Faculty Publications

2000-8

Trajectory Planning for Coordinated Rendezvous of Unmanned Air Vehicles

Timothy W. McLain

Brigham Young University - Provo, mclain@byu.edu

Randal W. Beard

Brigham Young University - Provo, beard@byu.edu

Follow this and additional works at: <https://scholarsarchive.byu.edu/facpub>

Part of the [Mechanical Engineering Commons](#)

Original Publication Citation

McLain, T. and Beard, R. Trajectory Planning for Coordinated Rendezvous of Unmanned Air Vehicles, Proceedings of the AIAA Guidance, Navigation, and Control Conference, AIAA-2000-4369, August 2000, Denver, Colorado.

BYU ScholarsArchive Citation

McLain, Timothy W. and Beard, Randal W., "Trajectory Planning for Coordinated Rendezvous of Unmanned Air Vehicles" (2000). *Faculty Publications*. 1518.
<https://scholarsarchive.byu.edu/facpub/1518>

This Peer-Reviewed Article is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact ellen_amatangelo@byu.edu.

TRAJECTORY PLANNING FOR COORDINATED RENDEZVOUS OF UNMANNED AIR VEHICLES

Timothy W. McLain^{1*} Randal W. Beard²

¹Department of Mechanical Engineering, Brigham Young University, Provo, Utah 84602

²Department of Electrical and Computer Engineering, Brigham Young University, Provo, Utah 84602

ABSTRACT

A trajectory generation strategy that facilitates the coordination of multiple unmanned air vehicles is developed. Of particular interest is the planning of threat-avoiding trajectories that result in the simultaneous arrival of multiple UAVs at their targets. In this approach, paths to the target are modeled using the physical analogy of a chain. A unique strength of the planning approach is the ability to specify or alter the pathlength by adding or subtracting links from the chain. Desirable paths to the target are obtained by simulating the dynamics of the chain where threats apply repulsive forces to the chain and forces internal to the chain tend to straighten it out. The result for multiple vehicles and targets is a set of smooth and flyable paths of equal length that reduces exposure to threats.

1 INTRODUCTION

In the future operation of unmanned air vehicles, the ability to coordinate the timing of activities of UAVs in a multiple vehicle system will be vital to many missions. In this paper, the issue of trajectory planning is addressed for UAVs operating in a hostile environment. Specifically, the objective is to plan trajectories to a target location in a way that minimizes exposure to threats while keeping fuel usage at acceptable levels. An added constraint is the trajectory planning strategy must allow the duration of the trajectory to be specified. Given that the cruising speed of future UAVs is not likely to be widely variable, this duration constraint effectively fixes the path length of the trajectory if the cruising speed is assumed to be constant.

Figure 1 depicts the mission scenario which was used to motivate the trajectory planner development. Three UAVs are shown flying in close for-

mation. Targets are depicted by stars and threats are depicted as circles. As the UAVs approach the forward edge of the battle area and acquire detailed information about threat locations, each UAV plans a trajectory to its assigned target location. It is desired that the UAVs arrive at their targets simultaneously, thereby enhancing the element of surprise. Since their cruising speeds are nominally the same, simultaneous arrival of the UAVs at their respective targets will require that their trajectory path lengths be the same. The trajectory planning problem for each UAV becomes one of threat avoidance with a specified path length. This paper presents a strategy for coordination of time over target for multiple UAVs with fixed-length, optimal trajectory determination being the primary focus.

Figure 1: Coordinated time-over-target mission.

The subject of vehicle trajectory planning is very broad and has been the focus of a significant body

* Corresponding author, email: tmclain@et.byu.edu

of research, especially in the field of robotics (e.g., see Ref. 1). With respect to UAVs, trajectory planning has received limited attention. In Ref. 2 an overview of path planning issues relevant to military aircraft applications is given along with possible solution strategies. Ref. 3 describes a genetic algorithm approach for air-vehicle route planning. Bortoff⁴ describes a method for modeling a UAV trajectory using a series of point masses connected by springs. Threats exert repulsive forces on the masses causing the trajectory to move away from the threats. The steady-state condition of the mass-spring system forms the basis for the UAV trajectory.

Chandler, et.al.⁵ have developed a Voronoi diagram-based approach for UAV cooperative path generation. Significant emphasis was placed on the development of paths within the dynamic capabilities of the UAV. Cooperative rendezvous of multiple UAVs at a target location was accomplished by varying each UAV's velocity along its path. A general coordinated control strategy for multiple UAVs is described in Ref. 6 where rendezvous of multiple UAVs at a target was addressed.

2 TECHNICAL APPROACH

The trajectory planning approach taken in this work is carried out in several steps. In the first step, an initial path from the projected start position to the target position is constructed based upon the configuration of the threats. This initial path is determined from a graph search through the edges of a Voronoi diagram formed from the threat locations. The second step involves discretizing the initial path by dividing each leg of the path into fixed-length segments. These segments form a chain with a shape that can subsequently be optimized to minimize the possibility of detection. It should be noted that the length of the chain (and thus the time required to arrive at target) can be changed by adding or taking away links. In the final step, the chain path is smoothed and optimized. The ends of the chain are fixed and the chain is subjected to two types of forces: 1) repulsive forces from the threats; and 2) straightening forces within the chain. Under the effects of these forces, the steady-state configuration of the chain is solved for in an interactive fashion. The result is a smooth path through the threats having the desired length. Each of these steps is described in detail below.

Initial Path Generation

One of the main objectives in generating a path to the target is to avoid threats. An initial threat-avoiding path to the target can be devised by constructing a Voronoi diagram⁷ based on the locations of the threats. Figure 2 shows the Voronoi diagram generated for the problem under consideration. Notice that the Voronoi diagram is created without regard for the initial UAV location or the target location. To generate an initial path the UAV starting point and the target point must be connected to the Voronoi graph in some way. In this work, we simply connect the starting point and the target point to the three closest nodes of the Voronoi diagram. These connecting legs are shown by dashed lines in Figure 2.

Figure 2: Voronoi diagram for threat locations.

With the Voronoi diagram indicating numerous paths to the target, the final step in finding the initial path becomes determining which path to take. There are two costs associated with traveling along an edge of the Voronoi diagram: threat costs and fuel costs. Threat costs are based on a UAV's exposure to enemy radar. Since the strength of a UAV's radar signature is proportional to $1/d^4$, where d is the distance to the threat, the threat cost for traveling along an edge is proportional to the inverse of the distance (to the threat) to the fourth power. An exact threat cost calculation of the threat cost for traveling along an edge would involve the integration of the cost along the edge. A simpler approach involves calculating the threat cost at several locations

along an edge and taking into account the length of the edge. For this work, the threat cost was calculated at three points along each edge: $L_i/6$, $L_i/2$, and $5L_i/6$, where L_i is the length of the edge i . This is depicted graphically for several Voronoi edges in Figure 3.

Figure 3: Threat cost calculation.

The threat cost associated with the i^{th} edge is given by the expression

$$J_{t,i} = L_i \sum_{j=1}^N \left(\frac{1}{d_{\frac{1}{6},i,j}^4} + \frac{1}{d_{\frac{1}{2},i,j}^4} + \frac{1}{d_{\frac{5}{6},i,j}^4} \right)$$

where N is the number of threats, and $d_{\frac{1}{6},i,j}$ is the distance from the $1/6^{\text{th}}$ point on i^{th} edge to the j^{th} threat.

Given that we assume that the UAVs are flying at constant speed, the fuel required to travel along one edge of the Voronoi diagram will be proportional to the length of the edge. The fuel cost associated with the i^{th} edge is simply

$$J_{f,i} = L_i.$$

The total cost for traveling along an edge comes from a weighted sum of the threat and fuel costs:

$$J_i = kJ_{t,i} + (1 - k)J_{f,i}, \quad 0 \leq k \leq 1.$$

Selection of k between 0 and 1 allows the designer to place weight on exposure to threats or fuel expenditure depending on the mission scenario.

With the costs for traveling a particular edge of the Voronoi diagram determined, the Voronoi diagram is searched to determine the lowest cost path. Numerous graph searching techniques exist that would be appropriate to accomplish this task.⁷ For simplicity, we have used Dijkstra's algorithm,⁸

which requires that the graph be a directed graph, i.e., the direction of travel is specified for each leg of the graph. For each leg, the direction of travel is assigned to be toward the target. An example of an initial path determined by a graph search through the Voronoi diagram is shown in Figure 4. In this case, the cost weighting parameter k was chosen to give a reasonable balance between the proximity of the path to threats and the path length. The path shown is neither the shortest possible path nor the safest possible path.

Figure 4: Initial path example.

Path Discretization

Once an initial path has been generated, our approach requires the discretization of the path. This is done by superimposing fixed-length segments over the initial path. A physical analogy to this discretization would be the process of laying a chain over the initial path. This process is depicted graphically in Figure 5. Because the legs of the initial path cannot be represented exactly by an integral number of links, links are added until they are within one-half of a link length of the path node. At this point, a straight-line path to the next node is calculated and links are laid down in that direction. This process is carried out until the initial path has been traversed completely. It should be noted that once the chain comes within two link lengths of the target point, the link positions are computed so that the final link ends on the target point.

Figure 5: Superimposing chain links.

Since our primary objective is to have multiple UAVs arrive at their targets simultaneously, it is desirable to have the path lengths for the UAVs be equal. In practice, coordinated time over target could be achieved with different path lengths corresponding to the speed range capabilities of each UAV. In our approach we can easily constrain the path lengths to be the same and use the variable speed capabilities of the UAVs to make up for tracking errors. Path lengths are equalized by adding the appropriate number of links to each of the shorter paths so that the total number of links in each path is the same. Links are added to the chain paths in regions where the threat costs are lowest. Links can be added in a variety of forms, depending on the number of links to be added. Figure 6 shows three chain paths with links added to give them all the same length.

Path Refinement

The final step in the trajectory generation process is the shape refinement of the chain path. Shape refinement is carried out by treating the chain as a dynamic system with the end-point positions constrained. Forces are applied to the chain causing it to change shape. The goal is to approximate the steady-state shape of the chain. The shapes of the chain paths are refined in two ways. First, straightening forces are applied to the links of the chain to reduce the relative angles between the links. In doing so, the resulting path is made “flyable” by the UAV. Repulsive forces are also applied to the links of the chain by the threats. This tends to cause the

Figure 6: Equal-length chain paths.

chain to move away from the threats to safer regions of the battle area.

The method for finding the steady-state chain configuration is based on the work of Udwadia and Kalaba.⁹ Their method of dealing with constrained dynamics is uniquely suited to this problem and will be described briefly. In particular we will introduce the approach by developing dynamic equations for a three-link chain path.

Figure 7 depicts the three-link chain. The joints of the chain are modeled as point masses, with unit mass, constrained to a two dimensional surface. Let $\mathbf{z}_1 = (x_1, y_1)^T \in \mathbb{R}^2$ and $\mathbf{z}_2 = (x_2, y_2)^T \in \mathbb{R}^2$ be the location of the masses. If the point masses are unconstrained, then the dynamic equations describing their motion are

$$\begin{aligned}\ddot{\mathbf{z}}_1 &= \mathbf{u}_1 \\ \ddot{\mathbf{z}}_2 &= \mathbf{u}_2,\end{aligned}$$

where $\mathbf{u}_1, \mathbf{u}_2 \in \mathbb{R}^2$ are the applied forces. Alternatively, defining $\mathbf{x} = (\mathbf{z}_1^T, \mathbf{z}_2^T)^T$ and $\mathbf{u} = (\mathbf{u}_1^T, \mathbf{u}_2^T)^T$ gives

$$\ddot{\mathbf{x}} = \mathbf{u}. \quad (1)$$

However, as Figure 7 shows, the motion point-masses is constrained by the kinematics of the chain. These constraints can be written as

$$\begin{aligned}\|\mathbf{z}_1 - \mathbf{z}_0\|^2 &= L^2 \\ \|\mathbf{z}_2 - \mathbf{z}_1\|^2 &= L^2 \\ \|\mathbf{z}_f - \mathbf{z}_2\|^2 &= L^2,\end{aligned}$$

Figure 7: Three link chain.

or alternatively as

$$\phi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) \triangleq \begin{pmatrix} \|\mathbf{z}_1 - \mathbf{z}_0\|^2 - L^2 \\ \|\mathbf{z}_2 - \mathbf{z}_1\|^2 - L^2 \\ \|\mathbf{z}_f - \mathbf{z}_2\|^2 - L^2 \end{pmatrix} = 0. \quad (2)$$

Differentiating the constraints once with respect to time results in the velocity constraint

$$\psi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) \triangleq \begin{pmatrix} 2(\mathbf{z}_1 - \mathbf{z}_0)^T(\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0) \\ 2(\mathbf{z}_2 - \mathbf{z}_1)^T(\dot{\mathbf{z}}_2 - \dot{\mathbf{z}}_1) \\ 2(\mathbf{z}_f - \mathbf{z}_2)^T(\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_2) \end{pmatrix} = 0. \quad (3)$$

Differentiating once more results in the acceleration constraint

$$\begin{aligned} 2(\mathbf{z}_1 - \mathbf{z}_0)^T \ddot{\mathbf{z}}_1 + 2(\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0)^T(\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0) &= 0 \\ 2(\mathbf{z}_2 - \mathbf{z}_1)^T \ddot{\mathbf{z}}_2 + 2(\dot{\mathbf{z}}_2 - \dot{\mathbf{z}}_1)^T(\dot{\mathbf{z}}_2 - \dot{\mathbf{z}}_1) &= 0 \\ -2(\mathbf{z}_f - \mathbf{z}_2)^T \ddot{\mathbf{z}}_2 + 2(\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_2)^T(\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_2) &= 0, \end{aligned}$$

which can be written in matrix notation as

$$\mathbf{A}(\mathbf{x})\ddot{\mathbf{x}} = \mathbf{b}(\dot{\mathbf{x}}), \quad (4)$$

where

$$\mathbf{A}(\mathbf{x}) \triangleq \begin{pmatrix} (\mathbf{z}_1 - \mathbf{z}_0)^T & 0 \\ -(\mathbf{z}_2 - \mathbf{z}_1)^T & (\mathbf{z}_2 - \mathbf{z}_1)^T \\ 0 & -(\mathbf{z}_f - \mathbf{z}_2)^T \end{pmatrix}$$

$$\mathbf{b}(\mathbf{x}) \triangleq - \begin{pmatrix} (\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0)^T(\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0) \\ (\dot{\mathbf{z}}_2 - \dot{\mathbf{z}}_1)^T(\dot{\mathbf{z}}_2 - \dot{\mathbf{z}}_1) \\ (\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_2)^T(\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_2) \end{pmatrix}.$$

Udwadia and Kalaba show that using Gauss's principle from analytical dynamics, that the equation of motion of system (1) subject to the constraints (4) is given by the equation

$$\ddot{\mathbf{x}} = \mathbf{u} + \mathbf{A}^+(\mathbf{x})(\mathbf{b}(\dot{\mathbf{x}}) - \mathbf{A}(\mathbf{x})\mathbf{u}), \quad (5)$$

where $\mathbf{A}^+$ is the pseudo-inverse of $\mathbf{A}$.⁹ The initial conditions for the system must be chosen such that both $\phi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) = 0$ and $\psi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) = 0$.

Equation (5) is a remarkably simple equation. Note that the constrained acceleration of the system is decomposed into two terms: the first term is the unconstrained acceleration, and the second term is a gain ($\mathbf{A}^+(\mathbf{x})$) multiplied by the amount that the unconstrained accelerations violate the constraints. Compare the simplicity of this model with that obtained by using Euler-Lagrange, Newton-Euler, or Kane's method. One of the major advantages is that it is in a form that is ideally suited for numerical implementation since the pseudo-inverse can be computed on-line using singular value decomposition routines.

Unfortunately, the simplicity of the model does not come without drawbacks. The first drawback is that there are four variables in $\mathbf{x}$ as opposed to the two generalized variables needed in Euler-Lagrange or Kane's method. The second drawback is that while solving Equation (5), numerical error may cause the constraints $\phi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f)$ and $\psi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f)$ to drift from zero. When the constraints drift from zero, then Equation (5) no longer models the physical dynamics of the chain. As it stands, there is no mechanism in Equation (5) to bring the constraints back to zero. An interesting research problem would be to design an ODE solver that propagates the solution of Equation (5) while simultaneously ensuring that the constraints given in Equations (2) and (3) are never violated.

To circumvent the second problem, we modified Equation (5) as follows:

$$\begin{aligned} \ddot{\mathbf{x}} &= \mathbf{u} + \mathbf{A}^+(\mathbf{x})(\mathbf{b}(\dot{\mathbf{x}}) - \mathbf{A}(\mathbf{x})\mathbf{u}) \\ &\quad - \gamma_1 \frac{\partial \phi^T}{\partial \mathbf{x}} \phi - \gamma_2 \frac{\partial \psi^T}{\partial \mathbf{x}} \psi. \end{aligned} \quad (6)$$

The additional two terms force the constrained accelerations to descend the gradient of the constraints until they are no longer violated. Making γ_1 and γ_2 large ensures that the modified equation approximately models the dynamics of the constrained physical system. When γ_1 and γ_2 are made large, stiff ODE solvers must be used to propagate the dynamics: the slow dynamics correspond to the dynamics of the mechanical system, and the fast dynamics act to satisfy the mechanical constraints.

One of the major advantages to this approach is that the technique is easily extended to an arbitrary number of links. Consider the case when there are N joints and let $\mathbf{z}_i \in \mathbb{R}^2$ be the location of the i^{th} joint. Letting $\mathbf{x} = (\mathbf{z}_1^T, \dots, \mathbf{z}_N^T)^T$ and $\mathbf{u} = (\mathbf{u}_1^T, \dots, \mathbf{u}_N^T)^T$ we conclude that the unconstrained accelerations of the system satisfy Equation (1). The constraints are

given by

$$\begin{aligned}\phi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) &\triangleq \begin{pmatrix} \|\mathbf{z}_1 - \mathbf{z}_0\|^2 - L^2 \\ \vdots \\ \|\mathbf{z}_f - \mathbf{z}_N\|^2 - L^2 \end{pmatrix} = 0 \\ \psi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) &\triangleq \begin{pmatrix} 2(\mathbf{z}_1 - \mathbf{z}_0)^T (\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0) \\ \vdots \\ 2(\mathbf{z}_f - \mathbf{z}_N)^T (\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_N) \end{pmatrix} = 0.\end{aligned}$$

As before, differentiating the constraint $\phi(\mathbf{x}; \mathbf{z}_0, \mathbf{z}_f) = 0$ twice, results in Equation (4) where

$$\mathbf{A}(\mathbf{x}) \triangleq \begin{pmatrix} (\mathbf{z}_1 - \mathbf{z}_0)^T & 0 & \cdots & 0 \\ -(\mathbf{z}_2 - \mathbf{z}_1)^T & (\mathbf{z}_2 - \mathbf{z}_1)^T & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & -(\mathbf{z}_N - \mathbf{z}_{N-1})^T & (\mathbf{z}_N - \mathbf{z}_{N-1})^T \\ 0 & \cdots & 0 & -(\mathbf{z}_f - \mathbf{z}_N)^T \end{pmatrix}$$

$$\mathbf{b}(\mathbf{x}) \triangleq - \begin{pmatrix} (\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0)^T (\dot{\mathbf{z}}_1 - \dot{\mathbf{z}}_0) \\ \vdots \\ (\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_N)^T (\dot{\mathbf{z}}_f - \dot{\mathbf{z}}_N) \end{pmatrix}.$$

The dynamic equations of motion are given by Equation (6).

To complete the description of the $N+1$ link chain dynamics, we need to specify the unconstrained accelerations $\mathbf{u}$ which are equivalent to the applied forces on the nodes since the mass of each node is equal to one. The applied force is composed of two components: the first to repel from threats, and the second to straighten the chain.

Let $\tau_i \in \mathbb{R}^2$ be the (x, y) location of the i^{th} threat, and let T be the index set of all threats. To cause the chain to repel from threats, the force

$$F_{threats}(\mathbf{z}_i) = \gamma_{threats} \sum_{j \in T} \frac{\tau_j - \mathbf{z}_i}{\|\tau_j - \mathbf{z}_i\|^5}$$

is applied at each node, where $\gamma_{threats}$ is a non-negative gain.

To compute a force that will cause the chain to straighten, consider Figure 8. As shown in the figure, the force applied to the i^{th} node is composed of two terms. The first depends on the position of the $(i-2)^{\text{nd}}$ node. The essential idea is that we would like to pull on the i^{th} node until the sub-chain composed of the $(i-2)^{\text{nd}}$, $(i-1)^{\text{st}}$ and i^{th} nodes are in a straight line. Similarly, there is a force that pushes the i^{th} node in such a way, that nodes i , $i+1$, and $i+2$ form a straight line. Accordingly, the straightening

Figure 8: Force used to straighten the Chain.

force on the i^{th} node is given by

$$F_{straighten}(\mathbf{z}_i) = \gamma_{straighten} \left(\frac{\mathbf{z}_i - \mathbf{z}_{i+2}}{\|\mathbf{z}_i - \mathbf{z}_{i+2}\|} + \frac{\mathbf{z}_i - \mathbf{z}_{i-2}}{\|\mathbf{z}_i - \mathbf{z}_{i-2}\|} \right). \quad (7)$$

The total force on the i^{th} node is given by

$$\mathbf{u}_i = F_{threats}(\mathbf{z}_i) + F_{straighten}(\mathbf{z}_i).$$

The dynamics of the chain can now be evolved using Equation (6) and an adaptive step size ODE solver, such as Matlab's `ode45` algorithm. The primary disadvantage of using that algorithm in this application is that the solution of the chain dynamics cannot be carried out at the near real-time rates desired due to the adaptive step size of the algorithm. To obtain the real-time solution behavior desired, Equation (6) was solved using an Euler-type approximation:

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \gamma_{step} \left[\mathbf{u} \right. \\ &\quad \left. + \mathbf{A}^+(\mathbf{x}^{(k)}) \left(\mathbf{b}(\mathbf{x}^{(k)}) - \mathbf{A}(\mathbf{x}^{(k)}) \mathbf{u} \right) \right. \\ &\quad \left. - \gamma_1 \frac{\partial \phi^T}{\partial \mathbf{x}^{(k)}} \phi - \gamma_2 \frac{\partial \psi^T}{\partial \mathbf{x}^{(k)}} \psi \right]. \quad (8)\end{aligned}$$

This solution strategy allows the steady-state shape of the chain to be obtained in a numerically efficient manner.

Figure 9 shows smoothed-chain paths superimposed over the initial chain paths. It can be seen that kinks in the chains have been smoothed out and that in the process the chains has moved away from the closest threats. Perhaps most important is that the paths are of equal length, which facilitates the goal of simultaneous arrival at the targets.

Figure 9: Smoothed paths of equal length.

Figure 10: Coordinated arrival at targets.

3 RESULTS

The trajectory planning strategy outlined above was applied to a three-UAV system. In this example, the distances from the start point of the trajectory to the target for each of the vehicles is significantly different. Alteration of path lengths by the trajectory planner is therefore desirable to achieve simultaneous arrival at the targets.

Figure 10 shows the flight path of three UAVs following trajectories produced by the planner. The battle area shown is 40 km on a side. The icons representing the UAVs are shown at 42 second intervals and are scaled up in size by a factor of 200 for visibility.

Simultaneous arrival of the UAVs at their targets is achieved by forcing the paths to be the same length. After the initial process of laying down links along the Voronoi paths, UAV 3 had a path of 37 links, while UAV 1 had a path of 31 links and UAV 2 had a path of 29 links. Path lengths are equalized by automatically adding a 6-link belly to the path of UAV 1 and an 8-link loop to the path of UAV 2. After the path lengths are equalized, they are smoothed to produce the trajectories of Figure 10.

The range of each UAV to its target is shown in Figure 11. The effect of the added links on the progression of UAVs 1 and 2 toward their target can be easily seen. At approximately 70 s into the flight, the range of UAV 1 to its target increases as the UAV tracks the added links. At approximately 140 s

into the flight, the range of UAV 2 to its target increases momentarily as the UAV tracks the added loop. These added links allow the UAVs arrive at their targets simultaneously at $t = 352$ s.

Figure 11: Range to target vs. time.

4 CONCLUSIONS

In this paper, the development of a trajectory planning strategy suitable for coordinated timing of UAV activities has been presented. The primary strengths of the method are: 1) the ability to find

safe paths to a target while avoiding threats; 2) the ability to fix the lengths of the paths to specified values; and 3) the ability to use a dynamic chain analogy to smooth the paths into flyable shapes while maintaining a safe distance from threats. Using this path planning approach, the ability to plan trajectories enabling the simultaneous arrival of multiple UAVs at their targets has been demonstrated.

REFERENCES

- [1] J.-C. Latombe, *Robot Motion Planning*. Kluwer Academic Publishers, 1991.
- [2] J. Goldman, “Path planning problems and solutions,” in *Proceedings of the National Aerospace and Electronics Conference*, pp. 105–108, IEEE, 1994.
- [3] M. Pellazar, “Vehicle route planning with constraints using genetic algorithms,” in *Proceedings of the National Aerospace and Electronics Conference*, IEEE, 1994.
- [4] S. Bortoff, “Path planning for UAVs,” in *Proceedings of the American Control Conference*, 2000.
- [5] P. Chandler, S. Rasmussen, and M. Pachter, “UAV cooperative path planning,” in *Proceedings of the Guidance, Navigation, and Control Conference*, AIAA, 2000.
- [6] T. McLain, P. Chandler, and M. Pachter, “A decomposition strategy for optimal coordination of unmanned air vehicles,” in *Proceedings of the American Control Conference*, 2000.
- [7] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. Cambridge, MA: The MIT Press, 1990.
- [8] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms*. New Jersey: Prentice-Hall, Inc., 2000.
- [9] F. E. Udwarda and R. E. Kalaba, *Analytical Dynamics: A New Approach*. Cambridge University Press, 1996.