



---

Faculty Publications

---

2005-01-01

## Coordination Variables, Coordination Functions, and Cooperative Timing Missions

Randal Beard  
beard@byu.edu

Timothy W. McLain  
mclain@byu.edu

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

 Part of the [Electrical and Computer Engineering Commons](#)

### Original Publication Citation

Timothy W. McLain and Randal W. Beard. "Coordination Variables, Coordination Functions, and Cooperative Timing Missions", *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 1 (2005), pp. 150-161.

---

### BYU ScholarsArchive Citation

Beard, Randal and McLain, Timothy W., "Coordination Variables, Coordination Functions, and Cooperative Timing Missions" (2005). *Faculty Publications*. 1225.  
<https://scholarsarchive.byu.edu/facpub/1225>

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](mailto:ellen_amatangelo@byu.edu).

# Coordination Variables, Coordination Functions, and Cooperative Timing Missions

Timothy W. McLain \*      Randal W. Beard †

## Abstract

This paper presents a solution strategy for achieving cooperative timing among teams of vehicles. Based on the notion of coordination variables and coordination functions, the strategy facilitates cooperative timing by making efficient use of communication and computation resources. The application of the coordination variable/function approach to trajectory planning problems for teams of unmanned air vehicles with timing constraints is described. Three types of timing constraints are considered: simultaneous arrival, tight sequencing, and loose sequencing. Simulation results demonstrating the viability of the approach are presented.

## Introduction

The ability to plan paths in a cooperative fashion for a system of vehicles is of great importance in a wide variety of applications. This is especially true for military missions requiring precise timing or sequencing of tasks and operations. The ability “to adjust mission timing on the move to compensate for inevitable changes to plans and still make the time-on-target goal” has been identified as a key capability for the effective military use of unmanned air vehicles (UAVs).<sup>1</sup> Other operations of importance in civil and military aviation that could benefit from cooperative path planning include flight traffic control and landing operations.

There are numerous technical challenges to overcome to develop viable cooperative planning methods for a distributed team of unmanned air vehicles. A significant issue is the level of complexity involved in planning paths for a team of vehicles with competing interests so that the team objective is attained in a satisfactory or optimal manner. Given that the path planning process must produce desired state trajectories for each UAV continually throughout a mission, a significant volume of information must be computed in real time.

---

\*Associate Professor, Department of Mechanical Engineering, Brigham Young University, Provo, Utah 84602

†Associate Professor, Department of Electrical and Computer Engineering, Brigham Young University, Provo, Utah 84602

The complexity of the cooperative path planning problem is increased by the possibility of a changing environment. In military missions, the path planner must be able to respond quickly to pop-up threats or other unanticipated changes in the threat scenario. To maintain desired levels of stealth, communication among vehicles should be limited. For missions requiring precise timing or sequencing of operations, the cooperative path planner must coordinate the timing of tasks so that the team objectives are achieved. Finally, trajectories provided by the path planner must be within the dynamic capabilities of the UAV. Precise timing of operations is not possible if significant tracking errors exist.

Although path planning for single UAVs has been an active area of research for some time (e.g., see Refs. 2–6), work on cooperative control and cooperative path planning for UAVs has only recently begun to appear. In Ref. 7, a decentralized optimization method is developed and applied to a multiple aircraft coordination problem. A bargaining algorithm based on sequential local optimization is used to approximate the global centralized optimization solution. Cooperative timing problems are sensitive to the assignment and ordering of tasks. One approach for handling cooperative timing is to apply timing constraints to the task assignment problem. In Refs. 8 and 9, mixed-integer linear programming (MILP) is used to solve tightly-coupled task assignment problems with timing constraints. The advantage to this approach is that it yields the optimal solution for a given problem. The primary disadvantages are the complexity of problem formulation and the computational burden involved. Pruning strategies for simplifying the MILP problem have been proposed to enable near-real-time solutions.

The objective of this paper is to introduce a general approach to cooperative control problems, and to specifically demonstrate the application of the technique to cooperative timing missions.

The fundamental axiom of our approach is:

*Cooperation among a team of vehicles requires information to be shared.*

Information may be shared in a variety of ways. For example, relative position sensors may enable vehicles to construct state information of other vehicles, information may be communicated between vehicles using a wireless network, or joint information might be pre-programmed into the vehicles before the mission begins. Our approach is to collect the information that must be jointly shared to facilitate cooperation into a single quantity called the *coordination variable*. The coordination variable represents the minimal amount of information needed to enable a specific cooperation objective.

Although it is known by different names, the notion of a coordination variable is found in other works on cooperative control. For example Refs. 10, 11 introduce an “action reference,” which if known by each vehicle, facilitates formation keeping. In leader-following applications,<sup>12,13</sup> the states of the leader constitute the coordination variable since the action of the other vehicles in the formation are completely specified once the leader states are known. In Refs. 14–16, the notion of a virtual structure is used to derive formation control strategies. The motion of each vehicle is causally dependent on the dynamic states of the virtual structure, therefore the states of the virtual structure is the coordination variable. Coordination variables may also be

more discrete in nature. For example, in Refs. 8 and 9, cooperative task allocation is addressed. Individual vehicle behavior is dependent on the task allocation vector which becomes the coordination variable.

A second main concept in the cooperative control strategy introduced in this paper is the notion of coordination functions. Coordination functions parameterize the effect of the coordination variable on the myopic objectives of each agent. The idea is to parameterize how changing the coordination variable impacts the individual myopic objectives, and then to use this information to select a feasible coordination variable value for the team. Although the notion of coordination variables is prevalent in other works, the notion of a coordination function and the concept of modifying the team coordination variable based on costs to individual team members, seems to be missing in most of the cooperative control literature. One of the contributions of this paper is to provide a formal mechanism for introducing this type of team feedback.

In the cooperative timing problems considered in this paper, the coordination variable defines mission-critical timing information, such as estimated-time-of-arrival (ETA) at a specified destination. The coordination function describes the cost to an individual UAV of achieving different values of the coordination variable that are feasible for the UAV. Cooperative path planning is enabled by communication of coordination functions and coordination variables among UAVs participating in the mission. Preliminary investigations of this approach have been reported in Refs. 17 and 18.

The approach presented here has several strengths that make it suitable for cooperative timing scenarios.<sup>19,20</sup> First, and perhaps most important, the coordination variable/function strategy reduces the dimensionality and complexity of the problem to tractable levels. The strategy requires information vital to the cooperative timing effort to be organized in an efficient manner. If a global optimization approach were attempted, determining the team-optimal timing for mission success would require knowing the states of all of the UAVs participating as well as the location of all threats and targets or destinations. Even for a relatively small number of vehicles, the task of determining optimal timing and the corresponding UAV trajectories in this global approach requires excessive communication bandwidth and is difficult due to the large number of decision variables involved.

Alternatively, our approach simplifies the task of determining team-optimal timing through the implementation of task-appropriate coordination variables and coordination functions. Rather than requiring the determination of UAV state trajectories for all UAVs on the team centrally, only the critical timing information represented by the coordination variable must be determined. Based on the team-optimal coordination variable, UAV trajectories are determined in a decentralized fashion on the individual UAVs. This decomposition of the cooperative path planning problem results in significant simplification of the team-level planning process and a substantial reduction in the volume of information communicated among UAVs.

# Cooperative Timing Problem

The primary objective of this paper is to describe a general cooperative control strategy based on coordination variables and functions. To aid and provide context for the presentation, the specific problem of cooperative timing among a team of fixed-wing UAVs will be considered. The UAVs are modeled as having 12th-order nonlinear dynamics as is typical for fixed-wing aircraft<sup>21,22</sup> and having altitude-hold, heading-hold, and velocity-hold autopilots. From a trajectory planning perspective, two constraints imposed by the dynamics are most significant: those on turn radius due to heading rate limits and on velocity due to stall and maximum thrust limits.

For the cooperative timing problems considered in this paper, the goal is to coordinate the arrival of a team of UAVs at preassigned targets to satisfy a particular timing constraint. The UAVs may have their own individual targets, or they may be assigned the same target. The timing constraint must be satisfied while the UAVs on the team minimize their exposure to threats. In this work, threats are assumed to be radar sites and the objective is to stay away from the point of emanation of the radar signal. Polygonal obstacles or no-fly zones have also been considered.<sup>23</sup> Although threats are assumed to be static to limit the scope of the discussion, the speed of the cooperative path planning algorithm allows pop-up threats and slow moving (relative to the UAV speed) threats to be handled effectively (see e.g., Ref. 24).

In this paper, three types of timing constraints are considered:

**Simultaneous Arrival.** Simultaneous arrival constraints imply that all vehicles on the team arrive at their destinations at the same time.

**Tight Sequencing.** Tight sequencing constraints require that the vehicles arrive at their destinations in a specified sequence with the time increments between arrival times specified exactly.

**Loose Sequencing.** Loose sequencing constraints require that the vehicles arrive at their destinations in a specified sequence with the time increments between arrival times given in terms of acceptable ranges.

For cooperative timing problems, these timing constraints imply that trajectories for all of the vehicles involved must be planned simultaneously. Given that near-real-time planning of dynamically feasible trajectories for a single UAV is challenging, cooperative planning for multiple vehicles poses significant difficulties. The coordination variable/function technique of this paper provides a computationally feasible approach to the problem by enabling efficient distributed computation of cooperative plans.

Figure 1 illustrates the trajectory planning architecture for a single UAV, which is composed of three main parts: the Coordination Manager (CM), the Waypoint Path Planner (WPP), and the Dynamic Trajectory Smoother (DTS). The WPP quickly calculates threat-avoiding straight-line paths. These candidate paths are used by the CM to determine coordination function information as well as cooperative timing information for the team. Because the straight-line waypoint paths produced by

the WPP are not dynamically feasible for the UAV to fly, the DTS is employed to produce flyable trajectories and autopilot commands. The DTS smoothes junctions in the waypoint path with a sequence of radial arcs that can be flown by the UAV. This smoothing of the waypoint path is carried out in real-time. Most importantly, the length of the original straight-line path is preserved in the smoothing process.<sup>6</sup> This is essential for timing-critical missions.

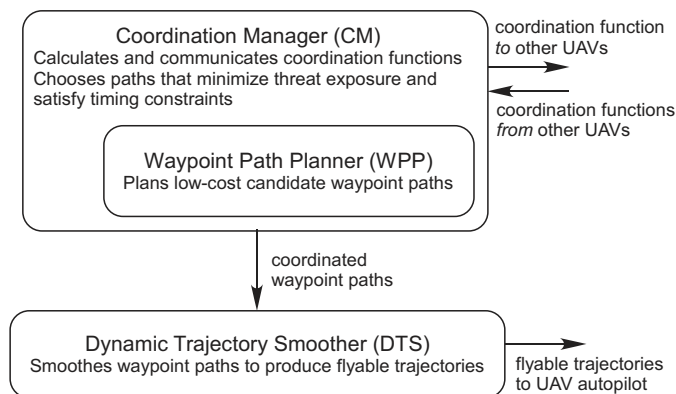


Figure 1: Trajectory planning architecture.

The cooperative control architecture for a team of three UAVs is shown in Figure 2. The architecture is distributed, enabling each UAV to perform its own trajectory computations (WPP and DTS). The CM algorithm implemented on each UAV is identical. By sharing the information contained in each UAV’s coordination function across the team, each UAV computes the same cooperative timing solution. This approach efficiently utilizes the computational resources of the team and increases the team’s robustness to a single point failure. Furthermore, coordination functions are designed to efficiently communicate only that information essential for cooperation, thereby reducing the communication demands on the system. Although shown for three UAVs, the distributed structure of the architecture allows it to accommodate larger numbers of UAVs without significantly impacting the computational requirements.

The coordination function information depicted as being passed among vehicles in Figure 2 is central to the cooperative control method presented in this paper. The following section describes coordination functions, how they are generated, and how they are used by the Coordination Manager to find team optimal timing solutions. Although the focus of this paper is on cooperative timing, the cooperative control strategy based on coordination variables and coordination functions is general and can be applied to other cooperative control problems.

## Coordination Variables and Coordination Functions

Cooperative control by a team of vehicles is dependent on the environment or mission scenario in which the vehicles are acting. To characterize the significant elements of

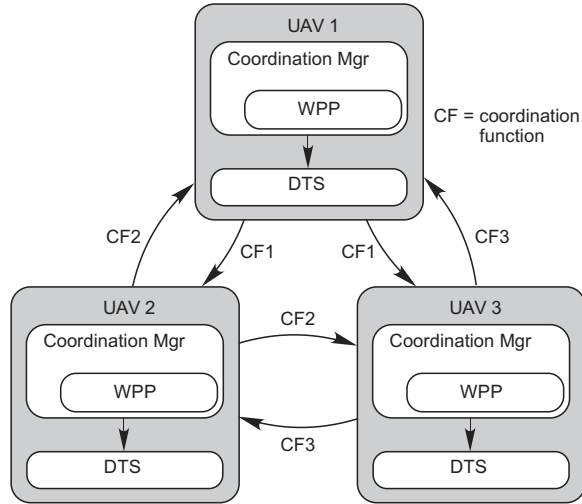


Figure 2: Cooperative control architecture for team of three.

the environment, define  $\mathcal{X}_i$  to be the situation state space for the  $i^{\text{th}}$  vehicle and let  $\mathbf{x}_i \in \mathcal{X}_i$  be the situation state of the  $i^{\text{th}}$  vehicle. For the timing problems considered, the situation state includes the location of threats and targets and the current location of the UAV. The situation state could also include weather conditions or significant geographic features.

For a given scenario, each vehicle can act to influence the effectiveness of the team. Let  $\mathcal{U}_i(\mathbf{x}_i)$  be the set of feasible decision values for situation state  $\mathbf{x}_i$ , and let  $\mathbf{u}_i \in \mathcal{U}_i(\mathbf{x}_i)$  be the decision variable for the  $i^{\text{th}}$  vehicle. In this paper, the set of feasible decision values for a vehicle is the set of threat-avoiding paths produced by the Waypoint Path Planner. As the notation implies, this set of paths is functionally dependent on the situation state.

Our basic axiom implies that there is a minimum amount of information needed by the team to effect cooperation. We will call this information the *coordination variable* and denote it by  $\theta$ . The coordination variable is a vector in coordination space  $\mathbb{R}^c$ . The essential idea is that if every agent knows the coordination variable and responds appropriately, then cooperative behavior will be achieved. As an example, for the simultaneous arrival constraint introduced in the previous section, the coordination variable is the arrival time. Similarly, for the tight and loose sequencing constraints, the coordination variable is the arrival time of the first vehicle. In general, the cooperation constraints, such as the timing constraints considered here, can be expressed in terms of the coordination variable. In the Coordination Manager, selection of the coordination variable is made to ensure that the cooperation constraints are met.

The distillation of information from situation state and decision variables (full information) to the coordination variable representation (minimal information) is central to this method. If  $f_i : \mathcal{X}_i \times \mathcal{U}_i \rightarrow \mathbb{R}^c$  is a function that maps situation state and decision vector pairs to coordination space  $\mathbb{R}^c$ , then the set of feasible

coordination variables for the  $i^{\text{th}}$  vehicle in situation state  $\mathbf{x}_i$  is given by

$$\Theta_i(\mathbf{x}_i) = \bigcup_{\mathbf{u}_i \in \mathcal{U}_i(\mathbf{x}_i)} f_i(\mathbf{x}_i, \mathbf{u}_i). \quad (1)$$

In the context of cooperative timing,  $f_i$  is simply a map between the waypoint path options for a certain situation and the resulting set of possible arrival times. Note that  $\Theta_i(\mathbf{x}_i)$  is not necessarily a connected set. For a specific path and velocity choice given by  $\mathbf{u}_i$ , the coordination variable takes on a unique value  $\theta_i = f_i(\mathbf{x}_i, \mathbf{u}_i) \in \Theta_i(\mathbf{x}_i)$ . Note also that the cooperative timing constraints considered previously can be represented in terms of the coordination variable. For example, simultaneous arrival constraints can be expressed as

$$f_i(\mathbf{x}_i, \mathbf{u}_i) = f_j(\mathbf{x}_j, \mathbf{u}_j), \quad \forall i, j \in \{1, \dots, N\}.$$

We assume that  $f_i$  is (pseudo) invertible in the sense that there exists a function  $f_i^\dagger : \mathcal{X}_i \times \Theta_i \rightarrow \mathcal{U}_i$  (called the pseudo-inverse of  $f$ ), such that for every  $\theta \in \Theta_i(\mathbf{x}_i)$ ,  $f_i(\mathbf{x}_i, f_i^\dagger(\mathbf{x}_i, \theta)) = \theta$ . Simply stated, if the situational state and the coordination variable are known, the decision variable is unique. For the timing problems considered, this can be interpreted to mean that if the desired arrival time is specified by the Coordination Manager, then a specific path and velocity to achieve that arrival time can be determined.

In addition to cooperation constraints (such as those on timing), the vehicles on the team have individual performance objectives that are typically in harmony with the team cooperation objective. As with the coordination variable, these individual myopic objectives are dependent on the situation state and the decision vector. For the  $i^{\text{th}}$  vehicle, this myopic performance objective can be represented by the function  $J_i : \mathcal{X}_i \times \mathcal{U}_i \rightarrow \mathbb{R}$ . By using the relationship  $\mathbf{u}_i = f_i^\dagger(\mathbf{x}_i, \theta)$  for each  $\theta \in \Theta_i(\mathbf{x}_i)$ , the myopic objective can be parameterized as a function of the coordination variable:

$$\phi_i(\mathbf{x}_i, \theta) = J_i(\mathbf{x}_i, f_i^\dagger(\mathbf{x}_i, \theta)). \quad (2)$$

The function  $\phi_i : \mathcal{X}_i \times \Theta_i(\mathbf{x}_i) \rightarrow \mathbb{R}$  given by Equation (2) is called the *coordination function* of the  $i^{\text{th}}$  vehicle. For a given situation state  $\mathbf{x}_i$ , the coordination function parameterizes the myopic performance objective of the  $i^{\text{th}}$  vehicle verses the coordination variable. For the cooperative timing problems posed here, the individual myopic objective is to avoid threats. For each UAV, the coordination function describes the threat exposure cost that would be incurred for all of the achievable arrival times. In this case the coordination function is an efficient representation of essential information necessary for team timing decisions.

In this paper, the cooperation problems of interest can be posed as a minimization of a team objective function, where the team objective is a function of the individual myopic objective functions. If  $\mathcal{J}_T : \mathbb{R}^N \rightarrow \mathbb{R}$  is defined as the team objective function, then the simultaneous arrival cooperative timing problem is to find decision variables  $\mathbf{u}_1, \dots, \mathbf{u}_N$  that solve the following optimization problem:

$$(\mathbf{u}_1, \dots, \mathbf{u}_N) = \arg \min_{\mathcal{U}_1 \times \dots \times \mathcal{U}_N} \mathcal{J}_T(J_1(\mathbf{x}_1, \mathbf{u}_1), \dots, J_N(\mathbf{x}_N, \mathbf{u}_N)), \quad (3)$$



subject to

$$f_i(\mathbf{x}_i, \mathbf{u}_i) = f_j(\mathbf{x}_j, \mathbf{u}_j), \quad \forall i, j \in \{1, \dots, N\}. \quad (4)$$

This optimization problem will clearly pose computational problems as the number of vehicles increase, and for large state and decision variable dimensions.

Using coordination variables and coordination functions, a decomposition of the optimization problem of Equations (3) and (4) that captures the information essential for cooperation can be posed:

$$\theta^* = \arg \min_{\theta \in \Theta_i(\mathbf{x}_i)} \mathcal{J}_T(\phi_1(\mathbf{x}_1, \theta), \dots, \phi_N(\mathbf{x}_N, \theta)). \quad (5)$$

Cooperation is said to occur when the cooperation constraint of Equation (4) is satisfied. In the decomposed formulation of Equation (5), the cooperation constraint is implicitly satisfied through the choice of a single coordination variable value  $\theta^*$ . The objective functions of Equations (3) and (5) give a measure of the quality of cooperation achieved. In the case of the cooperative timing problem, the best solutions not only satisfy the timing constraints, but also minimize the collective threat exposure of the team.

Once a team optimal value for the coordination variable  $\theta^*$  is found, individual vehicle decisions can be found by solving for the decision variable from the relationship

$$\mathbf{u}_i = f_i^\dagger(\mathbf{x}_i, \theta^*).$$

This two-level decomposition process significantly reduces the computation and communication loads. In the next section, we will apply the coordination variable/coordination function framework to several cooperative timing scenarios.

## Application to Cooperative Timing Scenarios

The situation state space  $\mathcal{X}_i$  for the cooperative timing problems considered in this paper, consists of the Cartesian product of a UAV position vector, a target position vector, and the set of threat locations. Therefore

$$\mathbf{x}_i = \begin{pmatrix} \mathbf{z}_{i0} \\ \mathbf{z}_{if} \\ \mathcal{H}_i \end{pmatrix},$$

where  $\mathbf{z}_{i0}$  is the current position of the UAV,  $\mathbf{z}_{if}$  is the target position, and  $\mathcal{H}_i$  is the set of threat locations known to the  $i^{\text{th}}$  vehicle.

The set of feasible decision vectors  $\mathcal{U}_i(\mathbf{x}_i)$  at  $\mathbf{x}_i \in \mathcal{X}_i$  is a range of feasible velocities  $[v_{\min}, v_{\max}]$ , and a set of waypoint paths from the UAV position to the target position. Therefore a decision vector

$$\mathbf{u}_i = \begin{pmatrix} v_i \\ \mathcal{W}_i \end{pmatrix}$$

consists of a feasible velocity  $v_i$  and a waypoint path  $\mathcal{W}_i = \{\mathbf{w}_{i1}, \mathbf{w}_{i2}, \dots, \mathbf{w}_{iP}\}$ , where  $\mathbf{w}_{i1} = \mathbf{z}_{i0}$  and  $\mathbf{w}_{iP} = \mathbf{z}_{if}$ .

For cooperative timing problems, coordination hinges on arrival times at the target. Therefore the coordination variable  $\theta_i$  is the estimated-time-of-arrival (ETA) if the UAV were to fly the waypoint path  $\mathcal{W}_i$  at velocity  $v_i$ . For a given path  $\mathcal{W}_i$ , the length of the path is given by

$$L(\mathcal{W}_i) = \sum_{j=2}^P \|\mathbf{w}_{ij} - \mathbf{w}_{i(j-1)}\|.$$

The mapping from state and decision vectors to the coordination variable is given by

$$f_i(\mathbf{x}_i, \mathbf{u}_i) = L(\mathcal{W}_i)/v_i. \quad (6)$$

Since  $v_i$  can vary over the feasible range  $[v_{\min}, v_{\max}]$ , for a given path  $\mathcal{W}_i$ , the set of possible coordination variable values associated with that path is given by the compact segment  $[L(\mathcal{W}_i)/v_{\max}, L(\mathcal{W}_i)/v_{\min}]$ . If  $\mathcal{U}_i(\mathbf{x}_i)$  consists of a finite set of waypoint paths, then the set of feasible coordination variables given by Equation (1) consists of the union of a set of compact segments on  $\mathbb{R}$ , as shown in Figure 3.

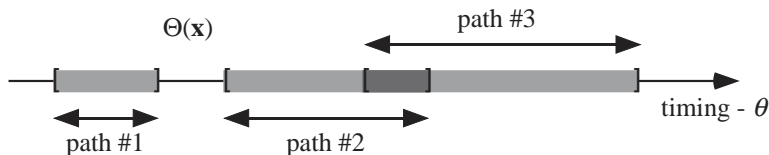


Figure 3: The set of feasible coordination variables  $\Theta_i(\mathbf{x}_i)$  is the union of a finite set of compact intervals on  $\mathbb{R}$ .

In this paper we will assume that the myopic performance objective  $J_i$  is given by a linear combination of threat cost and fuel cost:

$$J_i(\mathbf{x}_i, \mathbf{u}_i) = (1 - \kappa)J_{\text{threat}}(\mathbf{x}_i, \mathbf{u}_i) + \kappa J_{\text{fuel}}(\mathbf{x}_i, \mathbf{u}_i), \quad (7)$$

where  $\kappa \in [0, 1]$  gives the designer flexibility to emphasize exposure to threats or fuel expenditure depending on the particular mission scenario.

The threat cost model is based on exposure to threat radar sites and is influenced by the proximity of the threat and the length of time exposed. The signal reflected to the threat radar is assumed to be uniform in all directions and its strength is proportional to  $1/d^4$  where  $d$  is the distance from the UAV to the threat.<sup>25</sup> The total threat cost is given by

$$J_{\text{threat}}(\mathbf{x}_i, \mathbf{u}_i) = \sum_{\mathbf{h} \in \mathcal{H}} \sum_{j=2}^P \hat{J}_{\text{threat}}(\mathbf{h}, \mathbf{w}_j, \mathbf{w}_{j-1}), \quad (8)$$

where

$$\hat{J}_{\text{threat}}(\mathbf{h}, \mathbf{w}_j, \mathbf{w}_{j-1}) = \frac{1}{v_i} \int_{s=0}^1 \frac{1}{\|\mathbf{h} - (\mathbf{w}_{j-1} + s(\mathbf{w}_j - \mathbf{w}_{j-1}))\|^4} ds,$$

and where  $s \in [0, 1]$  parameterizes the straight-line path from  $\mathbf{w}_{j-1}$  to  $\mathbf{w}_j$ . A closed-form solution to this integral is given by

$$\hat{J}_{\text{threat}}(\mathbf{h}, \mathbf{w}_j, \mathbf{w}_{j-1}) = \frac{1}{2\beta^2 \|\mathbf{w}_j - \mathbf{w}_{j-1}\|^3} \left[ \frac{1 - \alpha}{(1 - \alpha)^2 + \beta^2} + \frac{1}{\beta} \tan^{-1} \left( \frac{1 - \alpha}{\beta} \right) + \frac{\alpha}{\alpha^2 + \beta^2} + \frac{1}{\beta} \tan^{-1} \left( \frac{\alpha}{\beta} \right) \right],$$

where

$$\alpha = \frac{(\mathbf{h} - \mathbf{w}_{j-1})^T (\mathbf{w}_j - \mathbf{w}_{j-1})}{\|\mathbf{w}_j - \mathbf{w}_{j-1}\|^2}$$

$$\beta = \frac{\|[\|\mathbf{w}_j - \mathbf{w}_{j-1}\|^2 I - (\mathbf{w}_j - \mathbf{w}_{j-1})(\mathbf{w}_j - \mathbf{w}_{j-1})^T] (\mathbf{h} - \mathbf{w}_{j-1})\|}{\|\mathbf{w}_j - \mathbf{w}_{j-1}\|^3}.$$

The fuel cost for traversing an edge is calculated based on the assumption that fuel usage rate is proportional to the aerodynamic drag force which is proportional to velocity squared. Accordingly, the fuel required to traverse an edge of a waypoint path from  $\mathbf{w}_{j-1}$  to  $\mathbf{w}_j$  is given by

$$\begin{aligned} \hat{J}_{\text{fuel}}(v_i, \mathbf{w}_j, \mathbf{w}_{j-1}) &= \int_{t_{j-1}}^{t_j} \dot{f} dt \\ &= c_{\text{fuel}} v_i^2 (t_j - t_{j-1}) \\ &= c_{\text{fuel}} v_i \|\mathbf{w}_j - \mathbf{w}_{j-1}\|, \end{aligned}$$

where  $c_{\text{fuel}} > 0$  is a constant. Under the assumption of constant, uniform velocity over the path, the total fuel cost is given by

$$J_{\text{fuel}}(\mathbf{x}_i, \mathbf{u}_i) = \sum_{j=2}^P \hat{J}_{\text{fuel}}(v_i, \mathbf{w}_j, \mathbf{w}_{j-1}) = c_{\text{fuel}} v_i L(\mathcal{W}_i). \quad (9)$$

Next consider the problem of determining a trajectory  $\mathbf{u}_i \in \mathcal{U}_i(\mathbf{x}_i)$  to achieve a specified arrival time  $\theta \in \Theta_i(\mathbf{x}_i)$  for a given situation state  $\mathbf{x}_i$ . If the function  $f_i$  is a one-to-one map between  $\mathcal{U}$  and  $\Theta$ , then the inverse of  $f_i$  exists and the trajectory  $\mathbf{u}$  can easily be determined from the specification of the arrival time  $\theta$ . In this case, the coordination function is equal to the myopic individual cost function:  $\bigcup_{\theta \in \Theta_i(\mathbf{x}_i)} \phi_i(\mathbf{x}_i, \theta) = \bigcup_{\mathbf{u}_i \in \mathcal{U}_i(\mathbf{x}_i)} J_i(\mathbf{x}_i, \mathbf{u}_i)$ . In practice, this is seldom the case. As illustrated by Figures 3 and 4, a specific arrival time can typically be achieved by more than one trajectory.

A method for constructing a pseudo-inverse for  $f_i$  is needed. As a first step in constructing  $f_i^\dagger$ , note that for a given situation state  $\mathbf{x}_i$ , each candidate trajectory  $\mathbf{u}_i \in \mathcal{U}_i(\mathbf{x}_i)$  results in both a myopic cost  $J_i(\mathbf{x}_i, \mathbf{u}_i)$  and a candidate coordination variable  $\theta = f(\mathbf{x}_i, \mathbf{u}_i)$ . It is interesting to plot the locus of points  $\bigcup_{\mathbf{u}_i \in \mathcal{U}_i(\mathbf{x}_i)} (J_i, \theta)$ , which is shown in Figure 4. Each line segment represents a specific path. The spread

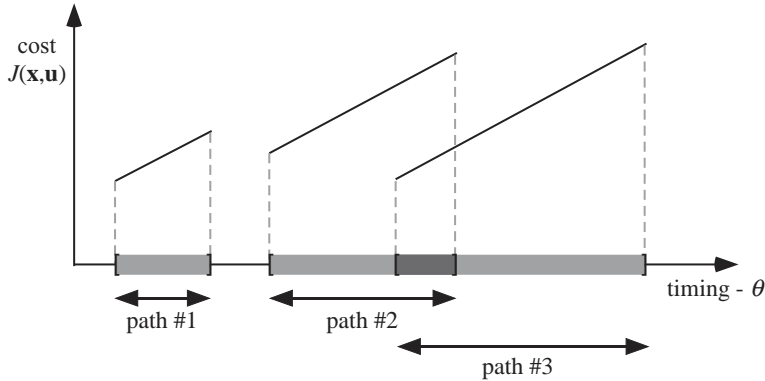


Figure 4: The locus of points  $(J_i, \theta)$  over the set  $\mathcal{U}(\mathbf{x}_i)$  for a fixed  $\mathbf{x}_i$ .

in arrival times and in the myopic cost results from the range of velocities that can be taken along each possible path.

While several pseudo-inverses are possible for  $f_i$ , in this paper we will select  $\mathbf{u}_i$  that results in the minimum cost trajectory:

$$f_i^\dagger(\mathbf{x}_i, \theta) = \arg \min_{\mathbf{u}_i \in \mathcal{U}(\mathbf{x}_i)} J_i(\mathbf{x}_i, \mathbf{u}_i)$$

subject to :

$$\theta = f_i(\mathbf{x}_i, \mathbf{u}_i).$$

The coordination function given by Equation (2) is shown in Figure 5. If  $f_i$  is not one-to-one (as it is not in Figure 5), then it follows that  $\bigcup_{\theta \in \Theta_i(\mathbf{x}_i)} f_i^\dagger(\mathbf{x}_i, \theta)$  may only be a proper subset of  $\mathcal{U}_i$ , and  $\phi_i(\mathbf{x}_i, \cdot)$  an approximation of  $J_i(\mathbf{x}_i, \cdot)$ . By taking the lowest cost trajectory for a specified arrival time, this approximation is made so that only good, locally optimal decisions are considered from  $\mathcal{U}_i$ .

For this problem the coordination function can be conveniently represented by a sequence of  $(J, \theta)$  pairs that define the straight-line segments represented in Figure 5. Therefore the coordination function for each vehicle is simple to represent and communicate.

In this paper, the team objective function  $\mathcal{J}_T$  is simply the sum of myopic objective functions:

$$\mathcal{J}_T(J_1, \dots, J_N) = \sum_{i=1}^N J_i \approx \sum_{i=1}^N \phi_i(\theta).$$

As such, the team objective can easily be expressed as a function of the individual coordination functions. The problem then becomes that of finding the best value of the coordination variable for the team.

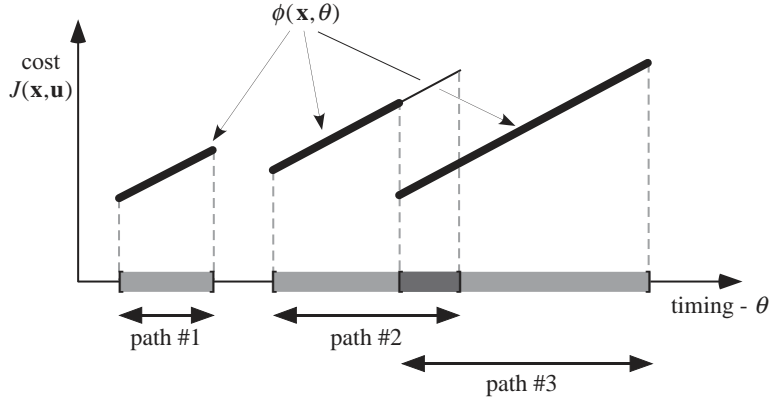


Figure 5: The coordination function is based on a pseudo-inverse of  $f$  that selects lowest cost path associated with the candidate coordination variable  $\theta$ .

### Simultaneous Arrival Constraints

For a team of  $N$  vehicles that are constrained to arrive simultaneously at their destinations, the simultaneous arrival constraint can be stated simply as

$$T_1 = T_2 = \dots = T_N = T_s,$$

where  $T_i = f(\mathbf{x}_i, \mathbf{u}_i)$  given in Equation (6), and  $\theta = T_s$  is the coordination variable. The upper plot in Figure 6 shows coordination functions for a team of three vehicles. The team optimization problem can be visualized as sweeping through the coordination functions while continually monitoring their sum (the team objective). For the timing problems considered here, the coordination functions are piecewise monotonically increasing. It can be shown that the team optimum occurs at the left extreme of a piecewise continuous segment of one of the coordination functions. The resulting optimization problem is therefore straightforward to solve, requiring only a global search through the left extreme points of each of the coordination functions.

### Tight Sequencing Constraints

Tight sequencing is characterized by enforcing specified intervals between the arrival times of each of the vehicles composing the team. The middle plot of Figure 6 shows coordination functions for a team of three vehicles with the vertical lines indicating the spacing in arrival times. The tight sequencing constraint for a team of vehicles can be formulated as

$$\begin{aligned} T_1 &= T_s \\ T_i &= T_s + \Delta_i \quad i = 2, \dots, N, \end{aligned}$$

where  $\Delta_i$  represents the interval between the arrival of the first and  $i^{\text{th}}$  vehicles, and  $\theta = T_s$  is the coordination variable. The mapping of Equation (6) must therefore be modified to

$$f_i(\mathbf{x}_i, \mathbf{u}_i) = L(\mathcal{W}_i)/v_i - \Delta_i.$$

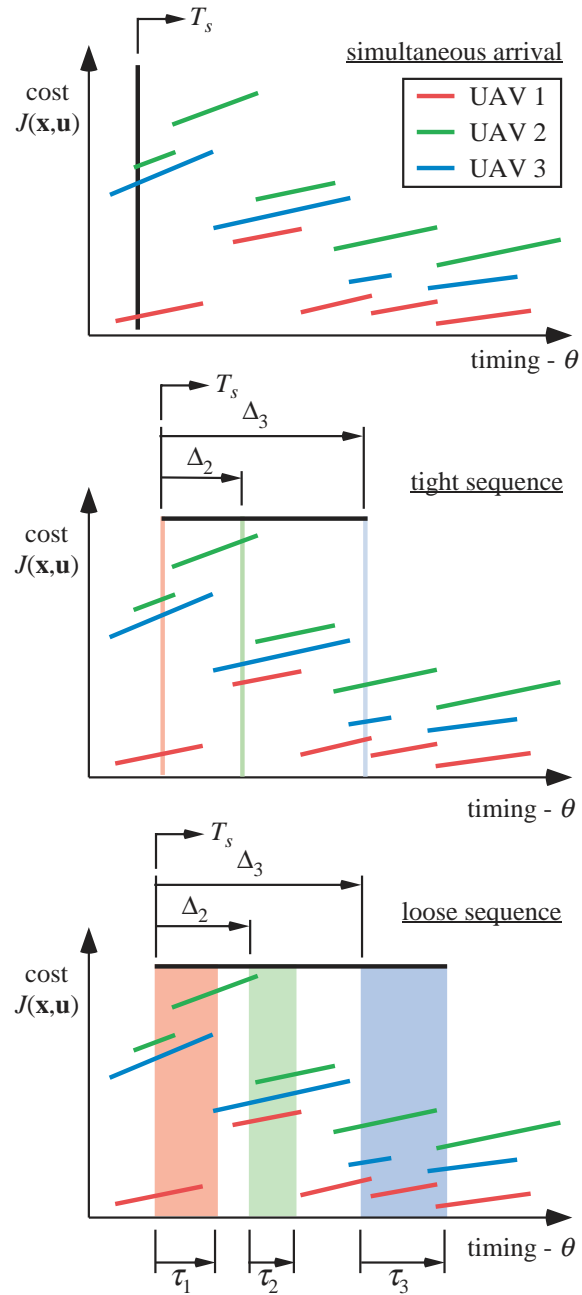


Figure 6: Coordination variable determination for three timing constraints.

The team-optimal arrival time for one of the vehicles will occur at the left extreme of a piecewise continuous segment of its coordination function. The corresponding optimal arrival times for the other vehicles will be determined by the specified intervals. Therefore, as indicated in Figure 6, the team optimization problem can be formulated as sweeping through the set  $\bigcap_{i=1}^N \Theta_i(\mathbf{x}_i)$  and examining the critical points where the vertical timing line of each vehicle intersects the left extremes of its coordination function.

### Loose Sequencing Constraints

Loose sequencing can be described as having desired arrival time windows for each vehicle on the team. The lower plot of Figure 6 depicts coordination functions with the vertical bars indicating acceptable time windows for each vehicle on the team. Loose sequencing constraints can be stated as

$$\begin{aligned} T_s &\leq T_1 \leq T_s + \tau_1 \\ T_s + \Delta_i &\leq T_i \leq T_s + \Delta_i + \tau_i \quad i = 2, \dots, N, \end{aligned}$$

where  $\Delta_i$  is the time interval between the opening of the first time window and the opening of the  $i^{\text{th}}$  time window and  $\tau_i$  indicates the duration of the  $i^{\text{th}}$  time window. The coordination variable is given by  $\theta = T_s$  and  $f_i$  in Equation (6) must be modified to

$$f_i(\mathbf{x}_i, \mathbf{u}_i) = L(\mathcal{W}_i)/v_i - \Delta_i - \sigma_i,$$

where  $\sigma_i \in [0, \tau_i]$  is a slack variable. In this case, the team optimal arrival time for one of the vehicles will occur when the right side of its time window intersects the left extreme of a piecewise continuous segment of its coordination function. Team optimal times for the other vehicles will either occur at the left side of their windows or at discontinuities in their coordination functions inside their time windows. Searching through these options to find the optimum is straightforward and fast.

In general, timing constraints for simultaneous arrival, tight sequencing, and loose sequencing can be stated in the form

$$T_s + \Delta_i \leq T_j \leq T_s + \Delta_i + \tau_i \quad i = 1, \dots, N, \quad (10)$$

where  $\Delta_1 = 0$  and  $\Delta_i$  and  $\tau_i$  are specified for each of the  $N$  vehicles composing the team. For loose sequencing,  $\Delta_i$  and  $\tau_i$  are positive constants. For tight sequencing,  $\Delta_i$  are positive constants and  $\tau_i = 0$ . For simultaneous arrival,  $\Delta_i = \tau_i = 0$ . The timing constraints of Equation (10), can be written as

$$\begin{aligned} T_s + \Delta_i - T_i &\leq 0 \\ T_i - T_s - \Delta_i - \tau_i &\leq 0 \quad i = 1, \dots, N \end{aligned}$$

which is of the general form  $\mathbf{c}(T_1, \dots, T_N) \leq 0$ . This formulation for timing constraints is inherently flexible and can accommodate a mixture of simultaneous arrival, tight sequencing, and loose sequencing constraints in the same mission.

# Waypoint Path Planning

The cooperative timing solutions described in the previous section depend upon the availability of a set of feasible decision vectors

$$\mathcal{U}_i(\mathbf{x}_i) = \left\{ \begin{pmatrix} v_i \\ \mathcal{W}_i \end{pmatrix} \right\},$$

where  $v_i \in [v_{\min}, v_{\max}]$  and  $\mathcal{W}_i = \{\mathbf{w}_{i1}, \dots, \mathbf{w}_{iP}\}$  is a waypoint path, given the current decision state

$$\mathbf{x}_i = \begin{pmatrix} \mathbf{z}_{i0} \\ \mathbf{z}_{if} \\ \mathcal{H}_i \end{pmatrix},$$

where  $\mathbf{z}_{i0}$  is the start location,  $\mathbf{z}_{if}$  is the target location, and  $\mathcal{H}_i = \{\mathbf{h}_1, \dots, \mathbf{h}_M\}$  is the set of known threat locations. The role of the Waypoint Path Planner (WPP) is to generate a sufficiently rich set of feasible decision variable vectors. This section briefly describes our approach to the waypoint path planner. More detailed descriptions appear in Refs. 18 and 19.

The WPP is called at the beginning of a specific mission and at other event driven instances during the mission, such as when a target is reached or when a previously unknown threat is detected. The WPP consists of three stages. In the first stage, a Voronoi graph is constructed using the initial and destination points and the known threat locations. In the second stage, costs are assigned to each edge of the Voronoi graph. In the third stage, paths from the present location to the desired destination are generated from a  $k$ -best path search.

For a battle area having  $M$  threats, the Voronoi graph partitions the battle area into  $M$  convex polygons or cells. Each cell contains one threat and every location within a cell is closer to the enclosed threat than to any other. The edges of the Voronoi graph represent lines that are equidistant from the two closest neighboring threats. Therefore, the graph edges maximize the distance from the two closest threats. The Voronoi graph also contains initial and final locations within cells to ensure that threats will be avoided when joining and leaving the graph. The current location and destination of the UAV are nominally not on the graph. To connect them to the graph, edges are connected between the start and end points and the adjacent nodes. Figure 7 shows a Voronoi diagram created for a set of specified threats, UAV location, and target location.

To search the graph, the cost associated with traversing each edge must be determined. Edge costs are assigned by using the myopic cost metric introduced in Equation (7), where the  $J_{\text{threat}}$  and  $J_{\text{fuel}}$  given in Equations (8) and (9) are applied separately to each edge.

With the costs for traversing an edge defined and the start and destination locations joined to the graph, the graph is searched using Eppstein's  $k$ -best paths algorithm.<sup>26</sup> Eppstein's algorithm is similar to Dijkstra's algorithm with the exception that the  $k$ -best paths are found rather than simply the best path. Modifications have been made to Eppstein's algorithm to allow edges to be traversed in both directions.



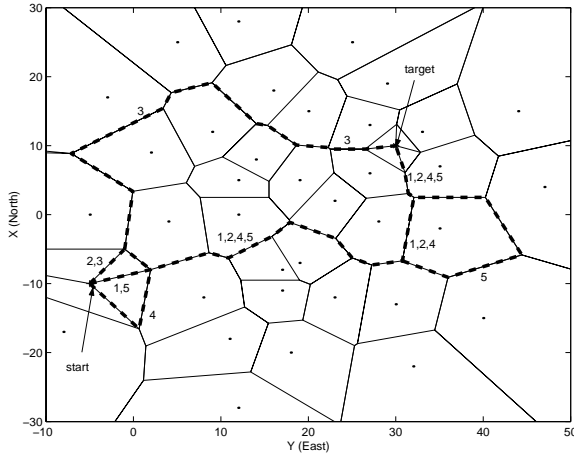


Figure 7: Threat-based Voronoi diagram. The threats are represented by dots.

Paths traversing an edge in one direction and then immediately in the opposite direction are not allowed, although edges can be traversed multiple times in the same path if the algorithm dictates that this is cost effective. The parameter  $k$  is used to specify the number of paths  $\mathcal{W}_{ij}$  in  $\mathcal{U}_i(\mathbf{x}_i)$ . Therefore

$$\mathcal{U}_i(\mathbf{x}_i) = [v_{\min}, v_{\max}] \times \{\mathcal{W}_{i1}, \dots, \mathcal{W}_{ik}\}.$$

Figure 7 shows the five best paths resulting from the WPP. Paths 1, 2, and 4 are very similar with the only differences occurring near the start point. Paths 1 and 5 are identical with the exception of the detour in path 5 near the end. Path 3 takes an altogether different route from the other paths.

Waypoint path planning based on the Voronoi graph is computationally efficient. Generation and searching of a Voronoi graph with 100 threats takes about 10 ms on a desktop personal computer. A primary advantage of the Voronoi graph approach is that it reduces the path planning problem from one having a continuum of solutions (that are infinite in number) to a problem having a finite number of threat-avoiding solutions. The efficiency of the method is directly associated with this reduction in the solution space. This abstraction makes the waypoint path planning problem feasible in real time. One disadvantage of the WPP is that it produces straight-line paths that cannot be flown accurately by a UAV. This problem is overcome by the dynamic trajectory smoother (DTS), which is discussed next.

## Dynamic Trajectory Smoothing

As shown in Figure 1, the Coordination Manager, using the WPP, produces waypoint paths for each vehicle that are both low in cost and satisfy the timing constraints. The objective of the Dynamic Trajectory Smoother (DTS) is to smooth the straight-line waypoint paths into time-parameterized trajectories that are flyable by the UAV.

Since the UAV will be operating in a dynamic environment with popup and dynamically moving threats, the smoothing process must take place in real-time. The waypoint paths have been chosen to satisfy timing constraints, therefore the trajectories must be smoothed so that the resulting path length is identical to the waypoint path. In addition, since individual and team objectives are based on the cost of the waypoint paths, the smoothed trajectory must deviate, as little as possible from the waypoint paths produced by the coordination manager/WPP.

In this paper we assume that the UAV is flying at constant altitude and is equipped with trajectory tracking capability. The input to the DTS is a constant feasible velocity  $v_i^c \in [v_{\min}, v_{\max}]$ , and a constant waypoint path

$$\mathcal{W}_i = \{\mathbf{w}_{i1}, \mathbf{w}_{i2}, \dots, \mathbf{w}_{iP}\},$$

where  $\mathbf{w}_{ij} \in \mathbb{R}^2$  denote the waypoints expressed in inertial coordinates.

The essential idea of our approach is to give the trajectory generator a mathematical structure that captures the dominant dynamic behavior of the UAV. In particular the DTS is given by the differential equations

$$\begin{aligned} \dot{z}_{ix}^c &= v_i^c \cos \psi_i^c \\ \dot{z}_{iy}^c &= v_i^c \sin \psi_i^c \\ \dot{\psi}_i^c &= u_i \end{aligned} \tag{11}$$

where  $u_i \in [-c, c]$  is chosen to minimize the deviation from  $\mathcal{W}_i$  and to ensure that the trajectory has the same path length as  $\mathcal{W}_i$ . Although relatively simple, Equations (11) effectively model the heading and position states of a UAV in level, constant-speed flight. The DTS equations are solved via a fixed-step ODE solver and are propagated in real-time. If a fourth-order Runge-Kutta algorithm<sup>27</sup> is used then  $u_i$  is computed four times each sample period. Therefore, the computational complexity is partially dependent upon the computation of  $u_i$ .

Note that if  $u_i = +c$ , then the DTS given in Equation (11) traces out a right-handed circle as shown in Figure 8. Similarly, if  $u_i = -c$  then the DTS traces out a left-handed circle. As shown in Figure 8, the local reachability region of the DTS is bounded by these two circles. The radius of the circles defining the local reachability region is given by  $R_i = \frac{v_i^c}{c}$ . Note that as the desired velocity increases, the minimum turning radius increases.

Consider the problem of turning from one waypoint path segment onto another in minimum time at constant velocity. If the trajectory is not constrained to pass through the waypoint connecting the two segments, then the time-optimal trajectory connecting the segments is indicated by the red line of Figure 9. If the trajectory is constrained to pass through the intermediate waypoint, then the time-optimal trajectory between the segments is indicated by the blue line.

One of the disadvantages of both minimum-time transitions and transitions constrained to go through the waypoint is that the trajectories will have different path lengths than the original Voronoi path. Since the Voronoi path is used for determining intercept times, it is important for the smoothed trajectory to have the same length as the original Voronoi path.

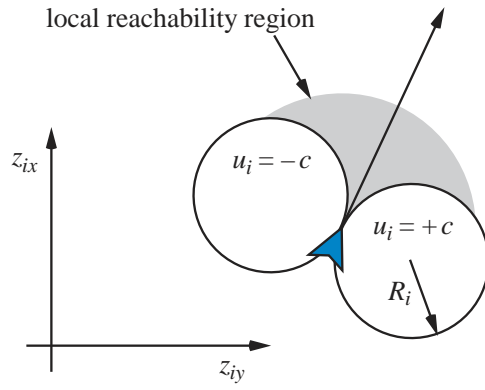


Figure 8: Local reachability region of the DTS.

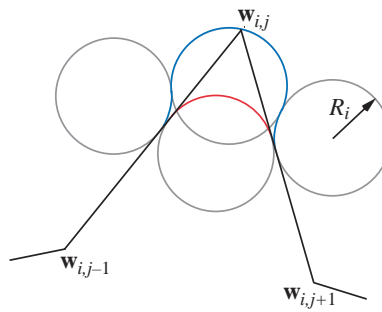


Figure 9: Time-optimal transitions between waypoint path segments.

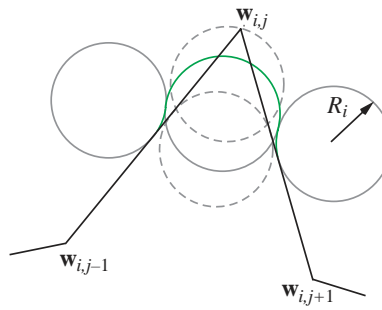


Figure 10: Length-matching transitions between waypoint path segments.

From Figure 9, it is clear that the path length of the minimum-time trajectory is shorter than the path length of the Voronoi path, while the path length of the constrained trajectory is longer than the Voronoi path. As Figure 10 shows, by positioning the transition circle between the inscribed circle and the circle that intersects the waypoint, a transitioning trajectory can be determined that has the same length of the original Voronoi path. Details of this trajectory generation strategy and algorithms for implementing it can be found in Refs. 28 and 6. The trajectory smoothing approach taken here is computationally efficient. On average, one step of the smoothing algorithm requires about 40  $\mu$ sec on a desktop computer. The output of the trajectory generator is a smooth trajectory  $(\hat{z}_{ix}(t), \hat{z}_{iy}(t))$  for the UAVs to follow, calculated in real time as they move between waypoints.

## Simulation Results

Simulation results are presented for a team of three UAVs flying three different missions: simultaneous arrival, tight sequencing, and loose sequencing. In each mission, there is one target and 33 threats distributed over a 5 km square battle area. The objective is to avoid the threats while meeting the timing constraints imposed for the mission. Collision avoidance is not treated explicitly and is achieved by flying the UAVs at different altitudes.

For the simulations, small 6 foot wingspan UAVs are simulated using 12<sup>th</sup>-order nonlinear dynamic models. The DTS on each UAV produces desired heading, velocity, and position commands to be tracked to accomplish the cooperative timing objectives. Commands are tracked using a trajectory tracking algorithm based on the satisficing framework.<sup>29,30</sup> This approach guarantees asymptotic tracking under the velocity and heading rate constraints imposed on the UAV. One particular advantage of this approach for real time implementation is its computational simplicity.

### Simultaneous Arrival Constraints

Simulation results for the simultaneous arrival mission are presented in Figure 11. The trajectories taken to the target are indicated by dotted lines. The larger dots along each trajectory indicate 50 second intervals. The initial jog in the path of the blue UAV is a low-risk deviation that enables simultaneous arrival at the target with the other team members.

The desired arrival time for the team determined by the Coordination Manager was 450.1 seconds. Figure 12 shows range-to-target information for each of the three missions. From the simultaneous arrival results in the upper plot it can be seen that the UAVs arrived at the target at 450 seconds as commanded. The small errors in the arrival time that do exist can be attributed to tracking errors by the UAVs.

Coordination functions for the team for each of the missions are shown in Figure 13 (paths resulting in arrival times outside  $\bigcap_{i=1}^N \Theta_i(\mathbf{x}_i)$  have been deleted). Each line segment represents one possible trajectory for a UAV. In the upper plot, the team-optimal ETA for simultaneous arrival is indicated by the black vertical line. It

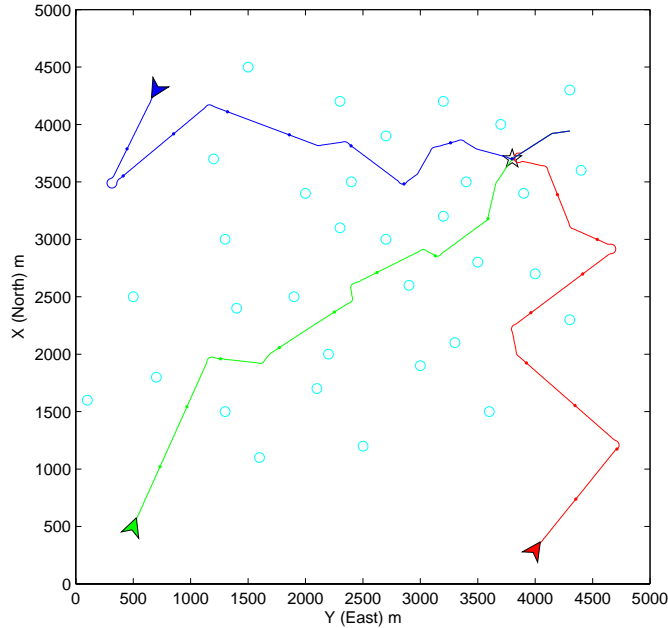


Figure 11: Simultaneous arrival paths to target.

can be seen that the optimal ETA occurs at the left extremes of a coordination function segment for UAV 3 (blue). From an individual, myopic perspective, this team ETA is optimal for UAV 3, close to optimal for UAV 1 (red), and suboptimal for UAV 2 (green). Considering the entire team however, the indicated ETA is most cost effective. In examining Figure 13, it is important to recognize that the coordination functions represent the essential information for good team timing decisions. Their value lies in the compactness of this representation.

## Tight Sequencing Constraints

Simulation results for tight sequencing are shown in Figure 14 where the UAVs are required to arrive at the target with a 40 second interval between UAV 1 and UAV 2 and a 50 second interval between UAVs 2 and 3. Comparing with simultaneous arrival case of Figure 11, it can be seen that UAV 1 and UAV 3 take the same paths, while UAV 2 takes a slightly longer, but less costly route.

The desired arrival times produced by the Coordination Manager for the team are 449.4, 489.4, and 539.4 seconds. The range data in the middle plot of Figure 12 shows that these times are met very closely. The middle plot of Figure 13 shows coordination function information for the tight sequencing mission. Vertical colored lines indicate the desired ETAs for each of the UAVs. Note that each line segment on the coordination function plot corresponds to a unique path. The width of each segment corresponds to a range of speeds that can be flown by the UAV. From this information, it can be seen that UAV 1 flies virtually the same trajectory as in the simultaneous arrival case, UAV 2 flies a slower, less costly path, and UAV 3 flies

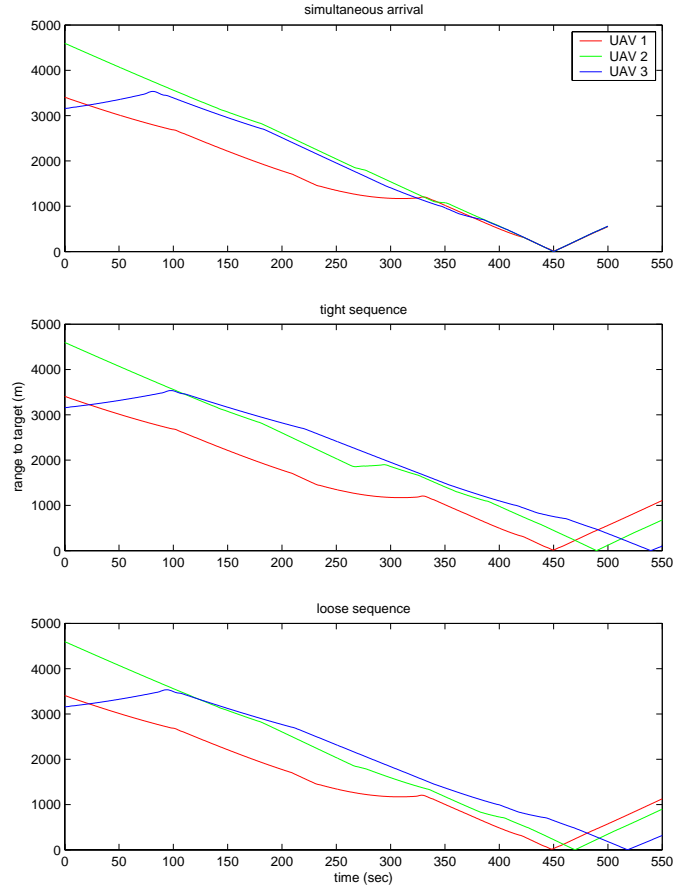


Figure 12: Range to target for missions.

the same path at a much slower velocity. The optimum occurs at a left extreme of a coordination function segment for UAV 2 (489.4 seconds). The arrival time for UAV 1 is offset 40 seconds earlier, while the arrival time for UAV 3 is offset 50 seconds later.

## Loose Sequencing Constraints

The loose sequencing constraints give the UAVs flexibility in their arrival times through the use of acceptable time windows. For the problem considered, the opening of the second arrival time window occurs 40 seconds after the first, while the opening of the third arrival time window occurs 50 seconds after the second. The first and second windows are 20 seconds wide, while the third window is 30 seconds wide. Figure 15 shows simulation results for the loose sequencing scenario. As before, UAV 1 and UAV 3 fly the same paths (although at different velocities) as in the other cases, but UAV 2 flies a different path. The additional flexibility provided by the time windows allows it to choose a lower-cost path.

The desired arrival times determined by the Coordination Manager are 448.1, 469.7, and 518.1 seconds. The bottom plot of Figure 12 shows range to target data

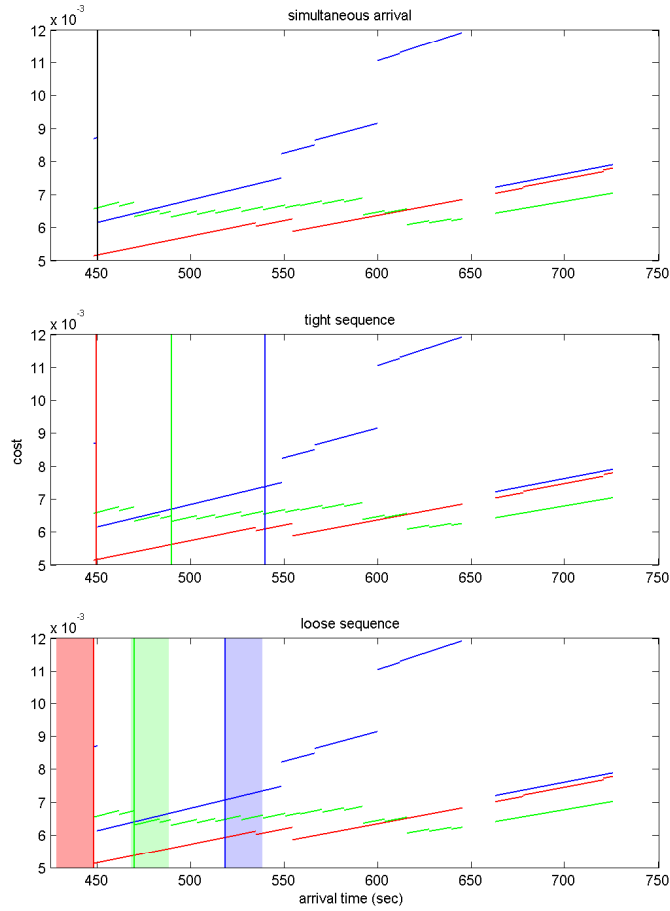


Figure 13: Coordination functions for missions.

for the UAVs that confirms arrival at times very close to those desired. The bottom plot of Figure 13 shows coordination function information for loose sequencing. In this case, the shaded regions show the acceptable time windows for each UAV, while the colored vertical lines indicate the desired arrival time for each vehicle. For UAV 1 the desired arrival time is at the upper limit of the time window, while for UAV 3 the desired arrival time is at the lower limit of the time window. By making their arrival times as close as the windows will allow, the cost to the team is minimized. For UAV 2, the minimum cost lies on the interior of the time window at the left extreme of a coordination function segment rather than the lower or upper bound. Clearly, the flexibility provided by time windows in the loose sequencing scenario results in a lower cost solution than the tight sequencing case.

## Conclusions

In this paper we have outlined a cooperative control strategy based on coordination functions and coordination variables. While sufficiently general to address a wide

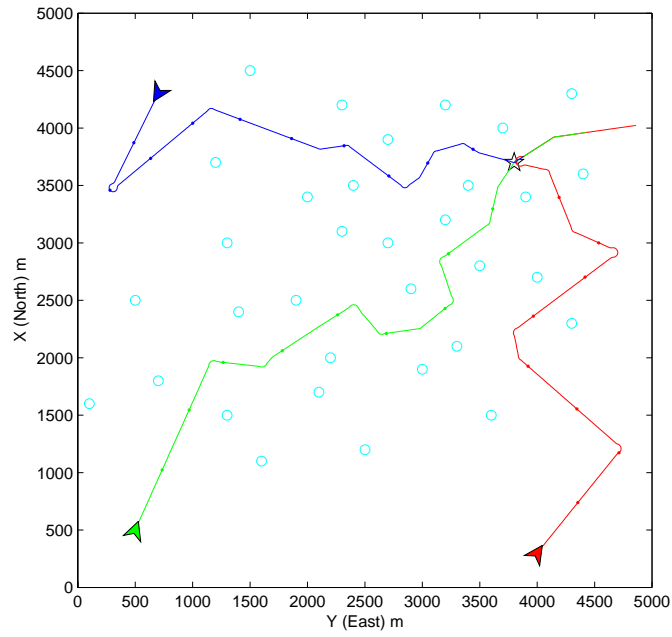


Figure 14: Tight sequencing paths to target.

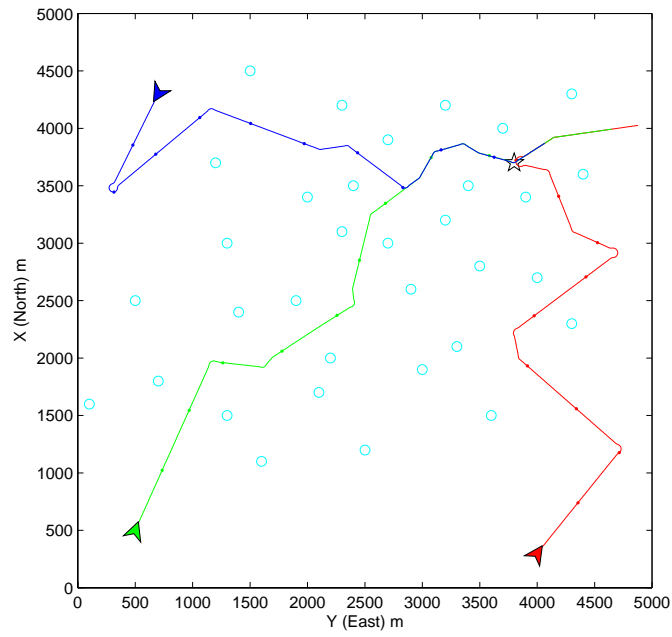


Figure 15: Loose sequencing paths to target.



range of problems, we have applied the approach to cooperative trajectory planning problems involving timing constraints. Simultaneous arrival, tight sequencing, and loose sequencing constraints can each be accommodated using the cooperative control algorithms and constraint formulations developed. Primary advantages of the approach include: (1) the distillation of information essential for cooperation leading to low communication demands and (2) the efficient formulation and solution of the team-optimal cooperation problem leading to real-time implementation on hardware platforms.

## **Acknowledgments**

This work was supported by AFOSR Grant No. F49620-01-1-0091. The authors wish to thank Phillip Chandler of the Air Force Research Laboratory, Air Vehicles Directorate for his introduction to cooperative timing problems for UAVs and for his periodic influential input on this topic.

## Nomenclature

$(\cdot)_i$	variable pertaining to the $i^{\text{th}}$ UAV
$c_{\text{fuel}}$	fuel consumption constant
$f_i$	map from decision variable to coordination variable
$\mathcal{H}_i$	known threat locations
$\mathbf{h}$	threat location
$J_i$	UAV myopic performance objective
$J_{\text{fuel}}$	waypoint path fuel cost
$J_{\text{threat}}$	waypoint path threat cost
$\mathcal{J}_T$	team performance objective
$L$	waypoint path length
$R_i$	turn circle radius
$T_i$	arrival time
$T_s$	time of first arrival for team
$\mathcal{U}_i$	decision space
$\mathbf{u}_i$	decision variable
$u_i$	heading rate input
$v_{\text{min}}$	UAV minimum velocity
$v_{\text{max}}$	UAV maximum velocity
$v_i$	UAV velocity
$v_i^c$	UAV velocity command
$\mathcal{W}_i$	waypoint path
$\mathbf{w}_{ij}$	waypoint
$\mathcal{X}_i$	situation state space
$\mathbf{x}_i$	situation state
$\mathbf{z}_{i0}$	current UAV position
$\mathbf{z}_{if}$	UAV destination position
$z_{ix}^c$	inertial $x$ position command
$z_{iy}^c$	inertial $y$ position command
$\Delta_i$	arrival time spacing interval
$\kappa$	cost function weighting factor
$\phi_i$	coordination function
$\psi_i^c$	heading command
$\Theta_i$	set of feasible coordination variable values
$\theta$	coordination variable
$\tau_i$	arrival time window width

## References

- [1] Unmanned aerial vehicle roadmap 2002–2007. Technical report, Office of the Secretary, U.S. Department of Defense, December 2002.
- [2] Emilio Frazzoli, Munther A. Dahleh, and Eric Feron. Real-time motion planning for agile autonomous vehicles. In *Proceedings of the American Control Conference*, pages 43–49, Arlington VA, June 2001.

## REFERENCES

- [3] Nadeem Faiz, Sunil K. Agrawal, and Richard M. Murray. Trajectory planning of differentially flat systems with dynamics and inequalities. *AIAA Journal of Guidance, Control and Dynamics*, 24(2):219–227, March–April 2001.
- [4] Oleg A. Yakimenko. Direct method for rapid prototyping of near-optimal aircraft trajectories. *AIAA Journal of Guidance, Control and Dynamics*, 23(5):865–875, September–October 2000.
- [5] Guang Yang and Vikram Kapila. Optimal path planning for unmanned air vehicles with kinematic and tactical constraints. In *Proceedings of the IEEE Conference on Decision and Control*, pages 1301–1306, Las Vegas NV, December 2002.
- [6] Erik P. Anderson and Randal W. Beard. An algorithmic implementation of constrained extremal control for UAVs. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Monterey, CA, August 2002. AIAA Paper No. 2002-4470.
- [7] Gokhan Inalhan, Dusan M. Stipanovic, and Claire J. Tomlin. Decentralized optimization with application to multiple aircraft coordination. In *CDC*, pages 1147–1155, Las Vegas NV, 2002.
- [8] John Bellingham, Michael Tillerson, Arthur Richards, and Johnathan P. How. Multi-task allocation and path planning for cooperating UAVs. In *Cooperative Control: Models, Applications and Algorithms*, pages 1–19. Conference on Coordination, Control and Optimization, November 2001.
- [9] A. Richards, J. Bellingham, M. Tillerson, and J. How. Coordination and control of UAVs. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pages AIAA–2002–4588, Monterey, CA, August 2002.
- [10] Wei Kang and Hsi-Han Yeh. Coordinated attitude control of mult-satellite systems. *International Journal of Robust and Nonlinear Control*, 12:185–205, 2002.
- [11] W. Kang, N. Xi, and Andy Sparks. Formation control of autonomous agents in 3D workspace. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1755–1760, San Francisco, CA, April 2000.
- [12] Shahab Sheikholeslam and Charles A. Desoer. Control of interconnected nonlinear dynamical systems: The platoon problem. *IEEE Transactions on Automatic Control*, 37(6):806–810, June 1992.
- [13] P. K. C. Wang and F. Y. Hadaegh. Coordination and control of multiple microspacecraft moving in formation. *The Journal of the Astronautical Sciences*, 44(3):315–355, 1996.
- [14] Randal W. Beard, Jonathan Lawton, and Fred Y. Hadaegh. A feedback architecture for formation control. *IEEE Transactions on Control Systems Technology*, 9(6):777–790, November 2001.
- [15] Jonathan Lawton and Randal Beard. A projection approach to spacecraft formation attitude control. In *23rd Annual AAS Guidance and Control Conference*, Breckenridge, Colorado, February 2000. Paper no. AAS 00-011.

## REFERENCES

- [16] Naomi Ehrich Leonard and Edward Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *Proceedings of the IEEE Conference on Decision and Control*, pages 2968–2973, Orlando, Florida, December 2001.
- [17] Timothy McLain, Phillip Chandler, Steven Rasmussen, and Meir Pachter. Cooperative control of UAV rendezvous. In *Proceedings of the American Control Conference*, pages 2309–2314, June 2001.
- [18] Timothy McLain and Randal Beard. Cooperative rendezvous of multiple unmanned air vehicles. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Denver, CO, August 2000. Paper no. AIAA-2000-4369.
- [19] Randal W. Beard, Timothy W. McLain, Michael Goodrich, and Erik P. Anderson. Coordinated target assignment and intercept for unmanned air vehicles. *IEEE Transactions on Robotics and Automation*, 18(6):911–922, December 2002.
- [20] Randal W. Beard, Timothy W. McLain, and Michael Goodrich. Coordinated target assignment and intercept for unmanned air vehicles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2581–2586, Washington DC, May 2002.
- [21] Jan Roskam. *Airplane Flight Dynamics and Automatic Flight Controls*. Design, Analysis, and Research Corporation, 2001.
- [22] Brian L. Stevens and Frank L. Lewis. *Aircraft Control and Simulation*. John Wiley & Sons, Inc., 1992.
- [23] Derek Kingston, Randal Beard, Timothy McLain, Michael Larsen, and Wei Ren. Autonomous vehicle technologies for small fixed-wing UAVs. In *AIAA 2nd Unmanned Unlimited Systems, Technologies, and Operations Conference*, San Diego, CA, September 2003.
- [24] Timothy W. McLain, Randal W. Beard, and Jed Kelsey. Experimental demonstrations of multiple robot cooperative target intercept. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Monterey, CA, August 2002. AIAA Paper No. 2002-4678.
- [25] Scott A. Bortoff. Path planning for UAVs. In *Proceedings of the American Control Conference*, pages 364–368, Chicago IL, June 2000.
- [26] David Eppstein. Finding the  $k$  shortest paths. *SIAM Journal of Computing*, 28(2):652–673, 1999.
- [27] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. PWS-KENT Publishing Company, Boston, 4th edition, 1988.
- [28] Erik P. Anderson. Constrained extremal trajectories and unmanned air vehicle trajectory generation. Master’s thesis, Brigham Young University, Provo, Utah 84602, April 2002.
- [29] Wei Ren and Randal W. Beard. CLF-based tracking control for UAV kinematic models with saturation constraints. In *Proceedings of the IEEE Conference on Decision and Control*, 2003. To appear.

## REFERENCES

- [30] Wei Ren and Randal W. Beard. Trajectory tracking for unmanned air vehicles with velocity and heading rate constraints. *IEEE Transactions on Control Systems Technology*, To appear.