_{S,i}(x,y,t) = \sum_{i=1}^{N}\left[\int_S f_{p,i}(x,y)f_{t,i}(t)\, l_1(x,y,t)\, l_2(x,y,t)dt\right] \tag{8}$$

by Union Bound (Stark & Woods, 1994). Thus we can easily compute an upper bound on the probability of a UAV becoming disabled if it follows a certain trajectory in an area with multiple threat sources. If l_1 and l_2 are assumed to be constant for any position and time on the trajectory and $f_{t,i}(t)$ is the same for all threat sources, then

$$p_S(x,y,t) \le l_1 l_2 \int_t f_t(t)\left[\sum_{i=1}^{N} f_{p,i}(x,y)\right]dt \tag{9}$$

since $f_{p,i}(x,y)$ and $f_t(t)$ are probability density functions and therefore integrable.

2.2 Probabilistic Threat Exposure Map (PTEM)

In the formulation of the probability of becoming disabled, introduced in the previous section, the dependency on position (the part of (9) in square brackets) is defined to be the Probabilistic Threat Exposure Map (PTEM), which quantifies the risk of exposure to sources of threat as a function of position. This concept is particularly useful in defining, in a single framework, various types of threats such as objects or locations that need to be avoided as far as possible, obstacles or restricted areas that should not be entered. This probabilistic map is not meant to provide the actual map of the area of operation, but to provide a way to plan the trajectory of a UAV to avoid obstacles and accomplish the given mission such as path planning and target tracking. All the threat sources (e.g. exposure to enemy radar, obstacles, and no-fly zones or restricted areas) are characterized in the same probabilistic framework using the sum of probability distributions of threats, obstacles and restricted

areas. As mentioned earlier, if a threat is characterized by a Gaussian *pdf*, there are two parameters needed to fully specify the threat; the mean value specifies the concentration point (location) of the threat source and the variance specifies how the threat can have an area (or volume for 3-D case) of effectiveness. The mean values and variances of each Gaussian function are specified such that the obstacles, restricted areas, no-fly zones and threats in the area of operation are all represented. Gaussian distribution can be used to model enemy radars or missiles as well as obstacles and restricted areas. Note that there is not necessarily one-to-one correspondence between the actual obstacles/restricted-areas/threats and the Gaussian functions used in the construction. An actual obstacle, for example, may require multiple Gaussian functions while a Gaussian function may be enough to represent a threat and restricted area, together.

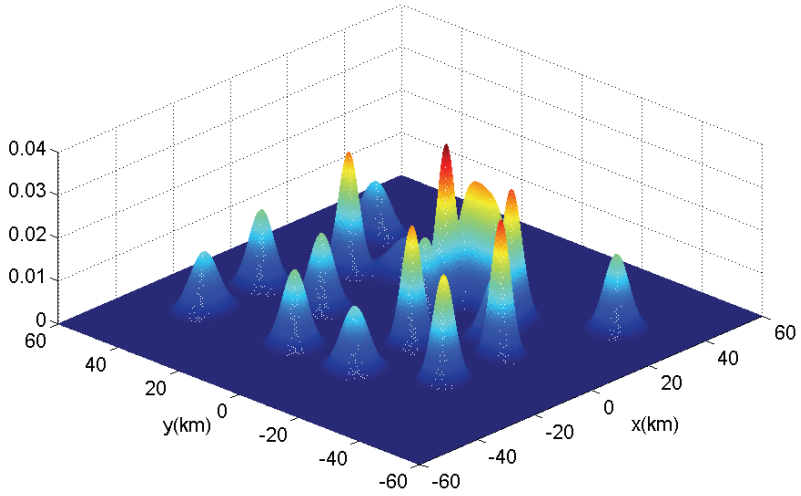


Figure 1. Probabilistic threat exposure map of an operation area

Once PTEM is constructed, there is no need to distinguish between the types of threats, obstacles, or no-fly zones and the use of the probabilistic map is sufficient for decision making. This is because the map already contains the information on the penalty of flying close to a source of threat or a restricted area. The PTEM quantifies the threat exposure level at a given position in the area of operation. A position in the area of operation is defined by its vector, $\underline{r}$, relative to the origin of a reference coordinate system. Let r be the representation of vector $\underline{r}$, i.e. $\underline{r} = [x \quad y]^T$, where x and y are the components of vector $\underline{r}$ along the x- and y- axes of the reference frame. The same notation will be used for all the other vectors and their representation throughout this chapter, i.e. $\underline{v}$ is a vector with its representation v in the reference frame. By using this notation, PTEM equation of an area of operation, modeled by Gaussian distributions, can be written as:

$$f(r) = \sum_{i=1}^{N} \frac{1}{2\pi\sqrt{\det(K_i)}} \exp\left[-\frac{1}{2}(r - \mu_i)^T K_i^{-1} (r - \mu_i) \right] dt \tag{10}$$

where μ_i and $\mathrm{K_i}$ are the mean vector and the covariance matrix of the i^{th} threat, respectively and defined as

$$\mu_i = \begin{bmatrix} \mu_{x,i} \\ \mu_{Y,i} \end{bmatrix}, \ \mathrm{K_i} = \begin{bmatrix} \sigma^2{}_{x,i} & 0 \\ 0 & \sigma^2{}_{y,i} \end{bmatrix} \tag{11}$$

Note that, threats modeled by Gaussian distributions are characterized by the well-known Multidimensional Gaussian Law. Fig.1 shows a sample PTEM constructed by a set of Gaussian *pdf*s. Note that if the parameters of *pdf*s are constants, then PTEM is time-invariant. Any other *pdf* can be used to construct the map as long as it is differentiable.

2.3 Restricted Regions Formulation

Based on the value of PTEM, restricted regions where the UAV should never enter can be defined in the area of operation. Such regions are quantified by a lower limit (f_r), where the value of PTEM is greater than or equal to f_r . Namely,

$$A_r(\mathrm{t}) = \left\{ \mathrm{r} : f(\mathrm{r}) \geq f_r \right\} \tag{12}$$

Note that, these regions are not fixed over time in the area of operation if the probabilistic map itself is time-variant with respect to position and effectiveness area. Nevertheless, if the position and effectiveness area of the threats are fixed but the effects of the threats are still time-variant, e.g. the level of threat exposure of the UAV increases as it stays longer in the area of operation, these regions will be fixed over time.

2.4 Gradient Search on PTEM

If the PTEM is differentiable, i.e. the area of operation is all modeled by Gaussians or any other differentiable distribution functions, the gradient search approach, which is extensively used in robotics and optimization literature (Konolige, 1996; Choi & Lee, 1996; Mitchell & Sastry, 2003; Ogren et al.,2004) can be employed. This determines the direction of minimum increase or steepest descent of the PTEM. In other words, it can be easily determined in which direction the UAV should move to minimize the threat exposure level or maximize the likelihood of avoiding a restricted region or a collision. Since the PTEM is constructed as the sum of differentiable functions, the determination of gradient or the sharpest-descent direction can be carried out by utilizing the sum of "directional" derivatives of $f(\mathrm{r})$ given in (10) along the axes of the reference coordinate system of choice. Let u be a direction, at position $\underline{\mathrm{r}}$, whose angle from the positive x-axis is ψ. Thus, the vector defining direction u is

$$\underline{\mathrm{u}} = \cos\psi \, \hat{I} + \sin\psi \, \hat{J} \tag{13}$$

where $\hat{I}, \hat{J}$ are the unit vectors of x- and y-axes of the reference frame, respectively. Then, the directional derivative of $f(\mathrm{r})$ along direction $\underline{\mathrm{u}}$ at position $\underline{\mathrm{r}}$ is

$$D_{\mathrm{u}} f(\mathrm{r}) = f_x(\mathrm{r})\cos\psi + f_y(\mathrm{r})\sin\psi \tag{14}$$

where $f_x(\mathrm{r})$ and $f_y(\mathrm{r})$ are the partial derivatives of $f(\mathrm{r})$ with respect to x and y, respectively:

$$f_{\mathrm{x}}(\mathrm{r}) = \sum_{\mathrm{i=l}}^{\mathrm{N}} -\frac{x-\mu_{x,i}}{2\pi\sigma_{x,i}^3\,\sigma_{y,i}} \exp\left\{-\frac{[(x-\mu_{x,i})^2\sigma_{y,i}^2 + (y-\mu_{y,i})^2\sigma_{x,i}^2]}{2\sigma_{x,i}^2\sigma_{y,i}^2}\right\} \tag{15}$$

$$f_{\mathrm{y}}(\mathrm{r}) = \sum_{\mathrm{i=l}}^{\mathrm{N}} -\frac{y-\mu_{y,i}}{2\pi\sigma_{x,i}\,\sigma_{y,i}^3} \exp\left\{-\frac{[(x-\mu_{x,i})^2\sigma_{y,i}^2 + (y-\mu_{y,i})^2\sigma_{x,i}^2]}{2\sigma_{x,i}^2\sigma_{y,i}^2}\right\} \tag{16}$$

At a given position, to find the direction along which the threat exposure is reduced the most, i.e. the steepest descent, the directional derivative in (14) should be minimized over angle ψ. Namely, "the minimizing direction", $\underline{\mathrm{u}}_{\min}$, at position $\underline{\mathrm{r}}$ in terms of its angle from the positive x-axis is

$$\psi_{\min}(\mathrm{r}) = \arg\min_{\psi} D_{\mathrm{u}} f(\mathrm{r}) \tag{17}$$

which yields "the minimizing angle" at position $\underline{\mathrm{r}}$ as

$$\psi_{\min}(\mathrm{r}) = \tan^{-1}\left[\frac{-f_y(\mathrm{r})}{-f_x(\mathrm{r})}\right] \tag{18}$$

Then, the vector representing the minimizing direction is

$$\underline{\mathrm{u}}_{\min} = \cos\psi_{\min}\hat{I} + \sin\psi_{\min}\hat{J} \tag{19}$$

Note, further, that the gradient of $f(\mathrm{r})$ is (Larson et al., 2002)

$$\nabla f(\mathrm{r}) = f_x(\mathrm{r})\hat{I} + f_y(\mathrm{r})\hat{J} \tag{20}$$

In terms of the gradient, the minimum value of the directional derivative is given by

$$\min D_{\mathrm{u}} f(\mathrm{r}) = -\left\|\nabla f(\mathrm{r})\right\| \tag{21}$$

Note that this is, in fact, the value of the directional derivative along direction $\underline{\mathrm{u}}_{\min}$, i.e.

$$\min D_{\mathrm{u}} f(\mathrm{r}) = D_{\underline{\mathrm{u}}_{\min}} f(\mathrm{r}) = f_x(\mathrm{r})\cos\psi_{\min} + f_y(\mathrm{r})\sin\psi_{\min} \tag{22}$$

3. Gradient Search Guidance Algorithm

The main goal of gradient search guidance algorithm is to generate feasible speed and heading commands for UAVs to safely pursue a moving target in an area with multiple sources of threats and/or restricted zones. This goal requires a trade-off between three possibly conflicting objectives given in the order of their priorities:(i) to avoid restricted

regions and obstacles (ii) to maintain the proximity of the target, (iii) to minimize the level of threat exposure. Thus, the strategy should guide the UAV autonomously to keep it within a pre-specified proximity of the target as well as trying to minimize the threat exposure at any time while avoiding restricted areas.

3.1 Target Following Strategy

The strategy guides a UAV by generating commanded heading, ψ_{cmd}, and speed, V_{cmd}. While generating these commands, the strategy takes the dynamic constraints of the UAV into account. Another requirement for the strategy, addressed in this algorithm, is that the strategy algorithm be executed on-line during the pursuit on an on-board computer/micro-processor. Thus, the strategy should be computationally feasible regarding the flight characteristics of the UAV and the configuration of the on-board processor/computer.

3.1.1 Preliminary Definitions

In this section, mathematical and geometric preliminaries are introduced to familiarize the reader with concepts, parameters and constraints used for the description and formulation of the strategy. Note that throughout the chapter C , D and V refer to circle, disk and cone, respectively. Let T_s be the "guidance update period" ', i.e., the commanded heading and speed are updated only when time $t = kT_s$, $k \in \{0,1,2,\ldots\}$. The positions of the UAV and the target are defined relative to the fixed reference frame where the PTEM is presented as a function of position.

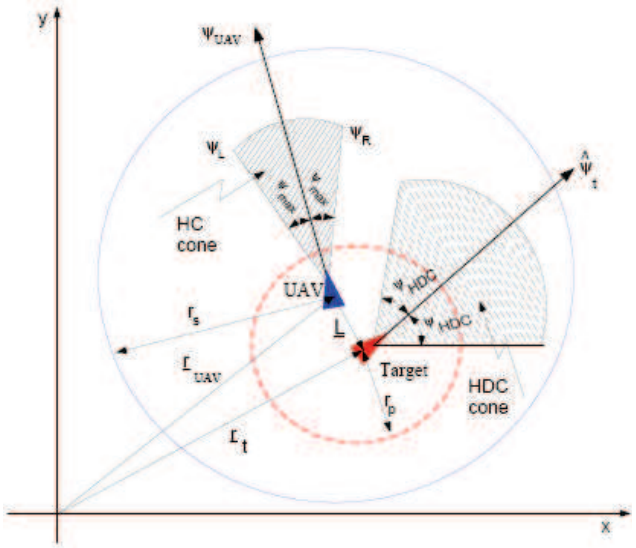


Figure 2. Proximity & sensor circles, HC & HDC cones, and LOS vector

As seen from Fig.2., the position vectors of the UAV and the target are defined to be $\underline{r}_{UAV}$ and $\underline{r}_t$ with (x_u, y_u) and (x_t, y_t) coordinates, respectively. The headings of the UAV and the target are angles measured from positive x-axis as ψ_{UAV} and ψ_t, respectively. Let

$\underline{L}$, ``line of sight'' (LOS) vector, be the position vector of target with respect to UAV and written as

$$\underline{L} = (x_t - x_u)\hat{I} + (y_t - y_u)\hat{J} \tag{23}$$

Thus, the LOS angle, ψ_{LOS}, can be computed as

$$\psi_{LOS}(k) = \tan^{-1}\left(\frac{y_t - y_u}{x_t - x_u}\right) \tag{24}$$

Then, the time rate of change of ψ_{LOS} is calculated as

$$\dot{\psi}_{LOS}(k) = \frac{(\dot{y}_t - \dot{y}_u)(x_t - x_u) - (\dot{x}_t - \dot{x}_u)(y_t - y_u)}{(x_t - x_u)^2 + (y_t - y_u)^2} \tag{25}$$

Proximity circle, $\mathrm{C_p}(k)$, denotes a circle, at the k^{th} (current) update instant, centered at the predicted target position at the $(k+1)^{th}$ (next) update instant and with a specified radius, r_p (see Fig.2.). Namely,

$$\mathrm{C_p}(k) = \left\{\underline{z} \in \Re^{2:} : \left|\underline{z} - \underline{\hat{r}}_t(k+1)\right| = \mathrm{r_p}\right\} \tag{26}$$

Similarly, *Proximity disk*, $\mathrm{D_p}(k)$, is defined to be the region bounded by $\mathrm{C_p}(k)$,i.e.

$$\mathrm{D_p}(k) = \left\{\underline{z} \in \Re^{2:} : \left|\underline{z} - \underline{\hat{r}}_t(k+1)\right| \le \mathrm{r_p}\right\} \tag{27}$$

$\mathrm{D_p}(k)$ is introduced to define the proximity of the target and thus it is a design parameter that quantifies how close to the target the strategy should keep the UAV during the pursuit. Note that the proximity of the target is the objective of the strategy with the second highest priority. Once this objective is secured, the strategy should try to achieve the last objective, minimizing the threat exposure level. In this regard, $\mathrm{D_p}(k)$ quantifies the trade-off between these two objectives. Similar to $\mathrm{C_p}(k)$, *Reachability circle*, $\mathrm{C_r}(k)$, is defined to be a circle, centered at the current UAV position (see Fig.3) and has a radius determined by the speed of the UAV and the guidance update period, $\mathrm{T_s}$:

$$\mathrm{C_r}(k) = \left\{\underline{z} \in \Re^{2:} : \left|\underline{z} - \underline{\mathrm{r}}_{UAV}(k)\right| = T_s \times V_{UAV}(k)\right\} \tag{28}$$

where $\underline{\mathrm{r}}_{UAV}(k)$ and $V_{UAV}(k)$ are the samples of the UAV position and speed at the k^{th} update instant, respectively. Note that $\mathrm{C_p}(k)$ is centered at the predicted target position at the next update instant while $\mathrm{C_r}(k)$ is centered at the UAV position at the current update instant. Thus, these two circles are used to determine (i) whether it is possible stay within $\mathrm{D_p}(k)$ until the next update instant and (ii) if not, the range of directions that would steer the UAV towards $\mathrm{D_p}(k)$. Based on $\psi_{HDC}(k)$ from Fig.2, HDC (*Heading Difference Constraint*) Cone (V_{HDC}) is defined to be the range of headings that are considered to be close to the estimated heading of the target $\hat{\psi}_t$. Thus, as shown in Fig.2, V_{HDC} moves with the target,

is centered at its estimated heading and expands in both clockwise and counter-clockwise directions by $\psi_{HDC}(k)$. Similarly, HC (*Heading Constraint*) Cone, V_{HC}, defines the range of headings that are admissible when only the dynamics of the UAV is considered. V_{HC}, as shown in Fig.2, moves with the UAV, is centered at its heading and expands in both clockwise and counter-clockwise directions by $\psi_{\max}$.

When the UAV is outside the proximity disk, $D_p(k)$, it is desirable to know (i) whether it is possible to move into $D_p(k)$ at the current speed and (ii) if it is, the range of feasible headings that would steer the UAV towards $D_p(k)$. To be able to answer the first question, the PR (Proximity Range) Cone, V_{PR}, is constructed, as shown in Fig.3, by the intersection points of $C_r(k)$ and $C_p(k)$.

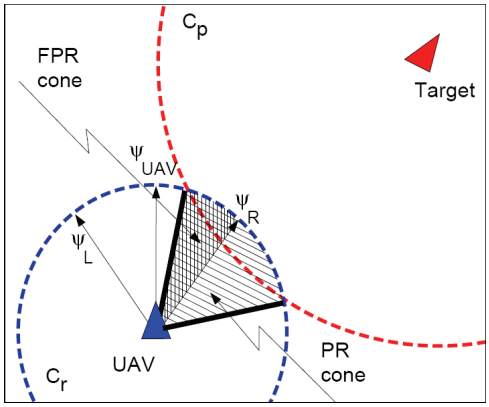


Figure 3. General proximity disk cone

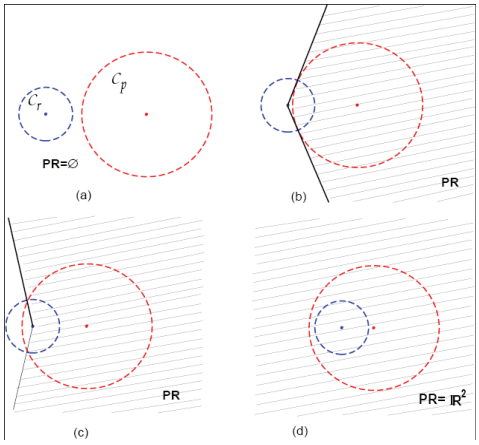


Figure 4. PR cone in various cases of position of C_r relative to C_p, (a) $C_r(k) \not\subset D_p(k)$ and $C_r(k) \cap C_p(k) = \varnothing$, (b),(c) $C_r(k) \not\subset D_p(k)$ and $C_r(k) \cap C_p(k) \neq \varnothing$, (d) $C_r(k) \subset D_p(k)$ and $C_r(k) \cap C_p(k) = \varnothing$

Obviously, if $C_r(k) \not\subset D_p(k)$ and $C_r(k) \cap C_p(k) = \varnothing$ (Note that $\varnothing$ denotes the empty set), then $V_{PR} = \varnothing$ and thus the answer to question (i) is negative (see (a) of Fig.4). When $C_r(k) \not\subset D_p(k)$ and $C_r(k) \cap C_p(k) \neq \varnothing$ (see (b) and (c) of Fig.4), however, the feasibility of V_{PR} should be investigated by determining its intersection with V_{HC}. If $V_{HC} \cap V_{PR} = \varnothing$, then the answer to question (i) is still negative. When $V_{HC} \cap V_{PR} \neq \varnothing$, the answer to question (i) is affirmative, and the answer to question (ii) is the intersection cone, which is defined to be FPR (Feasible Proximity Range) Cone, i.e. $V_{FPR} = V_{HC} \cap V_{PR}$ as shown in Fig.3. When $C_r(k) \subset D_p(k)$, as seen in (d) of Fig.4, $C_r(k) \cap C_p(k) = \varnothing$.

However, in this special case, $V_{PR} = \Re^2$ since any direction will lead to $D_p(k)$, and thus, $V_{FPR} = V_{HC}$. The objective of the strategy with the highest priority is to always avoid the restricted areas. To be able to do so, the range of the headings that would steer the UAV towards a restricted area, especially when the restricted area is close-by, should be determined. This can be accomplished by utilizing the directions that are tangent to the level curve of the PTEM that passes through a given UAV position. Note that the tangent directions are always normal to the gradient, $\psi_{\min}(k)$. The angles of the two tangent directions are referred to as $\psi_{tg_R}(k)$ and $\psi_{tg_L}(k)$ as shown in Fig.5. Then, SHR (Safe Heading Range) Cone, V_{SHR}, is defined to be the range of directions within the V_{HC} through which the directional derivative of the PTEM is zero or negative.

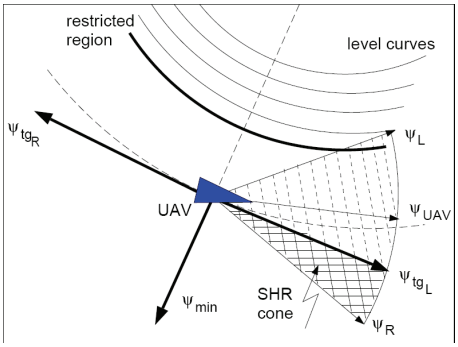


Figure 5. SHR cone

3.1.2 Decision Factors and Strategy States

The intelligent strategy first computes the "desired" heading and speed as well as the "admissible" ranges of heading and speed. The desired signals are computed in accordance with the three objectives of the strategy without considering the dynamic and strategy-imposed constraints. At the same time, the admissible ranges are determined based on the states of the UAV, the local PTEM and the constraints. Then, the strategy generates the commanded signals (heading and speed) by considering the desired signals and their respective admissible ranges. When a desired signal is within the respective admissible range, then obviously the desired signal is assigned as the commanded signal. Otherwise,

the boundary of the admissible range that is closest to the desired signal is selected to be the commanded signal. This section of the chapter introduces the decision factors and decision states that are defined based on the strategy objectives and their assigned priorities and used by the strategy to infer the desired heading, desired speed and their admissible ranges. There are four decision factors that are used to determine the decision states:

Factor1: This factor determines whether the UAV is in a risk of getting into a restricted region. It is quantified by using V_{SHR} and evaluating the PTEM at three positions that would be the positions of the UAV in n update periods ahead if it flies with its current speed in the current, maximum right and maximum left heading directions ($\psi_i, i = \{1,2,3\}$). Thus, these three prospective positions are calculated as

$$x_i(k+n) = x_c + V_c(k) n T_s \cos\psi_i(k) \tag{29}$$

$$y_i(k+n) = y_c + V_c(k) n T_s \sin\psi_i(k) \tag{30}$$

where n is the number of update periods that would take for the UAV to make a 90 degree-turn by utilizing the maximum available turn rate. It is computed by $n = ceil[\pi/(2\psi_{\max})]$ where *ceil* is a function that rounds a real number to the nearest integer towards positive infinity.

Then, the PTEM is evaluated at these three positions. If all $f(x_i, y_i)$ are greater than or equal to f_r, then the UAV is in HIGH risk since all the headings steer the UAV to a restricted area. If only some of $f(x_i, y_i)$ is greater than f_r and $V_{SHR} = \varnothing$ then the UAV is still in HIGH risk since $V_{SHR} = \varnothing$ means that the entire V_{HC} is directed completely towards the restricted area. If $V_{SHR} \neq \varnothing$ and there are some $f(x_i, y_i)$ less than f_r, then the UAV is considered to be in LOW risk. If all $f(x_i, y_i)$ are less than f_r, then the risk is NONE since none of the directions steers the UAV to the restricted area.

Factor 2: This factor determines whether the UAV will be within the proximity of the target at the next update instant. This is quantified by employing V_{PR}, V_{HC} and V_{SHR} depending on the answer to Factor 1. When Factor 1 is NONE (i.e. no risk of getting into A_r), the intersection of V_{PR} and V_{HC} is taken. If

$$V_{HC} \cap V_{PR} \neq \varnothing \tag{31}$$

then the answer is YES, i.e. it is feasible for the UAV to be within $D_p(k+1)$. Otherwise, Factor 2 is NO. When Factor 1 is LOW, V_{HC} is replaced by V_{SHR} in (31). Note that in the case when $V_{PR} = \Re^2$, i.e. $C_r(k)$ is completely in $D_p(k)$, Factor 2 is always YES when Factor 1 is NONE or LOW. When Factor 1 is HIGH, the answer to Factor 2 is not defined.

Factor 3: This factor determines whether the headings of the UAV and the target are close. This is quantified by the intersection of V_{HC} and V_{HDC} when Factor 1 is NONE.

If

$$V_{HC} \cap V_{HDC} \neq \varnothing \tag{32}$$

then the answer is YES, i.e. the UAV and the target headings are close, otherwise the answer is NO. When Factor 1 is LOW, V_{HC} is replaced by V_{SHR} in (32).

Factor 4: This factor determines whether the UAV is heading towards $D_p(k)$ only when Factor 1 is NONE or LOW, Factor 2 is NO and Factor 3 is YES. This is quantified by comparing $\psi_{LOS}(k)$ with ($V_{HC} \cap V_{HDC}$) or ($V_{SHR} \cap V_{HDC}$). When Factor 1 is NONE and

$$\psi_{LOS}(k) \subset (V_{HC} \cap V_{HDC}) \tag{33}$$

then the answer is YES, namely the UAV is heading towards $D_p(k)$, otherwise the answer is NO. When Factor 1 is LOW, V_{HC} is replaced by V_{SHR} in (32) as done in Factors 2 and 3.

STATES	DECISION FACTORS			
	1	2	3	4
1	NONE	NO	YES	YES
2	NONE	NO	YES	NO
3	NONE	NO	NO	N/A
4	NONE	YES	N/A	N/A
5	LOW	NO	YES	YES
6	LOW	NO	YES	NO
7	LOW	NO	NO	N/A
8	LOW	YES	N/A	N/A
9	HIGH	N/A	N/A	N/A

Table 1. Decision factors and strategy states (N/A: Not Applicable)

Table 1 summarizes all the rules to determine the states of the strategy based on the decision factors. In *States-1* to -4, the UAV is not in any risk of flying into a restricted area. In *State-1*, the UAV is not in D_p, but the headings of the UAV and the target are close and the UAV is heading towards D_p. However, In *State-2*, the UAV is not heading towards D_p. Note that *Factor 4* is considered only when there is NONE or LOW risk of flying into a restricted region and the headings of the UAV and the target are close. In *State-3*, the UAV is not in the proximity disk and moreover the headings of the UAV and the target are not close. As the simulation results will reveal, this state occurs rarely due to the imposed heading difference constraint by the strategy. In *State-4*, the UAV is within $\tilde{D}_p$. Note also that this is the state that strategy tries to keep the UAV in as much as possible during the mission. In *States-5* to -8, the UAV is in low risk region. In both *States-5* and -6 , the UAV is outside the proximity disk and the headings of the UAV and the target are close. However, in *State-6*, the UAV is not heading towards D_p . In *State-7*, the UAV is not in the proximity disk and moreover the headings of the UAV and the target are not close. In *State-8*, the UAV is within D_p. Note that *Factors* 3 and 4 are not considered in *States-4* and -8 because the UAV is already in the proximity of the target. In *State-9*, the UAV is in high risk region. Thus, other decision factors are not considered.

3.1.3 Computation of Desired Heading and Admissible Range

As stated earlier, the commanded heading is computed based on the desired heading ψ_{des} and the AHR (Admissible Heading Range) cone, V_{AHR}, which is defined to be the cone that consists of the feasible heading directions. Depending on the strategy state at a given time, different criteria are used to determine ψ_{des} and V_{AHR}. Note that the rules in this section and the next section can be defined to be the rules that represent a strategy to select the commanded signals. Recall that the strategy has three objectives with different levels of priority. There are three headings, $\hat{\psi}_{\text{t}}(k), \psi_{\min}(k)$ and $\psi_{\text{LOS}}(k)$, one of which is chosen as the desired heading, $\psi_{\text{des}}(k)$, at the k^{th} update instant, based on the objective of the strategy in a given strategy state.

In *States-1* and *-2*, the objective with the highest priority (avoidance of the restricted regions) is considered to be "achieved" since Factor 1 is NONE. However, the second objective needs to be targeted since Factor 2 is NO, i.e., D_{p} should be intercepted. Since the Factor 3 is YES, the headings of the UAV and the target are already close. To achieve the second objective, a pursuit guidance law is employed for the UAV to intercept D_{p} as soon as possible:

$$\psi_{\text{des}}(k) = \psi_{LOS}(k) + K_D \dot{\psi}_{LOS}(k) \tag{34}$$

where the first term represents the well-known velocity guidance (Pastrick et al., 1981) and the derivative term, with gain K_D, is added to improve the pursuit performance. The admissible heading range is selected such that the commanded heading does not violate the dynamic constraint of the UAV and the heading difference constraint imposed by the strategy itself, i.e.

$$\text{V}_{AHR} = \text{V}_{HC} \cap \text{V}_{HDC} \tag{35}$$

Note that the desired heading selection strategy is the same for *States-1* and *-2*. However, the gain, K_D, in (34) will be selected differently. Another difference originates from the selection of the desired speed as explained in the next section.

In *State-3*, similar to the first two states, the objective with the highest priority is achieved but the second objective is not. Furthermore, since the Factor 3 is NO, the headings of the UAV and the target are not close to each other. In a high speed pursuit, flying in a direction different from that of the target will lead the UAV to lose the proximity of the target very soon. In other words, it will lead to the failure of the second objective. To turn the UAV in the same direction with the target, $\psi_{\text{des}}(k)$ is selected to be $\hat{\psi}_{\text{t}}(k)$ and V_{AHR} is selected to be V_{HC} to allow the UAV to make the sharpest turn possible. Simulation experiments have shown that, during high speed pursuits, the strategy employed in *States-1* and *-2* would not be as efficient particularly due to occurrence of restricted regions between $\tilde{\text{D}}_{\text{p}}$ and the UAV. This is partly the reason why a different strategy is employed in *State-3*. Further, note that as pursuit speed decreases, it will be less likely for this state to occur because, as to be explained in Section 3.1.5, the heading difference constraint will be relaxed.

In *State-4*, since the first two objectives of the strategy are achieved, the third one can be targeted. Thus, the strategy, in this state, should try to minimize the threat exposure level

while ensuring that the other objectives are not compromised. Note that strategy should steer the UAV in a direction that would minimize threat exposure, which, in turn, implies that the UAV is guided away from any possible restricted area. In other words, objective with the highest priority has no conflict with the third objective. However, moving in a direction to minimize the threat exposure may be in conflict with the second objective, i.e. staying in D_p. Thus, in *State-4*, an efficient compromise between the second and the third objective should be formulated while considering that the second objective has a higher priority. As stated earlier, the third objective is quantified by the steepest descent direction, $\psi_{\min}(k)$. Namely, if the third objective was the only concern, $\psi_{\min}(k)$ would be the desired heading. On the other hand, if the UAV was commanded to fly tangent to C_p, the UAV would never leave D_p. Thus, a direction tangent to C_p is defined to quantify the second objective. Recall that the computation of $\psi_{\min}(k)$ is already introduced in Section 2.4. Now, calculation of the tangent direction will be presented. Then, the method will be explained that computes $\psi_{\text{des}}(k)$ based on $\psi_{\min}(k)$ and the tangent direction to quantify the trade-off between the two conflicting objectives. First, the admissible heading range is determined to ensure that the dynamic constraints of the UAV are not violated. Furthermore, any direction that would certainly move the UAV outside $D_p(k)$ should be eliminated. Thus,

$$V_{AHR} = V_{HC} \cap V_{PR} \tag{36}$$

V_{HDC} is not considered (i.e *Factor 3* is not considered) in this state because *Factor 2* is already YES. Note that the most likely heading of the UAV is $\psi_{\min}(k)$ provided it is within V_{AHR}. Hence, a temporary heading, $\psi_{\text{temp}}(k)$ is defined to be $\psi_{\min}(k)$ if $\psi_{\min}(k) \in V_{AHR}$ and, otherwise, the boundary of V_{AHR} that is closest to $\psi_{\min}(k)$. $\psi_{temp}(k)$ is used to define a local coordinate system, as shown in Fig.6, whose origin is at the current UAV position and y-axis, y_L, has angles $\psi_{temp}(k)$ from the positive x-axis of the inertial reference frame. The two points where y_L intersects C_p are defined to be y_{L_1} and y_{L_2}.

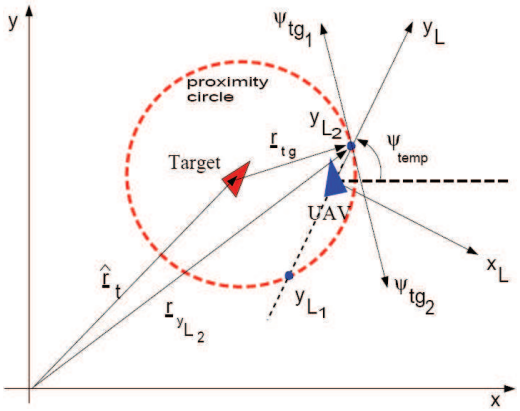


Figure 6. Tangent heading bounding the UAV turns

Transformation between the inertial reference frame and the local frame is used to facilitate the computation of $y_{L_1}(k)$ and $y_{L_2}(k)$. The rotation matrix from the inertial frame to the local frame is

$$\mathrm{R}_{LI}(k) = \begin{bmatrix} \sin\psi_{temp}(k) & -\cos\psi_{temp}(k) \\ \cos\psi_{temp}(k) & \sin\psi_{temp}(k) \end{bmatrix} \tag{37}$$

The predicted target position, then, can be written in the local frame by employing the rotation matrix as

$$\hat{\mathrm{r}}_{t,L}(k+1) = \mathrm{R}_{LI}(k)\left[\hat{\mathrm{r}}_t(k+1) - \mathrm{r}_u(k)\right] \tag{38}$$

where $\hat{\mathrm{r}}_t$ is the representation of $\hat{\underline{\mathrm{r}}}_t$ in the inertial frame, and $\hat{\mathrm{r}}_{t,L}$ is the representation in the local frame. Equation (38) gives the components of $\hat{\underline{\mathrm{r}}}_t$ in the local frame as

$$\hat{x}_{t,L}(k+1) = \left[\hat{x}_t(k+1) - x_u(k)\right]\sin\psi_{temp}(k) - \left[\hat{y}_t(k+1) - y_u(k)\right]\cos\psi_{temp}(k) \tag{39}$$

$$\hat{y}_{t,L}(k+1) = \left[\hat{x}_t(k+1) - x_u(k)\right]\cos\psi_{temp}(k) + \left[\hat{y}_t(k+1) - y_u(k)\right]\sin\psi_{temp}(k) \tag{40}$$

Now, $\mathrm{C_p}$ can be easily formulated in the (x_L, y_L) local frame as

$$\left[x_L - \hat{x}_{t,L}(k+1)\right]^2 + \left[y_L - \hat{y}_{t,L}(k+1)\right]^2 = {r_p}^2 \tag{41}$$

As Fig.6 implies, $y_{L_1}(k)$ and $y_{L_2}(k)$ are solutions to (41) when $x_L = 0$, since y_{L_1} and y_{L_2} are defined to lie on the y_L axis

$$y_{L_{1,2}}(k) = \hat{y}_{t,L}(k+1) \mp \sqrt{{r_p}^2 - \hat{x}_{t,L}(k+1)} \tag{42}$$

Note, from Fig.6, that $y_{L_1}(k)$ is always negative and $y_{L_2}(k)$ is always positive in *State-4* since the UAV is in $\mathrm{D_p}$. Since the UAV is likely to head towards y_{L_2}, the two tangent directions of $\mathrm{C_p}$ at this point need to be computed. Recall that $\mathrm{C_p}$ is centered at the predicted target position, $\hat{\underline{\mathrm{r}}}_t(k+1)$, at the next update instant. If the radial direction, $\underline{\mathrm{r}}_{tg}(k)$, from the center of $\mathrm{C_p}$ to y_{L_2} is known, then the tangent directions are easily computed as they are normal to the radial directions, as shown in Fig.6. Note that,

$$\underline{\mathrm{r}}_{tg}(k) = \underline{\mathrm{r}}_{y_{L_2}}(k) - \hat{\underline{\mathrm{r}}}_t(k+1) \tag{43}$$

where $\underline{\mathrm{r}}_{y_{L_2}}(k)$ is the position vector of point y_{L_2} relative to the inertial reference frame. By employing the inverse of the transformation used in (38), the representation of $\underline{\mathrm{r}}_{y_{L_2}}(k)$ in the inertial frame is computed as

$$\mathrm{r}_{y_{L_2}}(k) = R_{LI}^T(k)\begin{bmatrix} 0 \\ y_{L_2}(k) \end{bmatrix} + \mathrm{r}_u(k) \tag{44}$$

After carrying out matrix multiplication and addition, (44) yields $\underline{r}_{y_{L_2}}(k)$, written as a vector,

$$\underline{r}_{y_{L_2}}(k) = \left\{x_u(k) + y_{L_2}(k)\cos[\psi_{temp}(k)]\right\}\hat{I} + \left\{y_u(k) + y_{L_2}(k)\sin[\psi_{temp}(k)]\right\}\hat{J} \tag{45}$$

Substituting (45) in (43) yields

$$\begin{aligned}\underline{r}_{tg}(k) = &\left\{x_u(k) - \hat{x}_t(k+1) + y_{L_2}(k)\cos[\psi_{temp}(k)]\right\}\hat{I} \\ &+ \left\{y_u(k) - \hat{y}_t(k+1) + y_{L_2}(k)\sin[\psi_{temp}(k)]\right\}\hat{J}\end{aligned} \tag{46}$$

Then the tangent angles, shown in Fig.6, can be calculated as:

$$\psi_{tg1,2}(k) = \tan^{-1}\left\{\frac{y_u(k) - \hat{y}_t(k+1) + y_{L_2}(k)\sin[\psi_{temp}(k)]}{x_u(k) - \hat{x}_t(k+1) + y_{L_2}(k)\cos[\psi_{temp}(k)]}\right\} \tag{47}$$

Among the two tangent directions, ψ_{tg1} and ψ_{tg2}, the one, referred to as $\psi_{tg}(k)$, that is closest to the current UAV heading, $\psi_{UAV}(k)$, is selected to quantify the second objective. For example, in Fig.6, $\psi_{tg}(k) = \psi_{tg1}(k)$ since it is closer to the UAV heading.

Once the two objectives are quantified by $\psi_{temp}(k)$ to minimize threat exposure and $\psi_{tg}(k)$ to stay with the proximity of the target, the next step is to consolidate these two possibly conflicting objectives in an efficient way. Let $\underline{u}_{temp}(k)$ and $\underline{u}_{tg}(k)$ be the unit vectors along $\psi_{temp}(k)$ and $\psi_{tg}(k)$ directions, respectively. Namely,

$$\underline{u}_{temp}(k) = \cos\psi_{temp}(k)\hat{I} + \sin\psi_{temp}(k)\hat{J} \tag{48}$$

$$\underline{u}_{tg}(k) = \cos\psi_{tg}(k)\hat{I} + \sin\psi_{tg}(k)\hat{J} \tag{49}$$

Then, the vector with the desired heading direction is constructed as the weighted vectorial sum of $\underline{u}_{temp}(k)$ and $\underline{u}_{tg}(k)$ as

$$\underline{r}_{des}(k) = [1 - \alpha(k)]\underline{u}_{temp}(k) + \alpha(k)\underline{u}_{tg}(k) \tag{50}$$

where $\alpha(k)$, a real number between 0 and 1, is used to quantify the level of trade-off between the two objectives. Note that when the UAV is close to the center of C_p, there is no immediate danger of leaving D_p and thus minimizing the threat exposure should be the primary objective. This implies that $\alpha(k)$ should be 0 so that $\underline{r}_{des}(k) = \underline{u}_{temp}(k)$. On the other hand, when the UAV is close to C_p, there is an immediate risk of leaving the proximity of target and thus $\underline{r}_{des}(k) = \underline{u}_{tg}(k)$ (i.e. $\alpha(k) = 1$) so that the UAV does not head towards outside of D_p. When the UAV is between these two extreme cases, $\alpha(k)$ should take a value between 0 and 1 so that the two objectives are consolidated. This logic is formulated by a scheduling scheme for $\alpha(k)$ as

$$\alpha(k) = \begin{cases} 1, & \alpha_{sch}(k) \le \alpha_{tg} \\ \dfrac{\alpha_{sch}(k)(\alpha_{temp} - 1)}{\alpha_{temp} - \alpha_{tg}}, & \alpha_{tg} < \alpha_{sch}(k) \le \alpha_{temp} \\ 0, & otherwise \end{cases} \tag{51}$$

In this scheduling, α_{tg} and α_{temp} are strategy design parameters and α_{sch} quantifies how close the UAV is to leaving $\mathrm{D_p}$ as

$$\alpha_{sch}(k) = \frac{y_{L_2}(k)}{2r_p} \tag{52}$$

Note that when $\alpha(k)$ is close to 0, the UAV is actually close to $\mathrm{C_p}(k)$ but heading towards inside of $\mathrm{D_p}(k)$ and thus this case is not considered as a danger of leaving $\mathrm{D_p}(k)$. By using the angle of the resultant vector in (50), the desired heading angle in *State-4* is computed as

$$\psi_{des}(k) = \tan^{-1}\left\{\frac{[1-\alpha(k)]\sin\psi_{temp}(k) + \alpha(k)\sin\psi_{tg}(k)}{[1-\alpha(k)]\cos\psi_{temp}(k) + \alpha(k)\cos\psi_{tg}(k)}\right\} \tag{53}$$

In *States-5* to *-8*, exactly the same strategies as in *States-1* to *-4*, respectively, for computing $\psi_{des}(k)$ have been implemented. The only difference in these states is that there is a LOW threat region instead of having NONE threat region. Thus, V_{HC} is replaced by V_{SHR} during the calculation of V_{AHR} in all these states.

In *State-9*, the objective with the highest priority is not considered to be "achieved" since Factor 1 is HIGH. Thus, the UAV should be commanded to make a turn to stop approaching the restricted area. Since $\psi_{\min}(k)$ is the direction of the sharpest descent, it is selected to be $\psi_{des}(k)$. Furthermore, the UAV should make the turn as fast as possible, thus, V_{AHR} is selected to be V_{HC} to allow the UAV to make the sharpest turn possible.

3.1.4 Computation of Desired Speed and Admissible Range

In the previous section, the computation of the commanded heading at each update instant is presented. Once the commanded heading, $\psi_{cmd}(k)$, is computed at the k^{th} update instant, the strategy computes the desired speed, $V_{des}(k)$, based on $\psi_{cmd}(k)$ and the current decision state. Furthermore, an admissible speed range is determined based on the speed and acceleration constraints of the UAV. Note that during a pursuit mission, the target speed might be varying drastically. Further, the path of the UAV, determined by the commanded heading, might be significantly different from that of the target because the strategy steers the UAV to avoid restricted areas and minimize threat exposure level. Thus, the commanded speed is determined to address objective-2, i.e. to help maintain or obtain the proximity of the target in almost all decision states. The exception is *State-9* where the commanded speed is determined to help avoid restricted areas.

In *States-1* to *-8*, the desired speed is calculated by a proportional control algorithm based on two different error signals. The first one, speed error, is the difference between the speed of

the UAV and the estimated target speed. This is used to ensure that the UAV speed will be adjusted as the target speed varies. The second one, the position error, is defined to quantify the distance between the UAV and $D_p(k)$. To compute the position error, a local reference frame is defined as shown in Fig.7, such that its origin is at the UAV position and its y-axis, y_L, has an angle of $\psi_{cmd}(k)$ from the positive x-axis of the inertial frame. Note that this local frame is similar to the one used in the previous section except the angle used to define the orientation relative to the inertial frame. Then, the position error, $e(k)$, is defined to be the arithmetic mean of the two intersection points of y_L -axis with C_p.

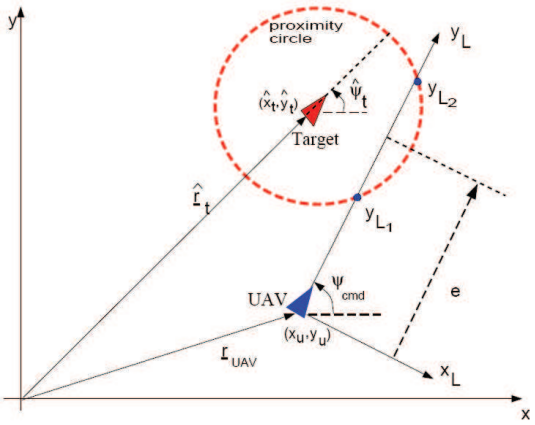


Figure 7. Inertial and UAV local coordinate systems

Note that y_{L_1} and y_{L_2} can be calculated by using the same approach detailed in previous section with $\psi_{cmd}(k)$ that replaces $\psi_{temp}(k)$. The proportional control algorithm is formulated as

$$V_{des}(k) = V_{UAV}(k) + K_s[\hat{V}_t(k) - V_{UAV}(k)] + K_e[e(k)/T_s] \tag{54}$$

where $\hat{V}_t(k)$ is the estimated target speed, K_s and K_e are the proportional gains for the speed and position errors, respectively. Note that different values can be assigned to the gains in different states. For example, for the simulations presented in Section 3.3, in *State-1* and *State-5*, the gains are twice as big as the values used in other states. This is because, in these states, the UAV is outside D_p but heading towards it (Recall that *Factor-4* is YES). Thus, a greater speed increase should be commanded so that it will take the UAV a shorter time to attain D_p. As stated earlier, in *State-9*, the objective with the highest priority needs to be addressed. Namely, there is a HIGH risk of incursion into a restricted area and the sharpest descent direction is commanded to turn the UAV away from the restricted area. The speed command is also utilized to improve the performance of the strategy. Note that the lower the speed of a UAV, the sharper turns it can make. Thus, the minimum speed possible, given the deceleration constraint of the UAV, is commanded, i.e.

$$V_{des}(k) = V_{UAV}(k) + a_{\min} T_s \tag{55}$$

Once the desired speed is determined, the admissible range for the speed should be determined to compute the commanded speed so that the speed and acceleration constraints of the UAV are not violated. The upper and the lower bounds of the admissible speed range are calculated as

$$V_{\max_{const}}(k) = \min\{V_{\max}, V_{UAV}(k) + a_{\max} T_s\} \tag{56}$$

$$V_{\min_{const}}(k) = \min\{V_{\min}, V_{UAV}(k) + a_{\min} T_s\} \tag{57}$$

3.1.5 Scheduling Scheme for Heading Difference Constraint

Recall that V_{HDC} is introduced to serve as a strategy imposed constraint on the heading of the UAV. This is necessary because the turning radius of a vehicle increases as its speed increases. If the heading difference is not bounded, the strategy may change the UAV heading drastically to minimize the threat exposure when the UAV is within $D_p(k)$. This, in turn, may increase the risk of the target getting outside $D_p(k)$ and eventually outside the sensor range. To restrict the motion of the UAV in and around the direction where $D_p(k)$ is heading, $\psi_{HDC}(k)$ is introduced. However, as stated earlier, this constraint, if it was fixed, would become a liability in the case of low speed and even more so when the target stops. Thus, a scheduling scheme is developed to impose a heading difference constraint in high speed pursuit and to relax it when the target moves slow or stops. The scheduling should be done in such a way that no discontinuity is introduced in the computation of the commanded heading. According to the scheduling scheme used (see Fig.8), $\psi_{HDC}(k)$ is set to a constant, $\psi^*_{HDC}(k)$, when the estimated target speed, $\hat{V}_t$, is high (i.e. greater than V_{thres_2}); when the target slows down, the constraint is gradually relaxed by linearly increasing $\psi_{HDC}(k)$. This increase is performed with such a slop that $\psi_{HDC}(k)$ becomes 180° when $\hat{V}_t$ is equal to another threshold, V_{thres_1}. When the target moves very slow or stops (i.e. $\hat{V}_t$ is less than V_{thres_1}), $\psi_{HDC}(k)$ is set to 180°, which effectively removes the constraint.

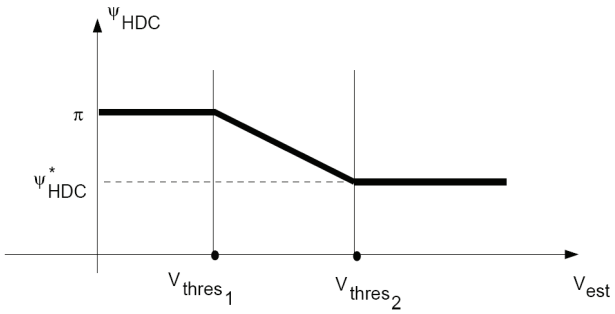


Figure 8. Heading difference constraint angle ψ_{HDC}

3.1.6 Detection of Local Minima

Note that $\underline{u}_{\min}(k)$ is the sharpest descent direction at the current update instant, k. Thus, the directional derivative of PTEM in this direction

$$D_{\underline{u}_{\min}(k)} f(\underline{r}(k)) \leq 0 \tag{58}$$

Let $\underline{r}(k+1)$ be the position vector of the UAV at the next update instant, $(k+1)$ if it flies with the current speed in the sharpest descent direction. The directional derivative at $\underline{r}(k+1)$ in the direction of $\underline{u}_{\min}(k)$ is referred to as $D_{\underline{u}_{\min}(k)} f(\underline{r}(k+1))$. Thus, the local minima is considered to be present ahead if (see Fig.9)

$$D_{\underline{u}_{\min}(k)} f(\underline{r}(k)) D_{\underline{u}_{\min}(k)} f(\underline{r}(k+1)) < 0 \tag{59}$$

When this condition occurs there is a local minimum of the PTEM ahead if the UAV would fly in the sharpest descent direction. If the strategy keeps commanding $\psi_{\min}$ through such a local minima, the simulation experiments have shown that the commanded heading may show an unnecessary oscillation. To prevent this, during the occurrence of local minima, a vectorial sum of the two sharpest descent directions is computed as

$$\underline{u}_{\min,avg}(k) = \underline{u}_{\min}(k) + \underline{u}_{\min}(k+1) \tag{60}$$

Thus, the angle of this new direction is

$$\psi_{\min,avg}(k) = \tan^{-1}\left[\frac{\sin\psi_{\min}(k) + \sin\psi_{\min}(k+1)}{\cos\psi_{\min}(k) + \cos\psi_{\min}(k+1)}\right] \tag{61}$$

which replaces $\psi_{\min}(k)$ in the current execution of the strategy.

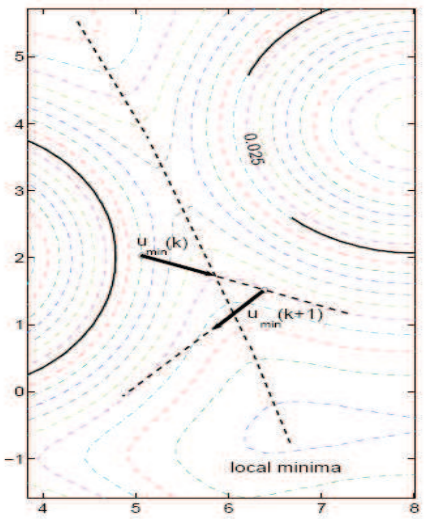


Figure 9. Detection of local minima

3.3 Simulation Results

To test the full capabilities of the strategy, 3 different tracking cases, which are representative of most likely scenarios, are simulated. In the $30\,km \times 30\,km$ area of operation, there are 17 threat sources along with the restricted regions. In all the simulations, the UAV is considered to have a minimum speed of $72\,km/h$, a maximum speed of $180\,km/h$, a maximum acceleration of $2\,m/s^2$, a maximum deceleration of $-2\,m/s^2$, and a turning rate of $10\,\text{deg}/s$ with a guidance and estimation update periods of 3 seconds. The first order transfer functions have time constants 0.5 and 0.01 for heading and speed responses of the UAV, respectively. Strategy design parameters f_r and $\psi^*_{HDC}(k)$ are selected to be 0.025 and $45\,\text{deg}$, respectively. Scheduling parameters for heading difference constraint V_{thres_1} and V_{thres_2} are $V_{\min}$ and $0.2V_{\min}$, respectively. The α-scheduling parameters α_{tg} and α_{temp} are 0.05 and 0.2, respectively. The proportional controller gains for the speed and position errors are 0.5 and 0.0004, respectively. The derivative gain, K_D, in (34) is 10 for States -1 and -5, and 60 in States -2 and -6.

The standard deviations of the noise added to the x and y positions of the target to obtain the measurements are selected to be $0.05km$ for Cases 1 and 2 and $0.1km$ for Case 3. The target position is measured during the time when the target is within a circle that is centered at the UAV and moves with it and whose radius r_s (see Fig.2) is equal to the sensor range. The position of the target is estimated from these measurements by employing a least-squares estimation technique with batch processing mode, based on a sliding window of measurements. Namely, only a specified number of measurements are stored and the measurement array is updated with new measurement by removing the oldest measurement and thus retaining the size of the array. Then, the kinematic equations are used to calculate the heading and speed of the target. The initial conditions, sensor and proximity ranges in each case are given in Table 2.

Case	Target			UAV				
	IP	IS	IH	IP	IS	IH	SR	PR
1	(7,-13)	80	97.4	(7,-13)	144	60	2.0	1.5
2	(2.85,-13)	144	23.0	(4,-12)	144	60	2.0	1.5
3	(4,-12)	80	57.3	(5,-11)	144	60	2.0	1.0

Table 2. Parameters: Initial Position (IP) [*km*], Speed (IS)[*km/h*], Heading (IH) [deg], Sensor Range (SR)[*km*], Proximity Range (PR) [*km*]

Fig.10 shows the first case where a target accelerates until it reaches its maximum speed and continues with this speed. As seen from Fig.10, target passes through three restricted regions. UAV, when guided by the algorithm, avoids these regions while continuing the pursuit of the target even if target gets outside the sensor range during the second and the third restricted regions.

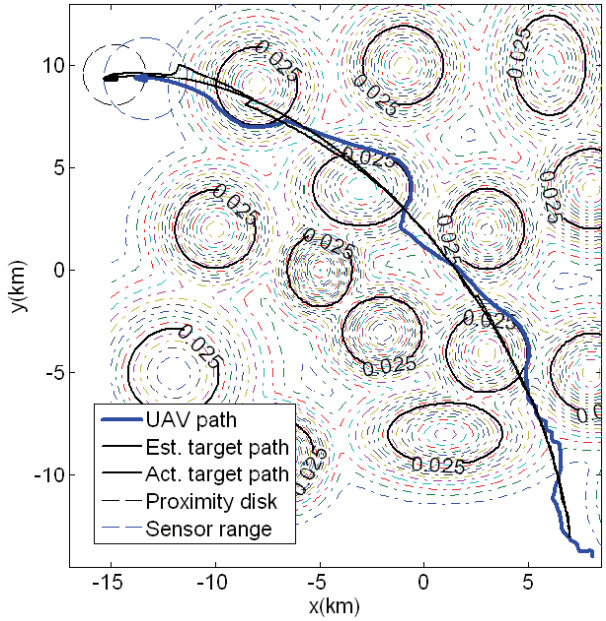


Figure 10. Case 1: UAV trajectory following an accelerating target

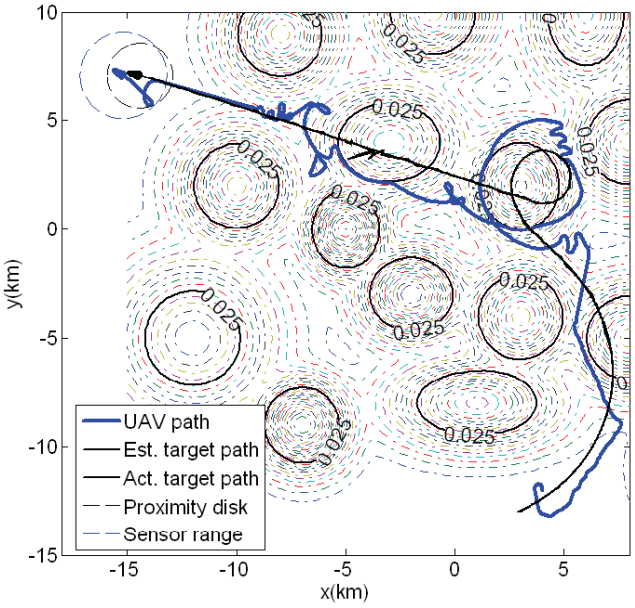


Figure 11. Case 2: UAV trajectory following a slowing target

Fig.11 shows the second case where the target, while moving on the path shown in Fig 11, reduces its speed to $36\ km/h$ in 600 seconds and then maintains it in the rest of the pursuit. Note that the speed of the target is much less than the minimum speed of the UAV during the most of the pursuit. Thus, the UAV reduces its speed to the minimum speed and tries to loiter within the proximity disk when there is no restricted area while minimizing the threat exposure level as shown in Fig.11. Note that there is no pre-defined loitering mode in algorithm. In fact, there is no need for the strategy to have a separate loitering mode because the strategy puts the UAV in loitering autonomously based on the speed of the proximity circle and the local PTEM within the proximity disk.

The third case is shown in Figs.12 and 13. In this case target continuously reduces its speed and stops at 660 seconds in a region where there is no restricted area. After staying in this region for 140 seconds, the target, at 800 seconds, starts accelerating and reaches its maximum speed at 1000 seconds. At 1100 seconds, it starts decelerating again and stops at 1360 seconds in a restricted region during the rest of the pursuit. This case shows the full capability of the strategy for autonomously putting the UAV in loitering mode both with and without restricted areas close-by. As seen from Fig.13a, when the target stops in a region where there is no close-by restricted region, the algorithm puts the UAV in loitering around the local minimum of the PTEM inside the proximity circle. This shows the benefit of utilizing the proximity-circle tangent-direction. Also note from Fig.13b that when the target stops the second time within a restricted area, a small portion of the proximity circle is still outside the restricted area. The algorithm loiters the UAV around this portion of the proximity circle. This shows the benefit of utilizing the pursuit guidance, with the right choice of the gain, based on the LOS angle and its derivative in the algorithm.

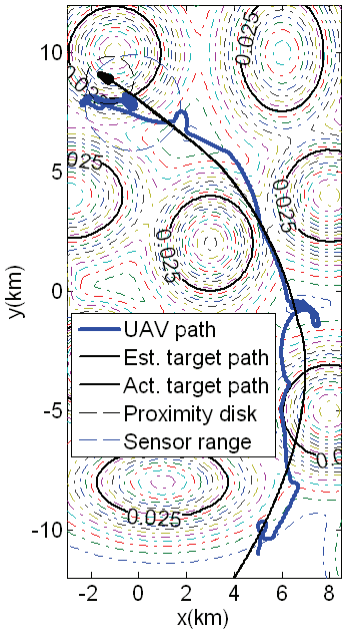


Figure 12. Case 3: UAV trajectory following a move-stop-move-stop target

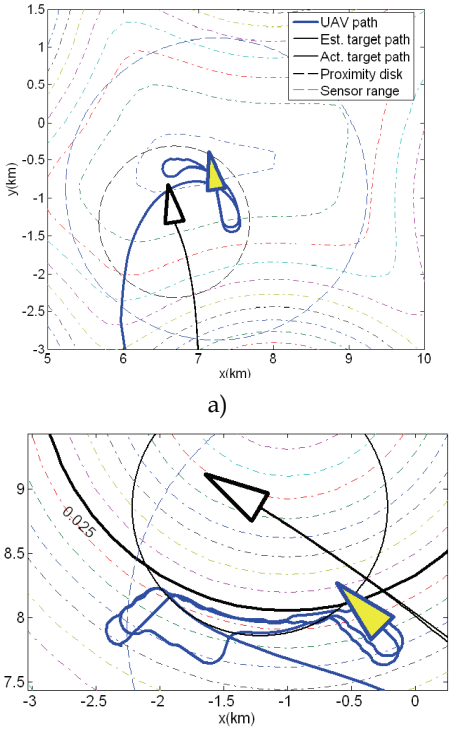


Figure 13. Case 3: When the target stops in a) non-restricted region and b) restricted region

4. Conclusion

A rule-based guidance strategy is developed for autonomous UAVs to track targets moving in an area with various types of threats, obstacles and restricted areas. The concept of PTEM (the Probabilistic Threat Exposure Map) is introduced as a mathematical formulation of the area of operation in terms of threats, obstacles and restricted areas. PTEM defines various types of threats, obstacles and restricted areas in a single framework that quantifies the threat exposure level as a function of position. A gradient search algorithm is applied on PTEM to determine the directions to avoid obstacles and restricted areas and to minimize threat exposure level. To keep the UAV within the proximity circle of the highly mobile target is an objective that is generally in conflict with the objectives of avoiding obstacles/restricted-areas and minimizing threat exposure. The rule-based guidance strategy is formulated to quantify the trade-off between these conflicting objectives and to generate the commanded heading and speed for the UAV. The rule-based intelligent decision approach has provided a very systematic method of developing the autonomous guidance strategy. First, the objectives of the guidance strategy and their priorities are determined. Then, based on the local threat information extracted from PTEM, position, heading and speed of the UAV relative to the target at a given time, the primary objective and/or the level of trade-off between the objectives are quantified. At the same time,

admissible ranges for the heading and speed are determined based on the dynamic and strategy-imposed constraints. This approach has facilitated the formulation of the guidance strategy that takes into account all the objectives of the mission with defined priorities and the constraints of the host UAV. On the other hand, utilization of the pursuit-guidance techniques based on LOS angle, proportional control for speed command and the weighted vectorial summation of minimizing direction and proximity-circle tangent has enabled the algorithm to perform better. In all simulation cases, guided by the algorithm, the UAV safely avoided restricted-areas/obstacles while continuing the pursuit of the target.

5. References

Rathbun, D.; Kragelund, S.; Pongpunwattana, A.; & Capozzi, B. (2003). An evolution based path planning algorithm for autonomous motion of a uav through uncertain environment, *Proceedings of the 22nd Digital Avionics Systems Conference*, Indianapolis, Indiana, Oct. 2003.

Finke, J. ; Passino, K. M. ; & Sparks, A. (2003). Cooperative control via task load balancing for networked uninhabited autonomous vehicles, *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, Dec. 2003.

Flint, M. ; Polycarpou, M. ; & Gaucherand, E. F. (2002). Cooperative control for multiple autonomous uav's searching for targets, *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 2823–2828, Las Vegas, Nevada, Dec. 2002.

Jun, M. ; Chaudry, A. I. & D'Andrea, R. (2002). The navigation of autonomous vehicles in uncertain dynamic environments: A case study, *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 3770–3775, Las Vegas, Nevada, Dec. 2002.

Pongpunwattana , A. & R. Rysdyk, (2004) Real-time planning for multiple autonomous vehicles in dynamic uncertain environments, *Journal of Aerospace Computing, Information, and Communication*, Vol. 1, Dec 2004, pp. 580–604.

Nikolas, I. K. ; Valavanis, K. P.; Tsourveloudis, N. C. & Kostaras, A. N. (2003). Evolutionary algorithm based offline/online path planner for uav navigation, *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, Vol. 33, No. 6, 2003, pp. 898–912.

Zhu, R.; Sun, D.; & Zhou, Z. (2005). Cooperation strategy of unmanned air vehicles for multitarget interception, *Journal of Guidance, Control and Dynamics*, Vol. 28, No. 5, 2005, pp. 1068–1072.

Waydo S. & Murray, R. M. (2003). Vehicle motion planning using stream functions, *Proceedings of the 2003 IEEE International Conference on Robotics&Automation*, pp. 2484–2491, Taipei, Taiwan, Sep. 2003.

Sengupta, R. ; Hedrick, J. K. & et al, (2003). Strategies of path planning for a uav to track a ground vehicle, *Proceedings of the Second Annual Symposium on Autonomous Intelligent Networks and Systems*, Menlo Park, CA, 2003.

Spry, S. C. ; Girard, A. R. & Hedrick, J. K. (2005). Convoy protection using multiple unmanned aerial vehicles:organization and coordination, *Proceedings of American Control Conference*, pp. 3524–3529, Portland, OR, June, 2005.

Jang, J. S. & Tomlin, C. J. (2005). Control strategies in multi-player pursuit and evasion game," *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibi*t, San Francisco, CA, Aug. 2005.

Antoniades, A. ; Kim, H. J. & Sastry, S. (2003). Pursuit-evasion strategies for teams of multiple agents with incomplete information, *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 756–61, Maui, Hawaii, Dec. 2003.

Vidal, R. ; Shakernia, O. ; Kim, H. J. ; Shim, D. H. & Sastry, S. (2002). Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation, *IEEE Transactions on Robotics and Automation*, Vol. 18, No.5, pp. 662–69, Oct. 2002.

Hespanha, J. P. ; Prandini, M. & Sastry, S. (2000). Probabilistic pursuit-evasion games:a one-step nash approach, *Proceedings of the 39th IEEE Conference on Decision and Control*, pp. 2272–2277, Sydney,Australia, Dec. 2000.

Hespanha, J. P. ; Kim, H. J. & Sastry, S. (1999). Multiple-agent probabilistic pursuit-evasion games, *Proceedings of the 38th IEEE Conference on Decision and Control*, pp. 2272–2277, Phoenix,AR, Dec. 1999.

Schumacher, C. (2005). Ground moving target engagement by cooperative uavs, *Proceedings of American Control Conference*, pp. 4502–4505, Portland, OR, June. 2005.

Sinha, A. ; Kirubarajan, T. & Bar-Shalom, Y. (2004). Optimal cooperative placement of gmti uavs for ground target tracking, *Proceedings of IEEE Aerospace Conference*, pp. 1859–1868, 2004.

Shea , P. J. & et al, (2000). Precision tracking of ground targets, *Proceedings of IEEE Aerospace Conference*, pp. 473–482, Vol. 3, 2000.

Koch , W. and Klemm, R. (2001). Ground target tracking with stap radar, *IEE Proc.-Radar, Sonar Navig.*, Vol. 148, pp. 173–185, June, 2001.

Jun , M. & D'Andrea, R. (2003). Probability map building of uncertain dynamic environments with indistinguishable obstacles, *Proceedings of the American Control Conference*, pp. 3417–3421, Denver, CO, June 2003.

Hespanha, J. P. ; Kizilocak, H. & Ateskan, Y. S. (2001). Probabilistic map building for aircraft-tracking radars, *Proceedings of the American Control Conference*, Arlington, VA, Jun. 2001.

Zengin , U. & Dogan, A. (2004). Dynamic target pursuit by uavs in probabilistic threat exposure map," *Proceedings of AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, Chicago, IL, Sep. 2004.

Dogan , A. & Zengin, U. (2006). Unmanned aerial vehicle dynamic-target pursuit by using a probabilistic threat exposure map, *AIAA Journal of Guidance, Dynamics and Control*, Vol. 29, No. 4, pp. 944–954, 2006.

Negnevitsky, M. (2002). *Artificial Intelligence: A Guide to Intelligent Systems*, Addison-Wesley, England.

Stark , H. & Woods, J. W. (1994). *Probability, Random Processes, and Estimation Theory for Engineers,* Uppper Saddle River, NJ : Prentice-Hall, Inc.

Konolige K. (1996). A gradient method for real-time robot control, *Proceedings of the International Conference on Intelligent Robots and Systems*, pp. 639-646, Takamatsu, Japan, Jan. 1996.

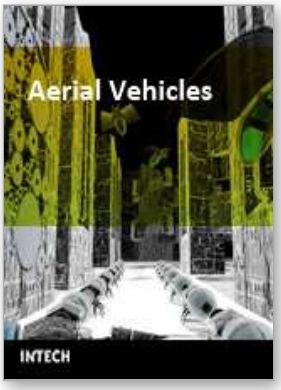
Choi, C. & Lee, J. (1996). Dynamical path planning algorithm of a mobile robot : Local minima problem and nonstationary environments, *Mechatronics*, Vol. 6, No. 1, pp. 81–100, June 1996.

Mitchell, I. M. & Sastry, S. (2003). Continuous path planning with multiple constraints, *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 5502–5507,Maui,Hawai, Dec. 2003.

Pastrick, H. L. ; Seltzer, S. M. & Warren, M. E. (1981). Guidance laws for short-range tactical missiles, *Journal of Guidance, Control and Dynamics*, Vol. 4, No. 2, pp. 98–108, 1981.

Ogren , P.; Fiorelli, E. & Leonard, N.E. (2004). Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment, IEEE *Transactions on Automatic Control*, Vol. 49, No. 8, pp. 1292–1302, 2004.

Larson, R.; Hostetler, R. P.; Edwards, B. H. & Heyd, D. E. (2002). *Calculus with Analytic Geometry,* Boston, NY: Houghton Mifflin Company.



Aerial Vehicles

Edited by Thanh Mung Lam

ISBN 978-953-7619-41-1
Hard cover, 320 pages
Publisher InTech
Published online 01, January, 2009
Published in print edition January, 2009

This book contains 35 chapters written by experts in developing techniques for making aerial vehicles more intelligent, more reliable, more flexible in use, and safer in operation.It will also serve as an inspiration for further improvement of the design and application of aeral vehicles. The advanced techniques and research described here may also be applicable to other high-tech areas such as robotics, avionics, vetronics, and space.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ugur Zengin and Atilla Dogan (2009). Autonomous Guidance of UAVs for Real-Time Target Tracking in Adversarial Environments, Aerial Vehicles, Thanh Mung Lam (Ed.), ISBN: 978-953-7619-41-1, InTech, Available from: http://www.intechopen.com/books/aerial_vehicles/autonomous_guidance_of_uavs_for_real-time_target_tracking_in_adversarial_environments

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821