Differential Geometric Path Planning of Multiple UAVs

Madhavan Shanmugavel
e-mail: m.shanmugavel@cranfield.ac.uk
Antonios Tsourdos
e-mail: a.tsourdos@cranfield.ac.uk
Brian A. White
e-mail: b.a.white@cranfield.ac.uk
Rafael Żbikowski
e-mail: r.zbikowski@cranfield.ac.uk

1 Introduction

Unmanned air vehicles of the future will be more autonomous than the remotely piloted reconnaissance platforms in use today. One of the open issues in their development is path planning. A path planning algorithm produces one or more safe, flyable paths for unmanned air vehicles (UAVs). The path has to be of minimal length, subject to a stealth constraint. As the UAV has limited range, the time spent over surveying territory should be minimized, so path length should always be a factor in the algorithm. Also, the path should be feasible for the aircraft to follow. The trajectory has to meet the speed and turn limits of the UAVs. The path-planning algorithm must also be compatible with the cooperative behavior envisioned for multiple UAVs. Finally, path-planning algorithms are expected to be coded in software that runs on an airborne processor. Thus, they must be computationally efficient and real time, enabling the UAV to replan its trajectory if needed.

Path planning, in general, connects any two points of interest by a curve, satisfying certain constraints. However, the algorithm differs as different approaches are used to generate the path. Chandler [1], in his approach, uses Voronoi diagram to produce polygonal paths connecting start and goal locations for each UAV by minimizing radar detection. Later, he refines the path by adding fillets of minimum turning radius. The time of arrival of each UAV and hence simultaneous arrival are coordinated by a high level manager based on the sensitivity function (cost versus time of arrival) sent by each UAV. A similar approach is adopted by Bortoff [2], where he uses an analogy of a chain connected by sequences of spring-mass-damper system to the UAV path. The ends of the chain are located at the initial and final configurations. The threats induce a repulsive force, which causes the masses in the chain to move away from the threats. However, this method involves complexity in solving ordinary differential equation (ODEs) with the curvature constraints. Also, accumulation of only a few masses around the threat location will lead to coarse path resolution, which is undesirable. McLain and Beard [3] extend the above approach by replacing the spring-damper system with rigid links between masses to eliminate sharp corners. However, this method does not guarantee that the resultant path is flyable by an UAV. Later, Judd and McLain [4] interpolated the Voronoi path with a series of cubic spline assigning a cost to each obstacle or threat position.

In other approaches, optimization techniques, such as probabilistic methods, mixed integer linear programming, and genetic programming, are used to produce paths by optimizing certain cost function. The cost functions differ based on the applications, such as minimum time arrival, optimizing fuel consumption, and coordinated attack. Jun and D’Andrea [5] propose a probabilistic method to path planning assuming positional uncertainty to threat regions. The final path is refined with circular arcs at the points of line intersection. Tom et al. [6], and Arthur and How [7] use mixed integer linear programming techniques for path planning to meet the path-planning constraints. These methods produce a safe route for UAVs to fly through, but these routes have to be smoothed further to make them flyable. Also, these optimization

Contributed by the Dynamic Systems, Measurement, and Control Division of ASME for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received April 3, 2006; final manuscript received January 16, 2007. Review conducted by Tal Shima.

Copyright © 2007 by ASME Transactions of the ASME

DOI: 10.1115/1.2767657

Keywords: path planning, trajectory generation, coordinated guidance, Dubins sets, pythagorean hodograph
methods are associated with high computational time. Achieving the mission objectives with physical and functional limitations of UAVs increases the complexity of solution to path-planning problem [8,9]. An overview of coordinated control of UAVs and their complexities can be found in Ref. [10].

The Voronoi diagram produces routes for the UAVs and the routes are then refined to make them flyable. Also, in the optimization approaches, the final outcome is a route plan, which satisfies certain constraints. If the route is refined by adding fillets, the resulted path is a series of lines and arcs, which is a subset of Dubins paths [11]. Instead of refining the path in later stage of path-planning approach, Shanmugavel et al. [12–14] use the Dubins set of curves and Pythagorean curves directly connecting the start and goal points. Though the path generation is straightforward, this approach uses iteration to satisfy the constraints. The convergence of this approach depends on how quickly the paths satisfy the constraints.

This paper presents the path planning of a group of UAVs for simultaneous arrival on target. The path planning is divided into two phases: (i) design of paths and (ii) planning of paths. In the first phase, flyable paths are produced using Dubins curves and Pythagorean hodograph (PH) curves. Both the Euclidean and differential geometric principles [15] of producing the Dubins paths are explained. Also, it is shown that the path length of the Dubins path is a function of its turning radii (curvature). Due to curvature continuity and rational properties, the PH curve is employed for path planning. The PH paths are tuned to curvature bounds to make them flyable. In the second phase, the flyable paths are adjusted to meet the safety constraints of the path planning, thus producing safe, flyable paths. Finally, the safe, flyable paths are adjusted to have equal in length to achieve simultaneous arrival for multiple UAVs. The novelty of the paper is in the (i) use of the Dubins and PH paths directly for initial path generation, (ii) design of the Dubins paths by basic principles of Euclidean and differential geometries, and (iii) tuning of the paths by varying the curvature.

The rest of the paper is organized as follows: Section 2 defines path planning for a single UAV and for multiple UAVs. Section 3 defines the problem of simultaneous arrival for a group of UAVs. The solution approach is outlined in Sec. 4 and this involves two steps: (i) design of flyable paths and (ii) planning of paths. The design phase begins with the Dubins path in Sec. 5. The properties and development of Pythagorean hodograph curves for path planning are described in Sec. 7. The safety constraints and production of paths of equal length are outlined in Sec. 8. The simulation results are discussed in Sec. 10. Finally, this paper ends with conclusions in Sec. 11.

2 Path Planning of Multiple Unmanned Air Vehicles

Path planning of a single UAV produces a feasible path whose end points are characterized by position (x, y) and orientation θ, together called a configuration P(x, y, θ). A feasible path is both flyable (meets kinematic and dynamic constraints) and safe to fly (no collisions). Provided with initial and terminal configurations P_i(x_i, y_i, θ_i) and P_f(x_f, y_f, θ_f), respectively, maximum curvature bound κ_max, and constraint Λ, the path planner generates a path r(t) with parameter t.

\[ P_i(x_i, y_i, θ_i) \rightarrow P_f(x_f, y_f, θ_f) \quad κ(t) < κ_{max} \text{ and } Λ \tag{1} \]

Extending the above equation (1) for a group of N UAVs,

\[ P_i(x_{i,x}, y_{i,y}, θ_{i,θ}) \rightarrow P_f(x_{f,x}, y_{f,y}, θ_{f,θ}) \quad κ_i(t) < κ_{max} \text{ and } Λ \tag{2} \]

where the suffix i represents the ith UAV, i=1, . . . , N and κ(t) is the curvature of the path.

The path r_i(t) in Eq. (2) is a curve and its properties change with t. Such a path is useful in predicting the future position and direction of the UAVs. Also, it helps the path planner to consider the kinematic limits at the early phase of the path planning. Also, the dynamics properties can be estimated by coupling the kinematic parameters with inertial properties of the UAVs.

3 Scenario and Problem Formulation

Consider N UAVs deployed for simultaneous arrival on target. All the UAVs leave the base at time t_{base} and have to reach the target at the time t_{target}, where t_{base} < t_{target}. The base and target are the two points of interest. Figure 1 shows the schematic of the mission. From the practical standpoint, it is necessary not only to consider the points of a feasible path as the center of mass of the UAV but to consider the size of the UAV and the space around it to establish safety margins imposed by the sensor range and maneuver capabilities of the UAV. Therefore, we assume that the center of two concentric circles represents the mass center of the UAV. The inner circle is the safety circle with radius R_s and the outer one is circmcircle with radius R_o. R_o ≥ (1/κ_{max}) > R_s. The radius of the circle circmcircle represents the sensor range. For any two UAVs, if the intersection of their safety circles is empty, safe flight path is ensured. The UAVs are assumed having equal capabilities and kinematical constraints and they are assumed to fly at a constant speed and at a constant altitude in a free space. Let the
configurations of $i$th UAV at the base and target, respectively, be
$P_i(x_{si}, y_{si}, \theta_{si})$ and $P_j(x_{sj}, y_{sj}, \theta_{sj})$. With the $k$ number of
constraints, the problem is formulated as

$$P_{si}(x_{si}, y_{si}, \theta_{si}) \rightarrow P_{sj}(x_{sj}, y_{sj}, \theta_{sj}) \quad \kappa(t) < \kappa_{i, \text{max}} \quad k$$

(3)

4 Solution Approach

The simultaneous arrival can be achieved either with constant speed
UAVs or with variable speed UAVs. In this paper, we consider constant speed
UAVs. As all the UAVs are flying at same constant speed, producing paths of equal
length ensures the simultaneous arrival. Accordingly, Eq. (3) changes into

$$P_{si}(x_{si}, y_{si}, \theta_{si}) \rightarrow P_{sj}(x_{sj}, y_{sj}, \theta_{sj}) \quad \kappa(t) < \kappa_{i, \text{max}}$$

(4)

where $x_j(t)$ and $y_j(t)$ are the path lengths of $i$th for $j$th UAVs, and
$i,j=1,\ldots,N$. The path length $s(t)$ of the path $r(t)=s(t),\gamma(t)$ is

$$s(t) = \int_{t_1}^{t_2} \sqrt{s'(t)^2 + y'(t)^2} \, dt \quad t \in [t_1,t_2]$$

(5)

where $s'(t)=dx/dt$ and $y'(t)=dy/dt$ are hodographs.

Equation (4) is solved sequentially as follows: First, a flyable
path is produced for each UAV by solving the equation

$$P_{si}(x_{si}, y_{si}, \theta_{si}) \rightarrow P_{sj}(x_{sj}, y_{sj}, \theta_{sj}) \quad \kappa(t) < \kappa_{i, \text{max}}$$

(6)

The flyable paths are Dubins and Pythagorean hodograph curves.
Then the flyable paths are adjusted to meet the safety conditions
by satisfying Eq. (3). Finally, the safe paths are made equal in
length by satisfying Eq. (4).

5 Producing Flyable Paths: Dubins

A Dubins path is the shortest path connecting two configurations
in a plane under the constraint of a bound on curvature [11].
In the plane, the line is the shortest distance between two
points and a circular arc is the shortest turn of constant curvature.
Combining these two provides the shortest path. The Dubins path is
formed either by concatenation of two circular arcs with their
common tangents or by three consecutive tangential circular arcs.
The former is CLC path and the latter is CCC path, where “C”
stands for circular arc and “L” stands for line segment. In this
paper, we are using the CLC paths for path planning and “Dubins
path” denotes a “CLC path.”

5.1 Design of Dubins Path Using Euclidean Geometry. In
Euclidean geometry, the Dubins path is designed by drawing
tangents between the circular arcs. Figure 2 shows the design of an
external tangent. The case of an internal tangent is analogous and
is omitted for brevity. Draw two circles of radii $p_i$ and $p_j$, respecti-
vately, at $O_j(x_{fj}, y_{fj})$ and $O_j(x_{fj}, y_{fj})$, where $p_i$ and $p_j$ are the start
and finish turn radii and $\kappa_i, \kappa_j$ are the maximum curvatures.
These two circles are called primary circles.

$$(x_{fj}, y_{fj}) = [x_j - p_i \cos(\theta_j \pm \pi/2), y_j - p_i \sin(\theta_j \pm \pi/2)]$$

(7a)

$$(x_{fj}, y_{fj}) = [x_j - p_j \cos(\theta_j \pm \pi/2), y_j - p_j \sin(\theta_j \pm \pi/2)]$$

(7b)

Draw a secondary circle of radius $p_i - p_j$ at $O_i$ and a perpendi-
cular $OT$ with $[O_i, O_j]$ as a hypotenuse such that $\Delta O_i O_j T$ forms a
right angle triangle, where $T$ is the tangent entry point on the
primary circle. Extend the line $O_i T$ to meet the primary circle
$C_T$ at $T_{EN}$. $T_{EN}$ is the tangent entry point on $C_i$. Draw a line
starting at $O_i$ and parallel to $O_i T_{EN}$ which meets the primary
circle $C_i$ at $T_{EX}$, $T_{EX}$ is the tangent exit point on $C_i$. The line
joining the points $T_{EX}$ and $T_{EN}$ is the external tangent between
the primary circles. Now, connect the start arc $P_i T_{EX}$, the line
$T_{EX} T_{EN}$, and the finish arc $T_{EN} P_f$. This is a Dubins path. In terms
of direction, the path is called a RSR, where $R$ stands for right
turn and $S$ stands for straight line.

From Fig. 2, $\angle(O_i O_j) = \phi$, $\angle(XO_i T_{EX}) = \phi_{en}$, $\angle(XO_j T_{EN})$
$= \phi_{en}$, and $\phi_{en} = \phi_{en}$. All the angles are assumed positive in
anticlockwise direction.

$$\phi = \arcsin \left( \frac{r_i - r_j}{c} \right)$$

(8)

where $c = \sqrt{(x_{ex} - x_{ei})^2 + (y_{ex} - y_{ei})^2}$.

The calculation of the tangent exit point $T_{EX}$ on $C_i$ and tangent
entry point $T_{EN}$ on $C_j$ is crucial in the design of Dubins paths.
Table 1 shows the formulas to calculate the tangent exit and entry
points for both RSR and LSR paths, where $L$ stands for left turn.
A similar procedure can be adopted to design RSL, and LSR paths
using internal tangents with the secondary circle of radius $|p_i$
$+ p_j|$. As there are two circles available (left and right turns) for a

Fig. 2 Dubins path

Transactions of the ASME

Received for review May 18, 2007. Accepted for
single configuration, four Dubins paths can be produced. A set of eight Dubins paths can be produced if either \( \theta_i \) or \( \theta_f \) is a free variable. If both the orientations are free variables, a set of sixteen paths can be produced. The shortest path can be selected from the set of available paths.

5.1.1 Existence of Dubins Paths. The tangent between the turning circles decides the existence of the Dubins path. The requirement on the external tangent is that the primary circles should not contain each other and on the internal tangent is that the primary circles should not intersect each other. These conditions can be written as functions of radii curvature of the start and finish turns and the central distance \( c \).

External tangent,

\[
(c + \rho_j) > \rho_f \quad \rho_f > \rho_i \tag{9a}
\]

Internal tangent,

\[
c > (\rho_i + \rho_f) \quad \rho_f > \rho_i \tag{9b}
\]

5.1.2 Length of the Dubins Paths. The path length is the summation of the lengths individual path pieces, two circular arcs, and their tangent. As the radii of curvature decide the existence of the tangent and in turn the existence of the Dubins path, the path length becomes a function of the radii. So, the lengths of any two Dubins path can be made equal by simply varying the radii of the turning arcs.

\[
L_{CLC} = \rho_i \alpha_i + L_i + \rho_f \alpha_f
\]

\[
L_{Dubins} = \int (\rho_i, \rho_f)
\]

where \( L_{Dubins} \) is length of the Dubins path, \( \alpha_i \) and \( \alpha_f \) are the included angles of the start and finish circular arcs, and \( L_i \) is the length of the tangent (\( ||T_{EX}, T_{EN}|| \)).

5.2 Design of Dubins Path Using Differential Geometry.

For a two dimensional maneuver, the initial and final tangent vectors are coplanar; hence the initial and final turning circles and the connecting tangent lie in a plane. A 2D Dubins path is shown in Fig. 3.

The sign of the initial and final maneuvers can be determined by designating either a left or right turn. Viewed from each position, a positive or negative rotation will define the sign of the curvature for each maneuver. Also, from the figure, we have

\[
e_j = [t_j, n_j] \tag{12}
\]

where \( \kappa_i \) is the curvature of the initial maneuver and

\[
r_j = e_j \left( 0 \pm 1/\kappa_j \right)
\]

\[
e_f = [t_f, n_f] \tag{13}
\]

where \( \kappa_f \) is the curvature of the final maneuver. The initial and final maneuver vectors \( t_j \) and \( t_f \) are related by

\[
t_f = R(\theta)t_i \tag{14}
\]

The connecting vectors \( a_i, a_j, \) and \( a_k \) form an orthogonal set of vectors. In order to determine the vectors, first define the connecting vector \( a_k \) as

\[
t_c = R(\theta)t_i \tag{15}
\]

where \( t_c \) is the basis vector defining the connecting vector. If the position of the final point \( p_f \) relative to the start position \( p_i \) is measured in start axes \( e_j \), we have

\[
p_f - p_i = e_j p \tag{17}
\]

Hence, the vector sum for the position vector in start axes is given by

\[
p = r_i - a_i + a_j + a_k - r_f
\]

The left hand side of this equation represents the vector connecting the centers of the turn circles. Hence,

\[
c t_i = -a_i + a_j + a_k \tag{19}
\]

where \( c \) is the length of the center vector. The remaining connecting vectors \( a_i, a_j, \) and \( a_k \) can be written in terms of the start basis vectors as

\[
a_i = R(\theta)^t \left( 0 \pm 1/\kappa_i \right)
\]

\[
a_j = R(\theta)^t \left( 0 \pm 1/\kappa_j \right) \tag{20}
\]

\[
a_k = R(\theta)^t \left( a \right) \tag{21}
\]

The center vector equation (19) now becomes

\[
c t_i = -R(\theta)^t \left( 0 \pm 1/\kappa_i \right) + R(\theta)^t \left( a \right) + R(\theta)^t \left( 0 \pm 1/\kappa_f \right)
\]

\[
= R(\theta)^t \left( a \pm 1/\kappa_j \pm 1/\kappa_f \right)
\]

This is a rotation equation; hence, the right hand vector must have the same magnitude as the left to give

\[
1 \left( a \pm 1/\kappa_j \pm 1/\kappa_f \right) = 1
\]
where

\[ \frac{a}{c} = \frac{1}{c^2} \left( \frac{\pm 1}{\kappa_1} - \frac{\pm 1}{\kappa_2} \right)^2 = 1 \]  
(22)

\[ \frac{a}{c} = 1 - \frac{1}{c^2} \left( \frac{\pm 1}{\kappa_1} - \frac{\pm 1}{\kappa_2} \right)^2 \]

This can be used to test for a feasible solution by

\[ 1 - \frac{1}{c^2} \left( \frac{\pm 1}{\kappa_1} - \frac{\pm 1}{\kappa_2} \right)^2 > 0 \]
(23)

In order to compute the rotation angle \( \phi_x \), the equation can be written in the form

\[ t_x = R(\theta_x) \]

\[ R(\theta_x) = \begin{pmatrix} \cos(\theta_x) & -\sin(\theta_x) \\ \sin(\theta_x) & \cos(\theta_x) \end{pmatrix} \]  
(24)

Solving for \( \phi_x \) gives

\[ \begin{pmatrix} \cos(\theta_x) \\ \sin(\theta_x) \end{pmatrix} = R(c, \kappa_1, \kappa_2) t_x \]  
(25)

where

\[ R(c, \kappa_1, \kappa_2) = \frac{1}{c} \left( \frac{\sqrt{c^2 - (\pm 1/\kappa_1 - \pm 1/\kappa_2)^2}}{\pm 1/\kappa_1 - \pm 1/\kappa_2} \right) \left( \frac{\sqrt{c^2 - (\pm 1/\kappa_1 - \pm 1/\kappa_2)^2}}{\pm 1/\kappa_1 - \pm 1/\kappa_2} \right) \]  
(26)

the final angle \( \phi_t \) can then be determined using

\[ \theta = \theta_x + \theta_t \]

\[ \theta_t = \theta - \theta_x \]
(27)

Thus, the curvature constraint is met by satisfying Eq. (6) and a flyable Dubins path is produced. The path length is calculated by summation of the arc lengths and the connecting tangent length,

\[ L = L_{arc} + L_{tangent} = \frac{\theta_1}{\kappa_1} + a + \frac{\theta_2}{\kappa_2} \]  
(28)

The equation of path length (28) is analogous to Eq. (10), and the condition for existence of the Dubins path (23) is analogous to Eqs. (9a) and (9b). However, the method derived by differential geometry is simple and easy to generalize, e.g., to polynomial curve, such as Pythagorean hodograph curve.

### 6 Dubins to Pythagorean Hodograph Path

The Dubins path is a composite curve of both lines and circles and is easy to produce. However, it lacks a smooth variation of curvature, that is, a sudden transition between the line of zero curvature and the circle of constant curvature, which is undesirable in practice. So, the use of the Dubins path is limited to piecewise smooth motion, possibly for a rotorcraft but not for a fixed wing UAV. Mathematically, the Dubins path provides only tangent continuity \( C^1 \). The curvature continuity is important as the curvature is proportional to the lateral acceleration of the UAV. A smooth motion needs curvature continuity \( C^2 \). Therefore, it is necessary to seek for an alternate path with curvature continuity. The equation of curvature is

\[ \kappa(t) = \frac{\ddot{r}}{r^2} = \frac{\ddot{r}^2}{|r|^2} \]  
(29)

From Eq. (29), the curvature is a function of first two derivatives of a curve, so the path needs to be at least twice continuously differentiable, that is, \( C^2 \) continuity. There are many polynomial curves that can provide \( C^2 \) continuity. However, we choose PH curve known for its rational properties.

### 7 Producing Flyable Path: Pythagorean Hodograph

A PH is a polynomial curve first introduced by Farouki [16,17]. Here, we give a brief introduction of the PH path with detail found in the papers by Farouki and associated references. In this paper, we use a fifth order PH curve as this is the lowest order curve which has inflexion points which can provide sufficient flexibility [16]. The PH path provides exact calculation of path length, its curvature, and the offset curve are rational. Substituting an appropriate polynomial \( \sigma(t) \) such that \( \sigma(t)^2 = \tilde{x}(t)^2 + \tilde{y}(t)^2 \) in Eq. (5) produces a path length \( s(t) \) and speed \( \dot{s}(t) \), which are reduced to an integral of the polynomial \( \sigma(t) \) and the polynomial itself, respectively. The offset curve \( r_a(t) \) is a rational curve, which is used to define the safety region around each UAV.

\[ \sigma^2(t) = \sqrt{\tilde{x}(t)^2 + \tilde{y}(t)^2} \]  
(30)

\[ s(t) = \int_{t_1}^{t_2} |\sigma(t)| dt \]  
(31)

\[ \dot{s}(t) = |\sigma(t)| \]  
(32)

\[ r_a(t) = r(t) \pm dN(t) \]  
(33)

where \( N(t) \) is unit normal of \( r(t) \) and \( d \) is a scalar variable.

From now on, a PH path refers to a quintic PH curve. The PH path is represented by the Bézier form with control points \( b_k = (x_k, y_k) \ k = 1,\ldots,5 \) as

\[ r(t) = \sum_{k=0}^{n} b_k (1-t)^{n-k} \]  
(34)

The configurations are interpreted as the boundary points for first order Hermite interpolation.

\[ b_0 = (x_0, y_0) \]  
(35a)

\[ b_0 = (x_0, y_0) \]  
(35b)

\[ d_0 = [\cos(\theta_0), \sin(\theta_0)] \]  
(35c)

\[ d_5 = [\cos(\theta_5), \sin(\theta_5)] \]  
(35d)

\[ b_1 = b_0 + (1/5)d_0 \]  
(35e)

\[ b_2 = b_1 - (1/5)d_5 \]  
(35f)

The control points \( (b_0, b_1, b_2, b_3) \) in Eq. (35) are fixed by configuration, while \( b_2 \) and \( b_3 \) are determined by satisfying the PH condition (31). From the result of four solutions \[18\], a minimum energy curve \[16\] is chosen.

As an original development, the PH curve only has the tangent continuity. So, it needs refinement for curvature continuity. As there is no closed form solution available \[19\], we seek iterative approach to meet the curvature bound. Bruyninckx \[19\] increased the tangent vectors by approximating the term \( \partial \kappa / \partial c \), where \( c_i \) is the magnitude of the tangent vector. However, we directly increasing the length of the tangent vector directly by modifying Eqs. (35c) and (35d) into

\[ d_0 = c_i [\cos(\theta_0), \sin(\theta_0)] \]  
(36a)
8 Flyable and Safe Paths

The flyable paths are produced by solving Eq. (6). The curvature at any point on the path shall not be greater than the maximum curvature bound allowed for each UAV.

\[ \kappa_i < \kappa_{\text{max}} \]  

where \( \kappa_i \) is the curvature of \( i \)th path and \( \kappa_{\text{max}} \) is the maximum curvature of the path.

The flyable path satisfies the principal constraint, maximum bound on curvature. However, safety of the UAVs needs additional constraints to be met to produce safe flyable paths. In this respect, the flyable paths have to meet the safety conditions: (i) minimum separation distance and (ii) nonintersection of paths at equal lengths.

8.1 Minimum Separation Distance. The minimum separation distance \( d_{\text{sep}} \) between any two UAVs should be at least equal to the summation of corresponding radii of the safety circles. For homogeneous UAVs, this will be twice the radius of the safety circle. The separation distance between a point \((x_i, y_i)\) on the \( k \)th path and another point \((x_j, y_j)\) on the \( l \)th path is

\[ d_{\text{sep}} = \sqrt{(y_j - y_i)^2 + (x_j - x_i)^2} \]

In the case of the PH path, the offset curves and the associated tube define the minimum separation distance or safety margin.

Figure 4 shows the schematic of both safety constraints. The non-overlapping of safety circles of any two paths meets the safety constraints. If the flyable paths meet Eq. (38), the paths are safe to fly and there is no need to replan the path. On failure of Eq. (38), the second condition (39) is tested on the paths. In the event of failure of both conditions, the replanning is done by increasing the radius of the turning circles or choosing the next shortest path from the Dubins set. In the case of the PH path, the lengths of the boundary tangent vectors are increased to meet the conditions.

8.2 Nonintersection of Paths at Equal Length. The minimum separation constraint is not imposed while producing the initial flyable path. If two flyable paths failed to meet this constraint, this does not necessarily imply a collision. To verify a collision, the path length of each path from its starting point to the point of failure is calculated. The difference between the path lengths must be less than the summation of corresponding radii of safety circles to confirm a potential collision.

Consider the \( k \)th and the \( l \)th paths to intersect each other at a point \( X \). The path length of the \( k \)th path from its start point to \( X \) is \( L_{k,\text{int}} \) and the path length of the \( l \)th path from its start point to \( X \) is \( L_{l,\text{int}} \) is calculated. The difference between the path lengths \( d_{\text{int}} = |L_{k,\text{int}} - L_{l,\text{int}}| \) must be at least equal to the summation of radii of safety circles of corresponding UAVs for a safe path. Hence,

\[ d_{\text{int}} \geq 2R \]  

\[ d_{\text{sep}} \geq 2R \]  

For a group of \( N \) UAVs, taking \( r \) UAVs at a time, the safety conditions have to be tested for \( n_r \) times, where

\[ n_r = \frac{N!}{r!(N-r)!} \]
9 Paths of Equal Length

The flyable and safe paths are required to be made equal in path length for simultaneous arrival. The variable speed UAVs can have difference in path lengths. But the constant speed UAVs has to have equal path lengths. This is achieved by increasing the shorter paths to that of the longest one. The path lengths of the flyable and safe paths are calculated using Eq. (24) or (28) for Dubins paths and Eq. (32) for PH path. For $N$ number of UAVs, with the length of each path $L_i$, the set of path lengths $L$ is

$$L = \{L_i\} \quad i = 1, \ldots, N$$

9.1 Reference Path. The longest of the safe flyable path is the reference path, that is, the maximum of $L$. The length of the reference path is thus

$$L_{\text{ref}} = \max(L)$$

where $L_{\text{ref}}$ is the length of the reference path.

9.2 Paths of Equal Length. The path lengths of $(N-1)$ UAVs are increased to that of the reference path. Lengths of the Dubins paths are increased by increasing the turn radii Eq. (10) or (28), while that of the PH path is achieved by increasing the length of boundary tangent vectors, Eqs. (36a) and (36b). As there is no closed form solution available, an iterative method is sought. This is implemented as follows:

Find $\kappa$, such that

$$L_j - L_{\text{ref}} = 0 \quad i = 1, \ldots, N - 1$$

Fig. 5 Shortest paths of UAVs

Fig. 6 Paths of equal length
10 Simulation Results

The proposed solution to the path planning is simulated with a group of UAVs, flying at constant speed and at constant altitude. The simulation results are discussed in this section for both the Dubins and the PH paths.

10.1 Trajectory Generations for Multiple Unmanned Air Vehicles Using Dubin’s Paths. Five UAVs are considered for simulation. The initial and final configurations are chosen randomly. The minimum turning radius is chosen as 1.2 units. The radius of safety circle is chosen as 2.5 units.
The shortest path of each UAV is calculated from the set of eight CLC paths. Thus, a set of five shortest paths is formed for five UAVs. Figure 5 shows the shortest paths of the UAVs. UAVs 1–5 follow the paths: \{LSR, LSL, RSR, LSL, RSL\}, respectively. All arcs are of minimum turning radius and the lengths of the paths are different.

The reference path is then found by using Eq. (42). The longest path from the set of shortest paths is the reference path. The path of UAV5 is the reference path as this is the longest in the set.

The bisection method is used to calculate the optimal radius of curvature of the shorter paths. This is because the solution to the equation of path length, Eqs. (5) or (28), is not unique and also the solution may have complex roots. The optimal radii of paths of UAV1, UAV2, UAV3, and UAV4 are \{9.82, 4.38, 9.86, and 14.42\} units, respectively. Figure 6 shows the paths of equal length. From the figure, it can be observed that the route of the paths is not the same as that of shortest CLC paths of the UAVs. The routes are \{LSL, LSL, RSL, LSL\}. The route of UAV1 and UAV3 are changed from \{LSR\} to \{LSL\} and \{RSR\} to \{RSL\}, respectively. This is because the original routes designed with minimum turning radii

![Figure 9 Path separations 9 and 10](image1.png)

![Figure 10 PH path of UAV1](image2.png)
did not meet the condition of existence of paths Eqs. (9a) and (9b) with modified radius of curvature. So, the next shortest path from the set of CLC paths was selected for finding optimal radius to produce the path of equal length to the reference path.

For any two paths, the safe flight path is ensured as follows: Using Eq. (38), the minimum separation of path is to be greater than two times the radius of safety circle. The minimum separation distance is verified by calculating the Euclidean distance between the paths. From these values, the maximum and minimum separations are found. The paths are safe to fly if they meet this condition. If the condition is not met at any point, the next condition Eq. (39) is verified, i.e., nonintersection of paths at equal distance. The length of each path to that point is calculated. If the difference between the lengths is greater than twice the radius of safety circle, the path is safe to fly. Otherwise, the path route must be changed to the next shortest path from the set of CLC paths.

Figure 7 shows the coupling of paths for UAV1 with other UAVs. The first figure (top left corner) shows two intersection of paths of UAV1 with UAV2. The minimum distance between the paths is 5.1 units. This just meets the condition of minimum separation. As the path increment is uniform with constant speed of UAVs, the UAV1 and UAV2 are safe to fly in the paths: {LSL} and {LSL}, respectively. The remaining paths UAV1 and UAV3, UAV1 and UAV4, and UAV1 and UAV5 are safe flight paths as they meet the minimum separation distance at all points along the paths.

Figure 8 shows the coupling of paths of UAV2 with UAV3, UAV4 and UAV5, and UAV3 with UAV4 (along row). The first two paths meet the minimum separation condition. Hence, they
are safe paths. The path of UAV2 traverses the path of UAV5. Hence, the UAV2 and UAV5 do not meet the minimum separation. The next condition, nonintersection at equal length is verified for these paths. The length of path of UAV2 to the point of minimum distance is 13.1 units and that of UAV5 is 20.71 units. The difference between the lengths is greater than two times the radius of safety circle; hence, this meets the condition for noncollision of UAVs and the paths are safe to fly. The coupling paths of UAV3 with UAV4 are well separated at corresponding points on the paths. The minimum distance between them is 10 units and are safe flight paths.

Figure 9 shows the coupling paths of UAV3 with UAV5 and UAV4 with UAV5. The paths are well separated meeting the minimum separation distance along the path. The paths are thus safe flight paths for UAV4 and UAV5. Thus, all paths are flyable, safe, and of equal length, ensuring the simultaneous arrival of the UAVs at the target.

10.2 Trajectory Generations for Multiple Unmanned Air Vehicles With Pythagorean Hodograph Paths. Three UAVs are considered in this case. The UAVs are named as UAV1, UAV2, and UAV3 and are flying at constant speed and at constant altitude. The initial and final configurations of the UAVs are well defined. All the UAVs leave the base at the same time. Figures 10–12 show the paths of the individual UAVs. The central path (solid line) shows the flight path. The dashed path on either side of the flight path shows the offset paths. The offset paths are generated at a distance given by the radius of the safety circle. Hence, the offset paths are defined by the diameter of the safety circle. The important points to be considered from the figures are as follows: (a) The paths have curvature continuity, thus providing smoothness. (b) Each path has a different route or trace.

The maximum curvature of the UAV $\kappa_{\text{max}}$ is taken as $\frac{\pm 1}{\rho}$. Hence, the path shall have to have the curvature not exceeding this value. The maximum and minimum curvatures of the paths are $(0.1142, -0.3000)$, $(0.3263, 0.002)$, and $(0.2718, 0.0055)$, respectively. The path of UAV2 is the reference path. The lengths of UAV1 and UAV3 are increased to that of UAV2 by the procedure using Eq. (36).

Figure 13 shows the paths of individual UAVs together. The start and finish points are shown together with the tangent circles, which define the maximum curvature of the UAVs. The safety conditions are tested for the three paths using Eq. (40). Taking any two UAVs at time, the total number of times the safe flight paths are to be tested are six.

Figures 14 and 15 are the paths of all UAVs with their offset paths. The UAV1 is not intersecting with other paths. The paths of UAV2 and UAV3 are intersecting at two points. The points of intersections of paths of UAV2 and UAV3 are found by the iterative method as mentioned in Sec. 10.2. The differences in the lengths of paths of UAV2 and UAV3 from their initial points are 8.381 and 7.321, respectively. The values are greater than the diameter of the safety circle. This ensures the safe flight paths.

11 Conclusions

Safe and simultaneous arrival of UAVs is planned using the Dubins and Pythagorean hodograph paths. The safe paths are ensured by meeting three constraints: (i) curvature constraint, (ii) minimum separation distance, and (iii) nonintersection of paths at
equal lengths. The simultaneous arrival is ensured by the design of paths of equal length. The offset path of the PH curve is used to generate a safety region along the path. The radius of this offset is taken as a constant throughout length of the path. However, it can be varied at any point on the curve.

This paper presents path planning for a group of UAVs for simultaneous arrival on target. This is achieved by meeting the objectives (e.g., shortest path, simultaneous arrival, safe flight) by varying the curvature of a path. The path planning is divided into two phases: (i) producing flyable paths and (ii) producing safe and flyable paths. Two flyable paths, the Dubins and the Pythagorean hodograph are used. The design of the Dubins path is shown both by principles of Euclidean and differential geometries. It is shown that the existence and length of the Dubins path are a function of curvatures of turning circles. The advantage of use of differential geometry to derive the Dubins path is demonstrated. Due to the undesirable transition of curvature of the Dubins path, we switch over to the Pythagorean hodograph path for its curvature continuity and rational properties. The offset paths of the PH curve are used to define the safety region around each path. In both Dubins and PH cases, the curvature of the path is tuned to meet the safety constraints. Finally, the lengths of the shorter paths are increased to that of the longest one by varying the curvature. The simulation is shown using the two locations. However, this can be extended into any number of locations.

The three main issues, convergence, computational time, and scalability, in the multiple UAV path planning were discussed in this paper. For the Dubins path, there is always a solution as long as the turning circles obey the conditions \( 9\alpha \) and \( 9\beta \) or \( 23 \). For the case of the PH path, the solution exists as long as the minimum energy path is available. However, there is a possibility that the tangent may increase without bound. This is controlled by maintaining the path length close to that of the Dubins. This also can be achieved by using composite PH paths with curvature continuity at the end points but unfortunately there is no closed form solution available for producing the flyable paths and satisfying the safety constraints. However, time for producing the flyable PH path is significantly less, and meeting the safety constraints consumes the most computation time. The effort in reducing this is planned to be achieved by use of composite paths. The simplicity of the proposed algorithms ensures that the computational time required for the coordinated guidance is implementable onboard. Finally, as it is shown in Eq. (40), the proposed algorithms are easily scalable and thus implementable for swarms of large numbers of UAVs.

References


