2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Path Planning for Multiple Unmanned Aerial Vehicles by Parameterized Cornu-Spirals

Ran Dai and John E. Cochran Jr.

Abstract—**In this paper, a group of cooperative planning paths for simultaneous starting and arriving Unmanned Aerial Vehicles (UAVs) are generated by parameterized Cornu-Spirals (CSs). The continuity and smoothness requirements for the designed flyable paths are achieved by the continuous curvature characteristics of CSs. The final curves are minimized in length with the least number of parameters representing the polynomial expression of the path curvature, while satisfying the maximum curvature constraints, equal length constraints, and collision avoidance constraints. The paths are integrated from initial points to final points by a trapezoidal integration algorithm. A nonlinear programming solver is used to calculate the optimized parameters. Simulation results for four simultaneous UAV paths are presented with designated initial and final positions and attitudes.**

I. Introduction

MUTIPLE Unmanned Aerial Vehicles (UAVs) are widely used in recent anti-terrorism activities and intelligence gathering to enhance the mission performance and maximize safety. The path planning strategies in hostile environments normally comprise two phases [1], [2]. The first phase is the Voronoi graph search which will generate polygonal graphs with optimized safety performance index. The second is to use the virtual forces emanated from the virtual field of each surveillance radar site to refine the generated Voronoi graphs. Although the virtual forces will reduce the vertices of the Voronoi polygonals, the curvature continuity of the refined graphs, which plays an important role in the stability of the UAVs turning maneuvers, does not meet the requirements of the continuously flyable path.

In order to generate a flyable path, many kinds of curves have been studied and designed for UAVs to accomplish their mission [3]-[5]. Dubins curves were first applied in robotics path planning that can move both forward and backward. This kind of circle-line-circle curve has a jump in curvature at the connection points between the circle and the line that will cause the robotics to stop at these connection points when traveling through the whole path. Other curves like Reeds and Shepp [6], [7] also have curvature discontinuities at their joint points. As an alternative choice, the composite clothoid-line-clothoid curves can be well designed with curvature being zero at the joint points to eliminate the discontinuities, but this kind of curve lacks flexibility in shape and does not leave enough space of change. Shanmugavel, et. al [3]-[5]. have proposed quintic Pythagorean Hodograph (PH) curves for a flyable path with ten parameters representing each curve. The PH curves are flexible in design and their curvatures are expressed in continuous polynomials. The parameter calculation of a PH curve is an iterative process in order to satisfy different constraints. Such kind of curves can be further simplified with fewer parameters and a more efficient optimization algorithm. These all leave room for improvement in the area of continuous curvature path planning.

The Cornu-Sprial (CS) [8], [9], also known as a clothoid or Euler's Spiral, has wide application in highway and railroad construction since it can be used to design gradual and smooth transitions in highway entrances or exits. Kelly and Nagy [10] used a parametric CS model to generate real-time, nonholonomic trajectories for robotics to minimize the terminal posture error. Here, we consider this CS model for use in UAV path planning and investigate how this parametric CS curve works under different constraints.

To generate flyable and safe paths for multiple UAVs with simultaneous starting and arriving time, the path constraints considered here include: (1) continuous curvature throughout its length which will ensure flyable path (2) maximum curvature corresponding to the lateral turn rate, (3) same flight length of each curve which will satisfy the simultaneous starting and arriving time condition under same flight speed and (4) minimum safety distance which will prevent the collision between the UAVs at equal lengths.

Unlike other path planning problems, including moving objects in finding the final path that normally results in motion planning or trajectory planning with system dynamics [11], the path considered here is in a static object environment without dynamic constraints. This paper follows the work of Shanmugavel, et. al. [3]-[5], by proposing a generalized CS curve along with simplified parameter identification procedure.

The following sections present the problem formulation in more detail followed by the introduction of the CS curve expression and properties. The different constraints and their mathematical expression are explained and the systematic solution using nonlinear programming (NLP) solver is presented. Finally, the simulation of four cooperative UAV paths with minimal length satisfying different constraints is calculated and presented.

R. Dai is with the Dynamics Research Inc., Torrance, CA 90501 USA (e-mail: rd@dynres.com).

J. E. Cochran, Jr., is with the Aerospace Engineering Department, Auburn University, Auburn, AL 36849 USA (e-mail: jcochran@eng.auburn.edu).

II. PROBLEM FORMULATION

There are N ($N \geq 2$) UAVs that are the same model and have the same capabilities starting from several bases simultaneously toward the desired target that is flying at constant equal speed. Each UAV is assumed to start and end at designated position (x, y) and heading angle θ specified as:

$$P_0 = (x_0, y_0, \theta_0) \\ P_f = (x_f, y_f, \theta_f) \tag{1}$$

These two points can be the starting and ending points in the planned path, they can also be treated as a pair of important way-points in a long term curve. In either case, the path designed to connect the points must be flyable and safe. Accordingly, the curvature of the planning path is required to be continuous and to not exceed a maximum bound on curvature $\kappa_{\max}$. Additionally, the minimum distance of any two paths among them is required to be larger than the safety radius R_s. Finally, the performance index to be optimized is the flight distance:

$$J = s_f - s_0 \tag{2}$$

III. CORNU-SPIRAL

The CS is a well known curve whose curvature is defined as a polynomial function of its arc length s as

$$\kappa(s) = \sum_{i=0}^{n} \alpha_i s^i \tag{3}$$

and the curvature is also the derivative of the curve angle θ with respect to the arc length:

$$\kappa(s) = d\theta / ds \tag{4}$$

The position of the points on this curve in Cartesian coordinates is calculated by the "Fresnel Integrals":

$$x(s) = \int_0^s \cos(\theta(s))ds \\ y(s) = \int_0^s \sin(\theta(s))ds \tag{5}$$

One advantage of using this curve in solving the path planning problem is the simplicity of calculation of its arc length:

$$s(t) = \int_0^t \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}\, dt = \int_0^t 1dt = t \tag{6}$$

The above expression are general, i.e., they apply to all forms of CS curves including special cases like straight line ($n = 0$, $\alpha_0 = 0$), circle ($n = 1$, $\alpha_1 = 0$) and unit CS ($n = 1$, $\alpha_0 = 0$, $\alpha_1 = 1$). In a unit CS, the curvature equals to the arc length and it will increase with the length. So the longer the curve, the more it will be curved. That is the origin of the term "spiral". The coefficients α_i affect the increase rate of the CS curve. In Fig. 1, all the CS curve curvature polynomials are in order $n = 1$ with $\alpha_0 = 0$, α_1 varied from 0.1 to 5. It is obvious that when transversing the same arc length, the larger the value of α_1, the greater the curvature increases. In addition, the order in the curvature expression also affects its shape. Fig.2 shows a series of CS curves with different order numbers in the curvature polynomials. The coefficients of the highest order are all set as $\alpha_n = 1$ with the others set as zero. It can be seen that curves with the higher order polynomials "curl" faster than others.

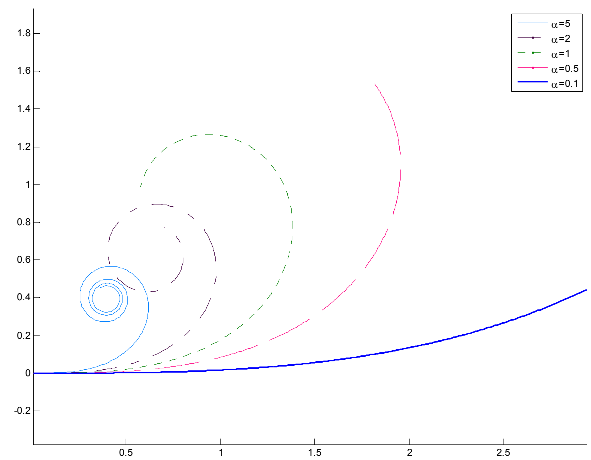

Fig. 1. CS curves of order $n = 1$ and $\alpha_0 = 0$ with coefficients α_1 changes from 0.1 to 5.

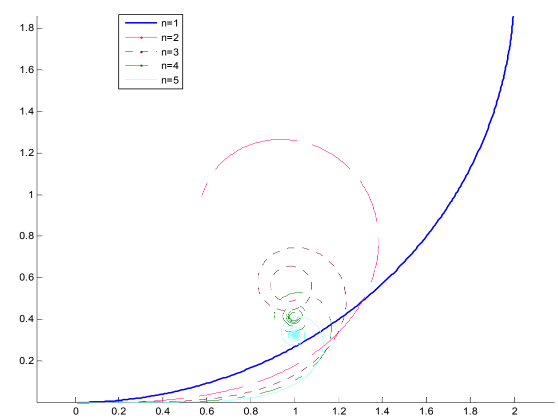

Fig. 2. CS curves of $\alpha_n = 1$ and other coefficients set as zero, order n changes from 1 to 5.

Combing the two types of effects that determine the shape of the CSs, we can produce an expression of the curvature polynomials to define any CSs possessing as many degrees of freedom as necessary to meet the required constraints. In the problem formulation stated above, the primary constraint for such curves is to meet the initial and final conditions which are classified as equality constraints. If the initial conditions $P_0 = (x_0, y_0, \theta_0)$ are incorporated as parameters in the CS curve, at least three additional parameters are required to meet the final boundary constraints. To have enough freedom to satisfy other inequality constraints and coordinate all the parameters to achieve the optimization purpose, one more parameter is required. Finally, four parameters are used in one CS curve. Three of them, a, b and c, form the cubic polynomials of the curve angle expression. The fourth one is the final arc length s_f. These parameters enter the problem formulation as follows:

$$\begin{aligned} \theta(s) &= \theta_0 + as + bs^2 + cs^3 \\ \kappa(s) &= \dot{\theta}(s) = a + 2bs + 3cs^2 \\ x(s) &= x_0 + \int_0^{s_f} \cos(\theta_0 + as + bs^2 + cs^3)ds \\ y(s) &= y_0 + \int_0^{s_f} \sin(\theta_0 + as + bs^2 + cs^3)ds \end{aligned} \tag{7}$$

Then the path planning problem has been cast as a parameter optimization problem with equality and inequality constraints.

IV. Flyable and Safe Path Constraints

A. Continuous Curvature Constraints

Continuous curvature is an important factor in designing a flyable path, because physically the curvature's path must be continuous and the rate of change of curvature with arc length is related to the directional command, a key input in maneuvering UAVs. Discontinuities in the curvature will cause jumps in the steering angle input. Unless the UAV reduces its speed and makes a stop to adjust the steering angle, such discontinuous input is not achievable. The continuous curvature will ensure smooth transitions in UAV autopilot. The curvature of a planar curve defined by $x(t)$ and $y(t)$ is

$$\kappa(t) = \frac{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)}{(\dot{x}(t)^2 + \dot{y}(t)^2)^{3/2}} \tag{8}$$

The CS curve specified in (7) with three parameters representing its polynomials can be expressed as

$$\kappa(s) = a + 2bs + 3cs^2 \tag{9}$$

Physically, the first and second derivatives of a curvature correspond to the lateral velocity and acceleration of a point (the vehicle) that is moving along the curve. Therefore, the curvature must be at least twice differentiable to meet the continuous velocity and acceleration requirement. The CS curve with curvature expression in (9) has second-order polynomials, is twice differential and can satisfy the continuity constraints.

B. Maximum Curvature Constraints

In CS expression of (7), the curvature is the only shape determining factor. By using the maximum curvature constraint, the kinematic acceleration of UAVs can be limited ensuring that the path designed can actually be traversed. Since the curvature is a parabolic function of the arc length, its maximum value points can only occur at one of three points: the initial point, the summit or bottom point, or the final point. Mathematically, the curvature values at those three points are

$$\begin{aligned} \kappa_0 &= a \\ \kappa_m &= a - \frac{b^2}{3c} \\ \kappa_f &= a + 2bs_f + 3cs_f^{\,2} \end{aligned} \tag{10}$$

If the curvature at none of these three points exceeds the maximum value, then the curvature throughout the whole length will suitably constrained.

C. Equal Length Constraint

Simultaneous arrival of UAVs may enhance the chance of success of a mission. When one or more UAVs have malfunctions or are ortherwise eliminated, it is desired that the others can continue to complete the mission. All the UAVs in this problem are essentially identical, i.e., they are same model with the same capabilities and fly at the same constant speed. So, when starting and arriving is performed simultaneously, the flight paths of all UAVs should have the same length. From the CS arc length calculation in (6), it is a simple matter to express the arc length L as

$$L = s_f - s_0 \tag{11}$$

When all the CSs start from the original point, the flight length is just s_f. As stated before, one CS curve is defined by four parameters a, b, c and t_f. If all the path lengths are equal to each other, they can share one parameter t_f. For N ($N \geq 2$) UAVs, the planning path of the N CSs will include $3N+1$ parameters like

$$p(a_1, b_1, c_1, \ldots, a_N, b_N, c_N, t_f) \tag{12}$$

Any additional UAV included in this path planning will only add three more parameters to this system. The path length t_f is the parameter they have in common.

D. Minimum Distance Constraints

Since the UAVs traverse their paths at the same rate and the paths are of equal length, to prevent collisions between any UAVs throughout the flight path, the minimum distance between any two planned paths should be larger than the safety separation distance R_s. Before the CS parameters are determined, it is difficult to find the minimum distance points between two curves. But since all the UAVs are flying at the same constant speed, at any equal length point, the distance between any two of curves should be greater than the specified safety separation distance so that the collision can be avoided. An approximate way to do this is to calculate the distance between any two paths at a series of equal length points. For example, in the ith and jth UAV paths, find m pair points at their equal lengths. For all the m pair points, if the constraints

$$d_g = \sqrt{(x_{ig} - x_{jg})^2 + (y_{ig} - y_{jg})^2} \ge R_s \quad (13)$$

are satisfied for all $i, j = 1, \ldots, N$, $i \ne j$ and $g = 1, \ldots, m$, collision can be avoided.

E. Boundary Condition Constraints

The initial and final conditions are the equality constraints in the system. All the paths designed by the CS curve start from the original points, so the initial condition is satisfied by setting the initial position and heading angle as parameters in the CS curve

$$\begin{aligned} x(0) &= x_0 \\ y(0) &= y_0 \\ \theta(0) &= \arctan \frac{\dot{y}(0)}{\dot{x}(0)} = \theta_0 \end{aligned} \quad (14)$$

The final point is determined by integrating using the trapezoidal rule (introduced in the next section) to satisfy the specified final boundary conditions.

At this point, all the constraints have been described. Some of them can be satisfied by the properties of the CS curve itself, others are expressed in corresponding equalities and inequalities. So the path planning problem is now a parameter optimization problem in which the parameters of the CS curvature expressions are identified while minimizing the path length with constraints.

V. DIRECT COLLOCATION AND NONLINEAR PROGRAMMING SOLVER

A. Trapezoidal Integration Rule

Since the CS curve is calculated from the "Fresnel Integrals", which have singularities, it is important to find an accurate and simple integration rule to find its coordinates at different arc lengths. Before introducing the trapezoidal integration rule, it is convenient to review the idea of direct collocation (DC) [12]. The basic idea of DC is to discretize the continuous solution to a problem represented by state and control variables by using linear interpolation to satisfy the differential equations. In this way, an optimal control problem (OCP) is transformed into a nonlinear programming problem (NLPP). Since solution to the OCP is in terms of an infinitely many values of state and control variables, DC is an approximation. For a first order differential function $\dot{x} = f(x,t)$, the DC method will divide it into n segments during the whole time interval such that: $t_0 = t_1 < t_2 < \cdots < t_{n+1} = t_f$, where t_0 is the initial time and t_f is the final time and the $n+1$ individual time points are called nodes. The value of the state vector at the i th node is represented by x_i. All the NLP variables can be written in one vector as:

$$x = \left[x_1, x_2, x_3 \cdots x_i, x_{i+1}, \cdots x_n, x_{n+1}, t_1, t_{n+1}\right] \quad (15)$$

The trapezoidal integration rule defines the defect vector of phase $i(= 1, \ldots, n)$ as:

$$d_i = x_{i+1} - x_i - \frac{h_i}{2}[f_i + f_{i+1}] \quad (16)$$

Here, $h_i = t_{i+1} - t_i$, f_i and f_{i+1} is the system first-order derivative function at node i and $i+1$. The defect vectors on each node are forced to equal to zero, so that the states at those nodes are constrained to satisfy the system equations.

The derivatives of the CS coordinates are trigonometric functions that can be easily calculated at known arc lengths. When the defect vector constraints are satisfied, the discrete points can approximately reproduce the CS curves. When each CS curve is discretized at equal step size h_i, the nodes will have equal length intervals which is convenient to calculate the distance between any two curves at equal length.

B. Nonlinear Programming Solver

The NLP solver used to solve the NLPP is SNOPT [13], which is based on a Sequential Quadrature Programming (SQP) algorithm. SNOPT can be used to solve problems like the following:

Minimize a performance index $J(x)$, subject to constraints on individual state and/or control variables:

$$x_L < x < x_U \quad (17)$$

constraints defined by linear combinations of state and/or control variables:

$$b_L < Ax < b_U \quad (18)$$

and/or constraints defined by nonlinear functions of state and/or control variables:

$$c_L < c(x) < c_U \quad (19)$$

With the above in mind, we can transfer the path planning problem into the following simple form:

Minimize $J = s_f$ with NLP variables set for each CS curve:

$$CS(i) = \left[x_{i,1}, x_{i,2}, \ldots, x_{i,n+1}, a_i, b_i, c_i, s_f\right] \quad (20)$$

where $i(= 1, \ldots, N)$, subject to the equality constraints including the defect vector constraints, $d_k = c_U = c_L = 0$ ($k = 1,2,\ldots n$), and boundary conditions, P_0 and P_f. These constraints will ensure all the coordinates on the CS curve will satisfy its differential function in trigonometric form. In addition, the inequality constraints to ensure the flyable and safe path can be set as:

$$-\kappa_{\max} \le a_i \le \kappa_{\max}$$
$$-\kappa_{\max} \le a_i - \frac{b_i^2}{3c_i} \le \kappa_{\max}$$
$$-\kappa_{\max} \le a_i + 2b_i s_f + 3c_i s_f^2 \le \kappa_{\max} \tag{21}$$
$$d_1 = \sqrt{(x_{i1} - x_{j1})^2 + (y_{i1} - y_{j1})^2} \ge R_s$$
$$\vdots$$
$$d_{n+1} = \sqrt{(x_{i,n+1} - x_{j,n+1})^2 + (y_{i,n+1} - y_{j,n+1})^2} \ge R_s$$

VI. SIMULATION RESULTS

Using the data provided in [3] for UAVs, a simulation of four UAVs starting and arriving with randomly selected positions and attitudes has been written and executed provided the safety radius from the starts and arrival points are satisfied. The data may be expressed as:

$$UAV1: P_i(8 \quad 6 \quad 12), \quad P_f(22 \quad 39 \quad 24)$$
$$UAV2: P_i(18 \quad 6 \quad 3), \quad P_f(32 \quad 39 \quad 113)$$
$$UAV3: P_i(28 \quad 6 \quad 74), \quad P_f(42 \quad 39 \quad 202)$$
$$UAV4: P_i(14 \quad 6 \quad 124), \quad P_f(27 \quad 39 \quad 120)$$

The maximum curvature is defined as $\kappa_{\max} = 1/3$ and the safety separation radii is $R_s = 3$. The objective is to find four CS curves with the same minimum length with second order polynomial expression for the curvature. Each path is discretized into 18 nodes with 3 curvature parameters and one parameter of the path length in common. So all together, there are 85 NLP variables. Unlike the method used in [3] that requires adjustment of the planning paths with shorter lengths to match the maximum one, the path length is the common parameter in each planned curve and the solution is achieved without iterative adjustment.

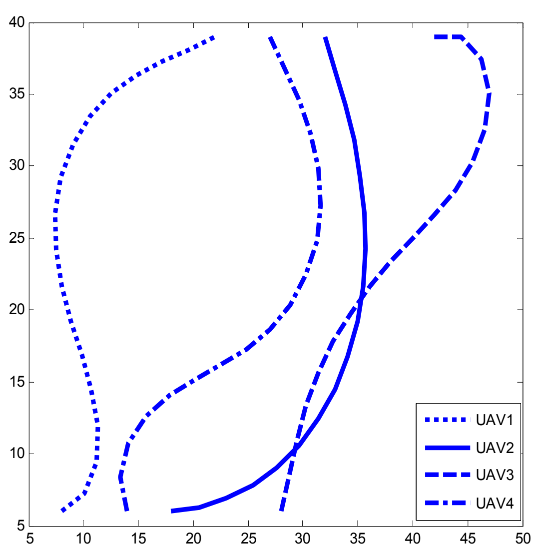

Fig. 3. Simulation Results of four UAVs planning paths starting and ending at designate positions and poses.

The simulation results including the planning paths, and their coordinate and curvature history with respect to the path length are shown in Fig.3 to Fig. 6, respectively. The minimum path length calculated here is 43.50 units with all the constraints satisfied. The optimal solution achieved here can be further improved with the introduction of higher order polynomial expressions for the path curvature which will possess more flexibility. But, since adding more parameters will increase the burden of calculation, there needs to be a compromise between them two.

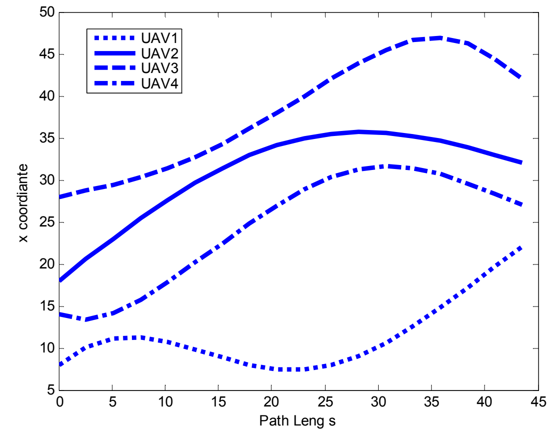

Fig. 4. Simulation Results of four UAVs x coordiante along the path length

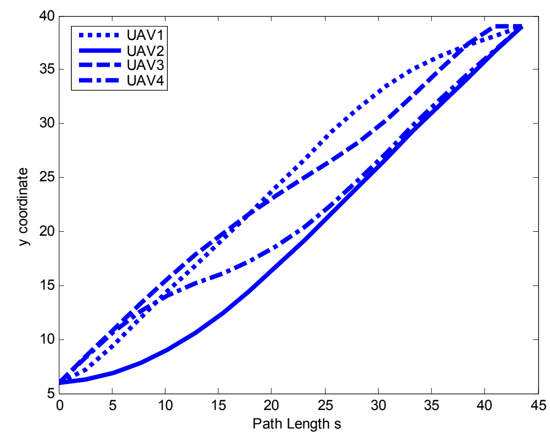

Fig. 5. Simulation Results of four UAVs y coordinate along the path length

VII. CONCLUSION

A group of simultaneous arriving UAV flying paths have been designed using the parameterized Cornu-Spirals. The planned paths have equal minimum length with continuous and bounded curvature. They can also prevent collision by satisfying the minimum distance constraints between any two paths. The algorithm includes all parameters as nonlinear programming variables in the nonlinear programming solver and gets the solution in one effort without tedious and iterative testing to adjust the coupled variables. This method is also reusable when more or less UAVs are included in the path planning. Without changing the algorithm, it is convenient and flexible to add or delete the same form of parameters in the new system to get the new solution.

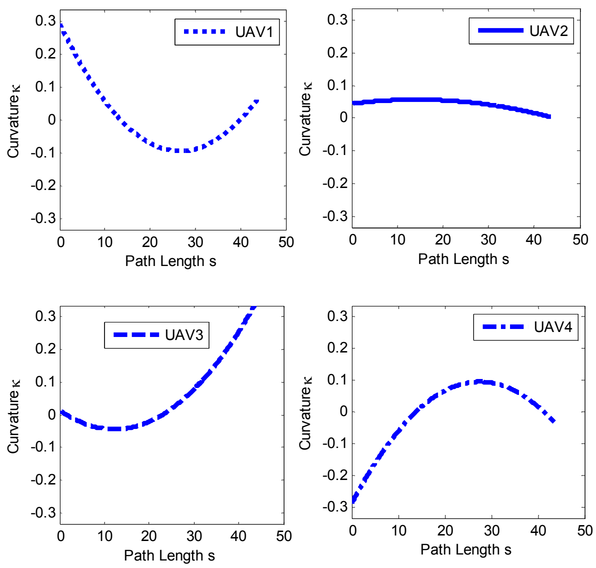

Fig. 6. Simulation Results of four UAVs curvature along the path length

Acknowledgment

The author wishes to thank Jim Boughton, Mike Van Auken and David Weir for their valuable suggestions and comments and other Dynamic Research Inc. staff members.

References

[1] T. W. McLain and R. W. Beard, "Trajectory Planning for Coordinated Rendezvous of Unmanned Air Vehicles", AIAA Guidance, Navigation and Control Conference, AIAA-2000-4369, August 2000.

[2] S. A. Bortoff, "Path Planning for UAVs," American Control Conference, pp. 364-368, June 2000.

[3] M. Shanmugavel, A. Tsourdos, R. Zbikowski, B. A. White, C.A. Rabbath and N.Lechevin, "A Solution to Simultaneous Aarrival of Multiple UAVs Using Pythagorean Hodograph Curves," American Control Conference, pp. 2813-2818, June 2006.

[4] M. Shanmugavel, A. Tsourdos, R. Zbikowski and B. A. White "3D Path Planning for Multiple UAVs Using Pythagorean Hodograph Curves," AIAA Guidance, Navigation and Control Conference, AIAA-2007-6455, August 2007.

[5] M. Shanmugavel, "Path Planning of Multiple Autonomous Vehicles," PhD Dissertation, Department of Aerospace, Power and Sensors, Cranfield University, May 2007.

[6] T. Fraichard, A. Scheuer and R. Desvigne, "From Reeds and Shepp's to Continuous-Curvature Paths," Proceedings of the International Conference on Advanced Robotics, pp. 585-590, October 1999.

[7] A. Scheuer and T. Fraichard, "Continuous-curvature path planning for car-like vehicles," Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems, vol. 2, pp. 997--1003, September 1997.

[8] J. Stoer, "Curving Fitting With Clothoidal Splines," *Journal of Research of the National Bureau of Standards.*, vol. 87, no. 4, pp. 317–346, July-August 1982.

[9] T. G. Davis, "Total Least-Squares Spiral Curve Fitting," Journal of Surveying Engineering, vol. 125, no. 4, pp. 159-175, Noverber 1999.

[10] A. Kelly and B. Nagy, "Reactive Nonholonomic Trajectory Generation via Parametric Optimal Control," The International Journal of Robotics Ressearch, vol. 22, no. 7-8 pp. 583-601, Jan. 2003.

[11] D. Hsu, J. C. Kindel, J. C. Latombe, and S. Rock, "Randomized kinodynamic motion planning with moving obstacles". In Proc. Workshop on Algorithmic Foundation of Robotics, 2000.

[12] C. R. Hargraves, and S. W. Paris, , "Direct Trajectory Optimization Using Nonlinear Programming and Collocation," J. of Guidance, Control and Dynamics, Vol. 10, No. 4, pp 338-342, July- August 1987.

[13] K. Holmstrom, A.O. Goran, and M. M. Edvall, "User's Guide For TOMLAB/SNOPT," Tomlab Optimization Inc., 2005, http://tomlab.biz.