Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Dubins Path Planning of Multiple UAVs for Tracking Contaminant Cloud

S. Subchan, B.A. White, A. Tsourdos
M. Shanmugavel, R. Żbikowski

Department of Aerospace, Power & Sensors, Cranfield University, Shrivenham, Swindon SN6 8LA, United Kingdom.

Abstract: : This paper presents cooperative path planning approach for multi UAVs to detect, model and track the shape of a contaminant cloud boundary. The objective of this research study is to manage the resources (airborne sensors) such as to determine the boundary of a cloud and track its motion with minimum information. The dispersing model of the contaminant cloud boundary is based on SCIPUFF. The UAVs are assumed to just have a sensor package which can sense nuclear, biological and chemical (NBC) contaminants. Therefore as a UAVs fly through a contaminant cloud the NBC sensors will recognise the entry and exit points of the UAVs from the contaminant cloud boundary and give these two points as measurements. Based on the measurements the splinegon approach uses to predict the contaminant cloud boundary and produces a segment for the next UAVs path.

1. INTRODUCTION

The advances of sensor technology and data fusion have broadened an interest in the use unmanned aerial vehicles (UAVs) especially with more information is available for decision making. This benefits of UAVs have been used either in civilian or military applications. Civilian applications involve grid searching for mining, oil exploration, surveillance and reconnaissance for traffic control, rescue missions, fire extinction, identifications of hazardous materials, oceanographic/geological surveys, marine and border inspection, see e.g. Swaroop (1999), Burns (2000) and Girard (2003). Military applications entail similar tasks such as target recognition, forward-deployed offensive missions, information gathering and communications, see e.g. Jeyaraman (2005). In security and defence unmanned aerial systems can provide significant reductions in manpower and risk to humans.

Other potential benefits entail cost efficiency, capability to minimize the risk to human life, ability to perform in hostile, hazardous and geometrically complex environments where direct human intervention is undesirable, enhanced detection capabilities based on novel sensor technologies and its ability to carry out long-time monotonous missions.

This paper focuses on the study of cooperative path planning for UAVs to track, detect and model the shape of a contaminant cloud using Dubins path planning. The threat of pollution from chemicals that are poisonous, odour less and opaque gases or due to biological agents or the threat of radiation due to industrial accidents or security challenges is high, the capability to monitor and track once such a cloud has been released is of utmost importance. This is because the rapid detection and geolocation of possible contaminant distribution is of paramount importance to prevent injury to the population.

2. PROBLEM DESCRIPTION

The main objective of this paper is that unmanned aerial system is able to sense and track a contaminant cloud. The scenario would be to use UAVs as a monitoring system launched upon suspicion or confirmation of the existence of a contaminant cloud. Once the UAVs have reached the contaminant cloud, they will need to check that it exists. If the UAVs sensor swarms follow a defined search pattern, when a UAVs detects the contaminant cloud it can send out a signal that it has detected the contaminant cloud. The UAVs can then centre on that location and begin to track. At this stage the UAVs needs to organise itself to be in positions that enable to sense and track it. Thus swarm control and guidance is a subset of the whole logistical problem of putting a suite of sensors into the right place at the right time to be able to provide situational awareness of contaminant cloud releases.

In this paper we propose an algorithm to track, estimate and reconstruct the boundary of the contaminant cloud. For this scenario we assume that there is a UAVs sensor swarm to take measurements of the air borne contaminant clouds. More over since we assume that the air borne contaminants are opaque gases there are no vision sensors to sense the contaminant cloud. The UAVs are assumed to just have a sensor package which can sense nuclear, biological and chemical (NBC) contaminants. Therefore as a UAVs flies through a contaminant cloud the NBC sensors will recognise the entry and exit points of the UAVs from the contaminant cloud boundary and give these two points as measurements. The definition of the cloud boundary will depend on the cloud composition. For a radiological cloud, this would be to sense the presence or absence of radioactivity. For a biological or chemical cloud, the concentration level at which the cloud is harmful would need to be determined. Similarly the UAVs in the sensor swarm will take measurements. But it should be noted that all the measurements will be

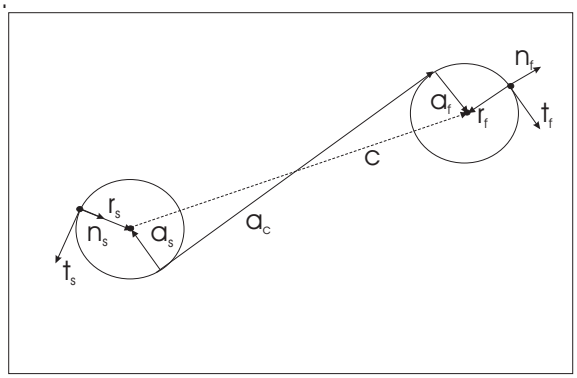

Fig. 1. Dubins Arc Geometry

taken asynchronously. Thus the task is to approximate a shape of the contaminant cloud from an emerging set of discrete location points obtained from the asynchronous measurements. The shape of the contaminant cloud will deform and will change shapes continuously and rapidly. This change in shape should not only be tracked, but if possible they should be predicted as well so that this prediction of the contaminated region can be scheduled to be visited by the UAVs in a systematic fashion i.e. these points will be given as way points to the path planning algorithm of the UAVs sensor swarms.

3. DUBINS PATH PLANNING

A Dubins path is the shortest path connecting two configurations in a plane under the constraint of a bound on curvature, see e.g. Shanmugavel (2007), Dubins (1957). 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. For a two dimensional manoeuvre, 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 Figure 1.

The sign of the initial and final manoeuvre 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 manoeuvre. Also, from the figure, we have:

$$\boldsymbol{r}_s = \boldsymbol{e}_s \begin{pmatrix} 0 \\ \frac{\pm 1}{\kappa_s} \end{pmatrix}, \quad \boldsymbol{e}_s = [\, \boldsymbol{t}_s \;\; \boldsymbol{n}_s \,] \tag{1}$$

where κ_s is the curvature of the initial manoeuvre and:

$$\boldsymbol{r}_f = \boldsymbol{e}_f \begin{pmatrix} 0 \\ \frac{\pm 1}{\kappa_f} \end{pmatrix}, \quad \boldsymbol{e}_f = [\, \boldsymbol{t}_f \;\; \boldsymbol{n}_f \,] \tag{2}$$

where κ_f is the curvature of the final manoeuvre. The initial and final manoeuvre vectors $\boldsymbol{t}_s$ and $\boldsymbol{t}_f$ are related by:

$$\boldsymbol{t}_f = \boldsymbol{R}(\theta)\boldsymbol{t}_s \tag{3}$$

where $\boldsymbol{R}(\theta)$ is the rotation matrix required to change the axis set from initial to final axes, see also (13) below. Hence, we have:

$$\cos(\theta) = \boldsymbol{t}_f' \boldsymbol{t}_s \tag{4}$$

The connecting vectors $\boldsymbol{a}_s$, $\boldsymbol{a}_f$ and $\boldsymbol{a}_c$ form an orthogonal set of vectors. In order to determine the vectors, first define the connecting vector $\boldsymbol{a}_c$ as:

$$\boldsymbol{t}_c = \boldsymbol{R}(\theta_s)\boldsymbol{t}_s \tag{5}$$

where $\boldsymbol{t}_c$ is the basis vector defining the connecting vector. If the position of the final point $\boldsymbol{p}_f$ relative to the start position $\boldsymbol{p}_s$ is measured in start axes $\boldsymbol{e}_s$, we have:

$$\boldsymbol{p}_f - \boldsymbol{p}_s = \boldsymbol{e}_s \boldsymbol{p}, \quad \boldsymbol{p} = \begin{pmatrix} p_t \\ p_n \end{pmatrix} \tag{6}$$

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

$$\boldsymbol{p} = \boldsymbol{r}_s - \boldsymbol{a}_s + \boldsymbol{a}_c + \boldsymbol{a}_f - \boldsymbol{r}_f \tag{7}$$

The left hand side of this equation represents the vector connecting the centres of the turn circles. Hence:

$$c\boldsymbol{t}_c = -\boldsymbol{a}_s + \boldsymbol{a}_c + \boldsymbol{a}_f \tag{8}$$

where c is the length of the centre vector. The remaining connecting vectors $\boldsymbol{a}_s$, $\boldsymbol{a}_f$ and $\boldsymbol{a}_c$ can be written in terms of the start basis vectors, as:

$$\boldsymbol{a}_s = \boldsymbol{R}(\theta_s)' \begin{pmatrix} 0 \\ \frac{\pm 1}{\kappa_s} \end{pmatrix}, \quad \boldsymbol{a}_f = \boldsymbol{R}(\theta_s)' \begin{pmatrix} 0 \\ \frac{\pm 1}{\kappa_f} \end{pmatrix}$$

$$\boldsymbol{a}_c = \boldsymbol{R}(\theta_s)' \begin{pmatrix} a \\ 0 \end{pmatrix} \tag{9}$$

The centre vector equation equation (8), now becomes:

$$c\boldsymbol{t}_c = \boldsymbol{R}(\theta_s)' \begin{pmatrix} a \\ \frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s} \end{pmatrix} \tag{10}$$

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

$$\left(\frac{a}{c}\right)^2 = 1 - \frac{1}{c^2}\left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right)^2 \tag{11}$$

This can be used to test for a feasible solution, by:

$$1 - \frac{1}{c^2}\left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right)^2 > 0 \tag{12}$$

In order to compute the rotation angle θ_s, the equation can be written in the form:

$$\boldsymbol{t}_c = \boldsymbol{R}(\theta_s)' \begin{pmatrix} \frac{\sqrt{c^2 - \left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right)^2}}{c} \\ \frac{\left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right)}{c} \end{pmatrix}$$

$$\boldsymbol{R}(\theta_s) = \begin{pmatrix} \cos(\theta_s) & -\sin(\theta_s) \\ \sin(\theta_s) & \cos(\theta_s) \end{pmatrix} \tag{13}$$

Solving for θ_s gives:

$$\begin{pmatrix} \cos(\theta_s) \\ \sin(\theta_s) \end{pmatrix} = \boldsymbol{R}(c, \kappa_s, \kappa_f)\boldsymbol{t}_c \tag{14}$$

where:

$$\boldsymbol{R}(c, \kappa_s, \kappa_f) = \frac{1}{c} \begin{pmatrix} \sqrt{c^2 - \left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right)^2} & -\left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right) \\ \left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right) & \sqrt{c^2 - \left(\frac{\pm 1}{\kappa_f} - \frac{\pm 1}{\kappa_s}\right)^2} \end{pmatrix} \tag{15}$$

The final angle θ_f can then be determined using:

$$\theta_f = \theta - \theta_s \tag{16}$$

Thus the curvature constraint is met by satisfying the equation

$$P_{si}(x_{si}, y_{si}, \theta_{si}) \xrightarrow{r_i(t)} P_{fi}(x_{fi}, y_{fi}, \theta_{fi}), \quad \kappa_i(t) < \kappa_{i,max} \tag{17}$$

The path length is calculated by summation of the arc lengths and the connecting tangent length.

$$L = L_{\text{arcs}} + L_{\text{tangent}} = \frac{\theta_s}{\kappa_s} + a + \frac{\theta_f}{\kappa_f} \tag{18}$$

4. CLOUD MODELLING USING SPLINEGONS

The contaminant cloud behaviour is complex. There are several ways to model its behaviour in a complex environment. Physical modelling of the cloud can be done using Gaussian dispersion models such as SCIPUFF, which predicts cloud behaviour using a statistical dispersion technique. Such techniques will have limited use for tracking as the cloud's behaviour must be expressed in a manner that allows the guidance of a group of UAVs equipped with suitable sensors to detect and track such a cloud. The requirement is for a cloud model that can be expressed in a compact format, thus enabling the exchange of a defining cloud dataset amongst the UAVs group with minimal communication overhead and with maximum utility in guidance algorithms. The UAVs are required to fly feasible paths that maximise the coverage of the cloud whilst avoiding local obstacles. The flight paths of the UAVs will be dictated by Dubins or related paths. These are defined using circular arcs and straight line segments. This suggests that the appropriate strategy should be to fly through the cloud and note the entry and exit points rather than transition the cloud boundary. This is further supported by the fact that the cloud density will not have a physical boundary, but will have a density profile that asymptotically approaches zero. Hence the sensor will produce a threshold transition rather than a physical boundary. There will also be density variations across the cloud that can be detected by transition rather than by boundary following. Finally, transitioning the cloud will enable the UAVs to perform search operations outside the detect cloud boundary to ensure that maximum use of the area coverage is maintained.

4.1 Splinegon Construction

A splinegon with constant curvature line segments can be defined with C^2 contact at the vertices. This implies that the line segments share both a common vertex and that the tangents at the vertices are also the same. In order to ensure C^2 contact between vertices, the line segments must meet both position and tangent end point constraints. A single arc segment between vertices only has one degree of freedom: the arc curvature. This is not enough to be able to match the tangent constraint at both end vertices, as at least two degrees of freedom are necessary. Extra degrees of freedom are thus required to ensure the C2 constraints at both line segment end vertices can be met. One solution to increasing the degrees of freedom is to introduce an intermediate vertex such that the line segment is replaced by two arc segments of different curvature, as shown in Figure 2. The entry and exit vertices identified by the UAV transitions are shown by the symbol "x" and the intermediate vertex is shown by the symbol "o" in Figure 6. Hence two arcs of differing curvature will connect the UAV vertices via the intermediate vertex. In order to develop the defining equations for such a solution, consider the intersection of two constant curvature arcs at a point with C^2 contact. Consider all tangent vectors in Figure 3, only the intermediate tangent vector $\boldsymbol{t}_I$ is unknown. Hence

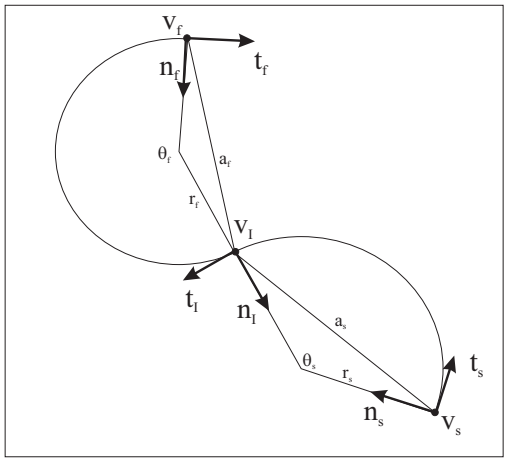

Fig. 2. Arc segment with C^2 contact intermediate vertex

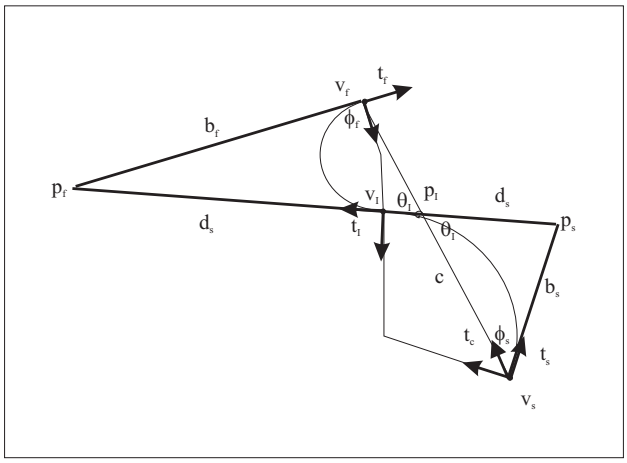

Fig. 3. Arc segment with C^2 contact intermediate vertex

$$[\,-\boldsymbol{t}_c \;\; \boldsymbol{t}_I \;\; \boldsymbol{t}_s \;\; \boldsymbol{t}_f\,] \begin{bmatrix} c \\ b_s + b_f \\ b_s \\ b_f \end{bmatrix} = 0$$

$$\boldsymbol{T}\boldsymbol{s} = 0 \tag{19}$$

Hence the solution vector lengths lie in the right null space of

$$\boldsymbol{T}\boldsymbol{N} = 0 \tag{20}$$

Thus a family of solution exist for a variety of $\boldsymbol{t}_I$ vector. In order to explore the bounds for feasible solutions, consider again Figure 3. Using the sin rule on the two triangles $[\boldsymbol{v}_s, \boldsymbol{p}_s, \boldsymbol{p}_I]$ and $[\boldsymbol{v}_f, \boldsymbol{p}_f, \boldsymbol{p}_I]$ yields:

$$\begin{aligned} \frac{\sin\phi_s}{d_s} &= \frac{\sin\theta_s}{b_s} \\ \frac{\sin\phi_f}{d_f} &= \frac{\sin\theta_f}{b_f} \end{aligned} \tag{21}$$

Now the vector $\boldsymbol{p}_f - \boldsymbol{p}_s$, along which the tangent $\boldsymbol{t}_I$ lies, must have length $b_s + b_f$ for a feasible solution and hence

$$d_s + d_f = b_s + b_f \tag{22}$$

Substituting for d_s and d_f from equation (21) yields:

$$\begin{aligned} b_s + b_f &= b_s \frac{\sin\phi_s}{\sin\theta_I} + b_f \frac{\sin\theta_f}{\sin\theta_I} \\ \sin\theta_I &= \bar{b}_s \sin\phi_s + \bar{b}_f \sin\phi_f \\ \bar{b}_s &= \frac{b_s}{b_s + b_f} \\ \bar{b}_f &= \frac{b_f}{b_s + b_f} \end{aligned} \tag{23}$$

Now

$$\begin{aligned} \bar{b}_s + \bar{b}_f &= 1 \\ \bar{b}_f &= 1 - \bar{b}_s, \qquad 0 < \bar{b}_s < 1 \end{aligned} \tag{24}$$

Hence a range solutions exist from $\bar{b}_s = 0$ to $\bar{b}_s = 1$. Thus the solution range is:

$$\begin{aligned} \sin\theta_I &= \bar{b}_s \sin\phi_s + (1 - \bar{b}_s)\sin\phi_f \\ &= \sin\phi_s, \bar{b}_s = 1 \\ &= \sin\phi_f, \bar{b}_s = 0 \end{aligned} \tag{25}$$

These parameters will give the requisite information to solve the arc parameters in Figure 2. Hence

$$\begin{aligned} \boldsymbol{a}_s &= b_s \boldsymbol{t}_s + b_s \boldsymbol{t}_I \\ a_s &= \mid \boldsymbol{a}_s \mid \\ \sin\theta_s &= \mid \boldsymbol{t}_s \quad \boldsymbol{t}_I \mid \\ \cos\theta_s &= \boldsymbol{t'}_s \boldsymbol{t}_I \\ \kappa_s &= \frac{2\sin(\theta_s/2)}{a_s}, \quad a_s \neq 0 \\ &= 0, \quad a_s = 0 \\ \boldsymbol{a}_f &= b_f \boldsymbol{t}_f + b_f \boldsymbol{t}_I \\ a_f &= \mid \boldsymbol{a}_f \mid \\ \sin\theta_f &= \mid \boldsymbol{t}_I \quad \boldsymbol{t}_f \mid \\ \cos\theta_f &= \boldsymbol{t'}_I \boldsymbol{t}_f \\ \kappa_f &= \frac{2\sin(\theta_f/2)}{a_f}, \quad a_f \neq 0 \\ &= 0, \quad a_f = 0 \\ \boldsymbol{v}_I &= \boldsymbol{v}_s + \boldsymbol{a}_s \end{aligned} \tag{26}$$

4.2 Solution Selection

There are many different selection mechanisms for the choice of a solution over the range of solution given by equation 25. The initial choice of solution for the cloud tracking containment is determined by considering the difference in absolute curvature κ_d, where:

$$\kappa_d = \min(\parallel \kappa_s \mid - \mid \kappa_f \parallel) \tag{27}$$

If this is minimised, the two arcs will be balanced and the intermediate vertex will lie close to the middle of the two UAV vertices. Extreme solutions will result in a high curvature short segment and a low curvature long segment which will place the intermediate vertex close to the vertex connected to the high curvature segment. Choosing a contour with minimum changes in curvature between segments will produce splinegons with less abrupt direction changes. Using this approach, the contaminant cloud can be adequately modelled by a splinegon.

5. MISSION SCENARIO AND SIMULATION

Consider a mission in which two UAVs team are tracking a contaminant cloud. The UAVs are assumed to be homogenous in their physical capabilities and flying at constant speed at constant altitude. The task is to sense and track a contaminant cloud while coordinating the data measurement to approximate the shape of the cloud.

The representative contaminant distribution is based on the SCIPUFF, see Sykes (1996). This shows contours of constant density at discrete levels set by SCIPUFF. Assuming the UAVs sensor is set to the lowest level, the description must describe the outer sensor threshold contour in sufficient detail to contain it and track it.

In this paper an approximate shape of the contaminant cloud is formed based on a set of point measurements using the splinegons. In order to track the cloud, the simplest approach would be just to form a shape of the cloud using splinegons every time a measurement arrives. This would involve sending the UAVs sensor swarms in random search paths in order to verify the presence or absence of the contaminant particles. The description will take into account the fact that the UAVs will fly through the contaminant cloud and hence will detect an entry and exit point on each transition. This suggests that the most efficient modeling approach should be to define these points as vertices and form a polygon with line segments. This raises an issue as to how to represent the curved nature of the density contour. One such approach is to use a generalisation of polygons to produce a set of vertices that are connected by line segments of constant curvature. This is a subset of a class of object named as splinegons, see e.g. Dobkin (1988), Dobkin (1990).

5.1 Algorithm and implementation

The algorithm detail for tracking contaminant cloud boundary is shown in Figure 4. The detail implementation of the algorithm can be given as follows. First, a waypoint must be defined for the UAVs sensor swarms to track the cloud. In Figure 5, the first UAVs (UAV-A) passes the waypoint ($t_0 - t_1$, magenta) and takes a measurement for the entry and exit points. Once the first two sets of measurements have been taken by the UAVs, the shape of the cloud at the first two measurement instances are approximately formed using the splinegons. The tangent and normal velocity for each of the vertices in the splinegon are then computed. The centre point of the splinegons is also computed, which is then used to compute the velocity at the centre of the splinegon. Based on the highest order of the curvature the new four vertices are introduced for the next waypoint. UAV-A broadcasts the first two vertices to the second UAVs (UAV-B). While UAV-B passes the cloud ($t_1 - t_2$, black) and broadcasts the path length, UAV-A goes to the next segment using Dubins path (blue-red-magenta) and extended path (cyan) to arrive at t_2 second. This time constraint is introduced to avoid collision during

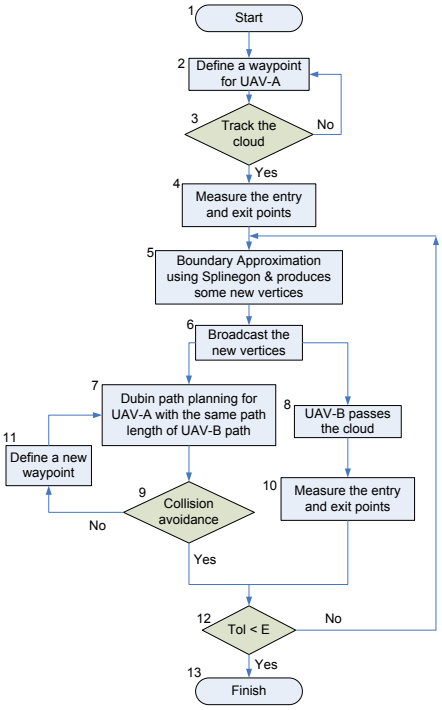

Fig. 4. Algorithm for tracking contaminant cloud.

a mission and to make sure only one UAVs passes the contaminant cloud.

Figure 6 shows UAV-B passes the cloud (t_1-t_2, magenta) and takes a measurement for the entry and exit points. Again the boundary approximation is performed by splinegons with two sets more of measurements. The two new vertices for the next waypoint are introduced for UAV-B instead of given to UAV-A. UAV-A broadcasts the path length ($t_2 - t_3$, blue) while UAV-B passes the Dubins path (blue-red-magenta) to the next segment.

Figure 7 shows UAV-A passes the cloud (t_2-t_3, magenta) and takes a measurement for the entry and exit points. The splinegons is used to approximate the cloud boundary. The two new vertices for the next waypoint are introduced. At ($t_3 - t_4$) UAV-B passes the contaminant cloud (blue) while UAV-A is passing the Dubins path (blue-red-magenta) and extended path (cyan). Similarly, Figure 8-10 show the transition of UAVs sensor swarms path for tracking contaminant cloud.

In Figure 11 the Dubins path will collide with the obstacle, therefore a new waypoint is introduced to avoid collision to the building, see Figure 12 for the corrected path. In this implementation the contaminant cloud boundary approximated by 16 vertices. It can be seen in Figure 12 that the approximated shape is good enough to compare with the SCIPUFF result.

6. CONCLUSION

In this paper, some computational results of tracking contaminant cloud boundary using UAVs path planning have been presented. The approach based on the combination

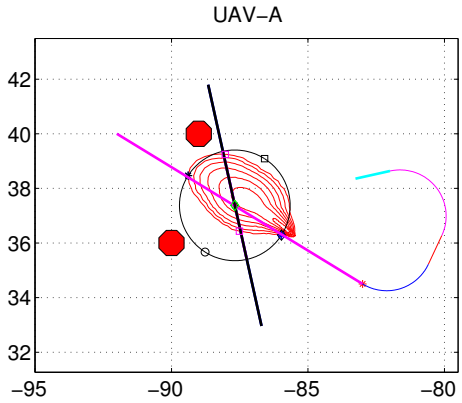

Fig. 5. First transition.

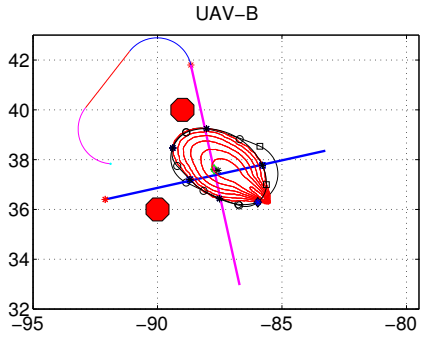

Fig. 6. Second transition

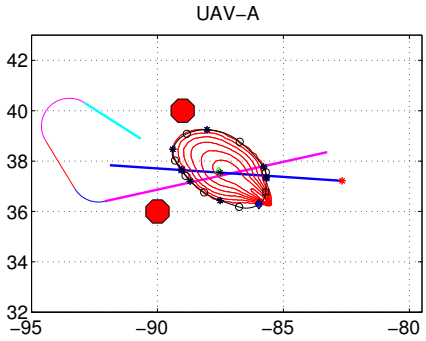

Fig. 7. Third transition

of Dubins path planning and tracking the evolving contaminant boundary using splinegons. The path planning solutions compose of segments which are produced by splinegons and the Dubins path to join both segments.

Splinegons are a generalisation of polygons which produce a set of vertices that are connected by line segments of constant curvature. Splinegons are used because the requirement is to model the contaminant cloud using a compact format there by enabling exchange of cloud datasets amongst the UAVs sensor swarms with minimal communication overhead and with maximum utility in guidance algorithms.

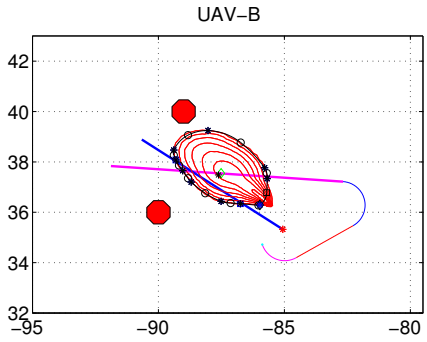

Fig. 8. Fourth transition

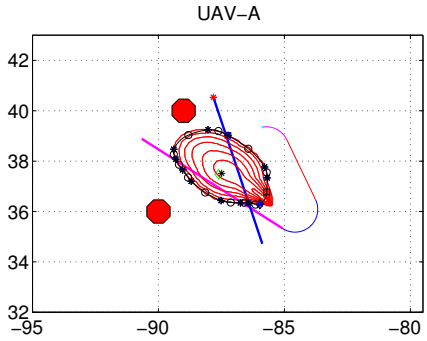

Fig. 9. Fifth transition

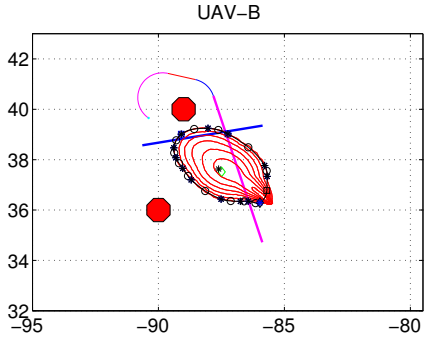

Fig. 10. Sixth transition

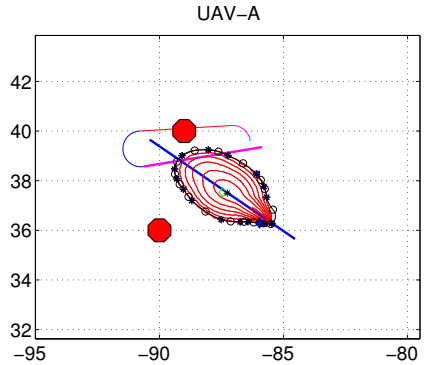

Fig. 11. Seventh transition, where the Dubins path must be corrected before the UAVs collides to the obstacle

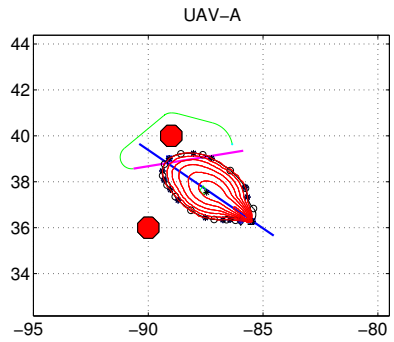

Fig. 12. Seventh transition with a new waypoint to avoid collision

The design of Dubins path is shown in both by principle of Euclidean and differential geometries. It is shown that the existence and length of the Dubins path are function of curvature of turning circle. The mission scenario is that only one UAVs passes the contaminant cloud to avoid collision. Therefore the Dubins path length constraints is proposed to guaranteed this mission fulfill.

REFERENCES

D. Swaroop and J. K. Hedrick. Constant spacing strategies for platooning in automated highway systems. *ASME Journal of Dynamical Systems, Measurement and Control*, 129: pages 462-470, 1999.

R. Burns and C. A. McLaughlin and J. Leitner and M. Martin. Tech 21: Formation design, control and simulation. Proceedings of the IEEE Aerospace Conference, pages 19-25, 2000.

A. R. Girard and J. B. Sousa and J. K. Hedrick. An overview of emerging results in networked multi-vehicle systems. Proceedings of the 40th IEEE Conference on Decision and Control, pages 1485-1490, 2003.

S. Jeyaraman and A. Tsourdos and R. Żbikowski and B. A. White. Kripke modelling of multiple robots with de-centralised co-operation specified with temporal logic. Journal of Systems and Control Engineering, Proceedings of the IMechE: Part I, page 15-31, 2005.

R. I. Sykes and D. S. Henn and S. F. Parker and R. S. Gabruk. SCIPUFF - A generalized hazard dispersion model. Ninth Joint Conference on the Applications of Air Pollution Meteorology with A&WMA, 1996.

L. E. Dubins On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. American Journal of Mathematics, 79, pages 497-516, 1957.

M. Shanmugavel and A. Tsourdos and B. A. White and R. Żbikowski. Differential geometric path planning of multiple UAVs. Journal of Dynamic sytems, Measurement and Control, 129, pages 620-632, 2007.

D. P. Dobkin and D. L. Souvaine and C. J. Van Wyk. Decomposition and Intersection of Simple Splinegons. Algorithmica, 3, 473-485, 1988

D. P. Dobkin and D. L. Souvaine. Computational Geometry in a Curved World. Algorithmica, 5(3), 421-457, 1990.