

Control of Multiple UAVs for Persistent Surveillance: Algorithm and Flight Test Results

Nikhil Nigam, Stefan Bieniawski, Ilan Kroo, and John Vian, *Senior Member, IEEE*

Abstract—Interest in control of multiple autonomous vehicles continues to grow for applications such as weather monitoring, geographical mapping fauna surveys, and extra-terrestrial exploration. The task of persistent surveillance is of particular significance in that the target area needs to be continuously surveyed, minimizing the time between visitations to the same region. This distinction from one-time coverage does not allow a straightforward application of most exploration techniques to the problem, though ideas from these methods can still be used. The aerial vehicle dynamic and endurance constraints add additional complexity to the autonomous control problem, whereas stochastic environments and vehicle failures introduce uncertainty. In this work, we investigate techniques for high-level control, that are scalable, reliable, efficient, and robust to problem dynamics. Next, we suggest a modification to the control policy to account for aircraft dynamic constraints. We also devise a health monitoring policy and a control policy modification to improve performance under endurance constraints. The Vehicle Swarm Technology Laboratory—a hardware testbed developed at Boeing Research and Technology, Seattle, WA, for evaluating a swarm of unmanned air vehicles—is then described, and these control policies are tested in a realistic scenario.

Index Terms—Control, coordination, exploration, flight test, hardware, multiple UAVs, persistent surveillance, refueling, swarm, unmanned air vehicle (UAV).

I. INTRODUCTION

RESEARCH interest for control and coordination of autonomous vehicles has shown an increase in the fields of artificial intelligence (AI) and controls [1], [2]. In particular, the task of search/exploration/coverage has received significant attention in the past two decades [3]–[5]. dynamic programming (DP) [6]–[8], mixed integer linear programming (MILP) [9], and receding horizon control [10], [11] are popular planning based methods. Traditional AI search techniques, such as A* and its variants have also been applied [12], [13], but do not address the problem of cooperation among multiple vehicles. A

vast amount of literature deals with problems involving obstacles and sensing uncertainties [14]. Space decomposition-based methods (such as boustrophedon [15], morse [16], and voronoi [17]) have proven effective in dealing with such problems [18]. Spanning tree coverage (STC) [19], [20] algorithms claim optimality for single robot coverage problems, but have no such guarantees for multiple robots. Market-based mechanisms have also been used to divide work among vehicles [21], [22], but their applications have been limited. At the other end of the spectrum are coordination field methods, that include particle swarm optimization, potential functions-based approaches, and digital pheromone mechanisms [23], [24]. They are simple and highly scalable, but often suffer from problems of local minima. Other novel approaches include work by Tumer *et al.* [25], [26], using neural networks to represent control policies.

Most of these methods for high-level vehicle control can be classified into two categories. One class is characterized by approaches with a formal derivation or proof of optimality but not scalable to a large number of vehicles [27], [28]. The other class includes approaches that are decentralized and scalable but heuristic [25], [29]. Some of the techniques cannot be applied in an online setting and may not be useful for sensor-based coverage. Certain literature has addressed problems similar to persistent surveillance [12], [13], [30], [31], but the application of most of these techniques to this particular problem is not straightforward.

Using unmanned air vehicles (UAVs) has gained considerable attention owing to their relative indifference to terrain and greater range compared to ground vehicles [32]. Bellingham *et al.* [9], [28], [33] study the problem of cooperative path planning and trajectory optimization. Flint *et al.* [27] study cooperative search. Lawrence, Donahue, Mohseni and Han [23] use large micro air vehicle (MAV) swarms for toxic plume detection. Reference [34] claims that the design of control laws for aerial vehicles ties in with the aircraft dynamic constraints, though most existing work ignores this coupling.

Long duration missions often require the UAVs to return to base-stations periodically for refueling. Mei *et al.* [35] survey techniques for energy and time constrained problems while addressing a deployment problem. Sujit and Ghose [8] also study a surveillance problem under endurance constraints. But, most studies do not address the issue of refueling for a long duration mission.

On the experimental side, there have been several demonstrations using robots in indoor environments [36]–[38], with emphasis on localization and mapping. But actual experiments using aerial vehicles are few. Frew *et al.* [39] have demonstrated road following, obstacle avoidance, and convoy pro-

Manuscript received September 24, 2010; revised March 29, 2011; accepted June 27, 2011. Manuscript received in final form August 31, 2011. Date of publication October 17, 2011; date of current version June 28, 2012. Recommended by Associate Editor C. A. Rabbath. This work was supported by Boeing Research and Technology, The Boeing Company, Seattle, WA, USA.

N. Nigam was with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA USA. He is now with Intelligent Automation Inc., Rockville, MD 20855 USA (e-mail: nnigam@i-a-i.com).

S. Bieniawski and J. Vian are with Boeing Research and Technology, The Boeing Company, Seattle, WA 98108 USA (e-mail: stefan.r.bieniawski@boeing.com; john.vian@boeing.com).

I. Kroo is with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305 USA (e-mail: kroo@stanford.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCST.2011.2167331

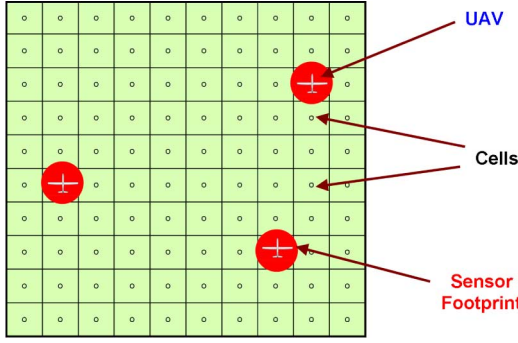


Fig. 1. Illustration showing the target space gridded using cells. Each UAV has a circular sensor footprint that equals the grid cell size.

tection using two UAVs. How *et al.* [40], [41] have demonstrated surveillance on the MIT RAVEN testbed, whereas Jodeh *et al.* [42] give an overview of flight demonstrations conducted by AFRL with surveillance and reconnaissance objectives. In hardware implementations, reliability and robustness also become major issues. Koenig *et al.* [12] have conducted some robustness studies in simulation with small disturbances. Enns *et al.* [43] provide simulation results using algorithms robust to winds, modeling errors, and failures. There have been a few other studies on robustness in hardware implementations [38] as well, but they tend to severely compromise on the efficiency of exploration.

A. Our Work

The novelty of this work lies in development of a control policy that directly addresses the persistence problem as defined below. The technique combines the benefits of formal derivation-based and scalable heuristic methods while maintaining performance close to the optimum. We also study the coupling between control policy and dynamic and endurance constraints suggesting modifications for the same, that has been hardly addressed in literature. Finally, we present flight test results highlighting the ability to maintain good performance even under realistic limitations and failures.

For the purpose of studying the control strategies, a simple 2-D target space (physical area to be surveyed) is used. It is gridded using an approximate cellular decomposition,¹ and each cell has an associated *age*, that is the time elapsed since it was last observed. Fig. 1 shows the target space with three UAVs surveying the region. The goal is to minimize the maximum age over all cells that is observed over a long period of time. This is equivalent to leaving no area of the target space unexplored for a long duration.²

For high-level control of a single UAV, we devise a semi-heuristic control policy that is optimum for a simple case, and extend it to a more realistic sensor-based coverage problem. The approach is compared to a heuristic potential function-based

¹This means that the sensor footprint (circular in our case) equals the cell size [4]. It is also assumed that the cells completely cover the target space.

²Note that this is different from a one-time coverage problem, where it is only necessary to know if a cell has been explored or not. It also differs from problems of minimizing map uncertainty, where the cumulative uncertainty (as opposed to maximum uncertainty of a cell) is the quantity of interest.

method, a DP-based planning method, and a bound on the optimum performance. An approach for multiple-UAV coordination is then developed, that is an extension of the reactive policy for a single UAV.

We look at the effect of aircraft dynamics (using a 3-DOF dynamics model) on the mission performance next. The aircraft is assumed to fly at constant altitude and speed to simplify the dynamics model and study the coupling between the control policy and aircraft dynamics. A simple modification to the control policy is then proposed, to improve mission performance under dynamic constraints. A minimum-length trajectory tracking controller used for this purpose is briefly described and implemented. To address the issue of limited endurance, a health monitoring policy is derived, that involves solving a linear program (LP) by each UAV to decide when to return for refueling, and another control policy modification is suggested, that results in desirable coordination between UAVs operating under endurance constraints. This approach is compared to two benchmark techniques—a pre-fixed health monitoring policy and a simple reactive technique—and performance improvement is analyzed as a function of mission specifications.

Next, to verify our claims—ease of implementation, low communication requirements, reliability, and robustness—we demonstrate flight test results. The technique is implemented on small quad-rotor vehicles flown at the Vehicle Swarm Technology Laboratory (VSTL) developed by Boeing Research and Technology. Using this setup, the feasibility of this approach is demonstrated for up to 4 UAVs. The concept of “persistence” under failures and limited endurance is also shown using up to 2 UAVs. Subsequently, a simulation of VSTL environment is used to study some cases with larger target spaces and up to 8 UAVs. Finally, we draw conclusions about our study and outline future work.

II. CONTROL OF A SINGLE UAV

A. Policy Structure

In order to find a control strategy for a single UAV, a problem with two cells is considered. The cells need to be visited so as to minimize the maximum of the ages observed for both of them. The UAV, stationed at distance d_1 from cell 1 and d_2 from cell 2 (see Fig. 2), is assumed to travel at constant velocity, V_{mission} , and A_j denotes the age of the j th cell. Our aim is to find an optimal control policy for this problem first.

The UAV can choose first to go either to cell 1 or cell 2 first. After the UAV has chosen a cell to observe first, the optimum policy is to keep moving back and forth between the cells. Hence a single action defines the optimum policy in this case.³ Assuming $A_1 \leq A_2$, we can construct a plot of the ages of cells for the case where the UAV chooses to go to cell 1 (Fig. 3) or cell 2 first. This is used to identify the maximum age (over the two cells) as a function of time, and our optimum policy tries to minimize the peak of this maximum age curve.

Let A_{max_1} and A_{max_2} denote the peaks of the maximum age curves when the UAV chooses cell 1 and cell 2, respectively.

³It is easily seen that any policy that makes a UAV start heading to a cell first and then turn back to go to the other cell is necessarily suboptimal.

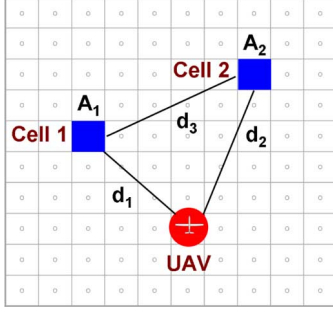


Fig. 2. Simple two-cell problem used to derive structure for control policy.

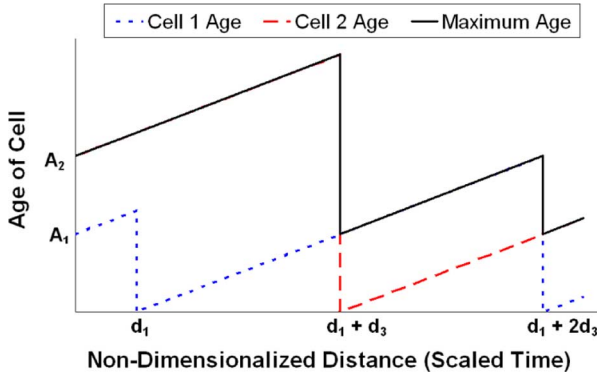


Fig. 3. Plot of the maximum age curve (i.e., the maximum of the ages of the two cells, as a function of time/scaled-distance) when the UAV chooses to go to cell 1 first.

This decision depends on the initial ages of the cells and the distance of the UAV from the cells, as illustrated in

$$\begin{aligned}
 \text{Choose cell 1} &\Leftrightarrow A_{\max_1} \leq A_{\max_2} \\
 &\Leftrightarrow \max \left\{ \left(A_2 + \frac{d_1 + d_3}{V_{\text{mission}}} \right), \left(\frac{2d_3}{V_{\text{mission}}} \right) \right\} \\
 &\leq \max \left\{ \left(A_2 + \frac{d_2}{V_{\text{mission}}} \right), \right. \\
 &\quad \left. \left(A_1 + \frac{d_2 + d_3}{V_{\text{mission}}} \right), \left(\frac{2d_3}{V_{\text{mission}}} \right) \right\}. \quad (1)
 \end{aligned}$$

These inequalities (and ones for $A_1 \geq A_2$) are solved for all possible cases, resulting in

$$\begin{aligned}
 &\text{If } A_1 \leq \frac{d_1}{V_{\text{mission}}}, \quad \text{ignore cell 1} \\
 &\text{If } A_2 \leq \frac{d_2}{V_{\text{mission}}}, \quad \text{ignore cell 2} \\
 &\text{If neither is true, choose cell 1} \\
 &\Leftrightarrow A_1 - \frac{d_1}{V_{\text{mission}}} \geq A_2 - \frac{d_2}{V_{\text{mission}}}. \quad (2)
 \end{aligned}$$

More concisely, the control policy can be expressed by defining a value associated with each cell, as in (3). Here, \mathcal{V}_j is the value of the j th cell, w_0 is a weighting parameter ($w_0 = (-1/V_{\text{mission}})$ for this two-cell case), and δ_{1j} is the distance between the UAV and j th cell.

$$\mathcal{V}_j = \max\{(A_j + w_0 \delta_{1j}), 0\}. \quad (3)$$

The UAV calculates the value for both cells and goes to the cell with the maximum value.

B. Extension to Multiple-Cell Case

The policy structure derived above, can be extended to the more realistic 2-D multiple cell scenario in two ways. The *sum-of-value* approach combines values of multiple cells to find the direction to go—it performs vectorial addition with the vector pointing from the current cell to the cell under consideration. The *target-based* approach directs the UAV to go towards the cell with the maximum value. Through logical arguments and simulation results described in [44], the latter approach can be shown to be better. Therefore, this control law involves finding the values of all cells at each time step and moving towards the cell with maximum value. If there are multiple cells with the same maximum value, then, instead of choosing randomly between them, the UAV chooses the one requiring least heading change.

We realize that value of the weight derived for the two-cell problem ($w_0 = -1/V_{\text{mission}}$), need not be optimal for the multiple-cell case. Hence the optimum value of the weight is found using an iterative sampling based optimizer (ISIS). This nongradient, population-based optimizer was developed by one of the authors (see [45] for example). The optimization is offline and the objective function is the actual mission cost (i.e., the maximum age observed over all cells). Basically the optimizer tries several values of weights (systematically) and runs the simulation to find the corresponding values of mission cost and then selects the weight resulting in minimum cost. The optimum weight found is in fact close to the analytical optimum—this also indicates that extending the policy as above is reasonable.

C. Testing Policy Performance

In order to quantify the performance of the target-based policy, it is first compared to certain benchmark techniques. A random action policy performs very poorly since certain regions in the target space are always left unexplored, so those comparison results are not shown here.

1) *Comparison to a Potential Field Like Approach:* The first benchmark technique is a heuristic approach similar to work by Tumer and Agogino [26] on a multi-rover exploration problem.⁴ The latter closely resembles a potential field approach. For comparison, a target space of unit length is considered, with a sensor footprint radius, $r_{\text{sensor}} = 0.025$, and a mission velocity, $V_{\text{mission}} = 0.03$. The plots of maximum age over a long time period for 50 trials are shown in Fig. 4. We observe that the target-based approach performs significantly better.

2) *Comparison to a Planning-Based Approach:* A popular planning algorithm used in literature is the *Dijkstra's shortest path algorithm* [46]. The algorithm described in [47] is modified to obtain a *longest path* algorithm. The nodes in the graph correspond to the cells in the grid and the weights of the edges are ages of the neighboring cells. The number of nodes in the graph are order $\tilde{O}(20^h)$, where h is the finite time horizon. The planning algorithm is implemented for time horizons of up

⁴In our implementation, we used a linear control policy, that performed better than training neural networks online using an evolutionary algorithm as in [26].

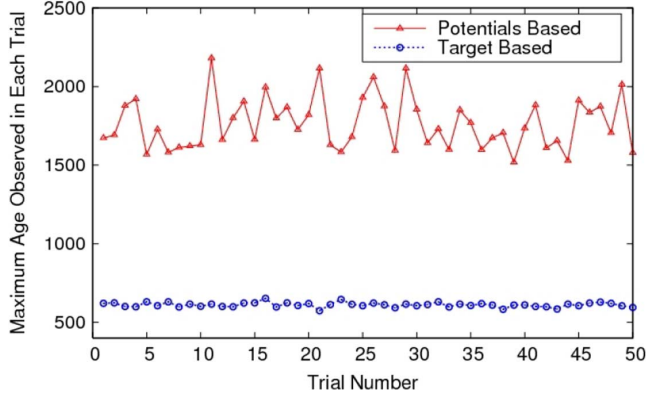


Fig. 4. Comparison of a heuristic policy, similar to a potential field method, and the target-based approach.

to 3 steps (due to computational limitations) and it performs much worse than the reactive target-based policy. This is expected, since the reactive policy tries to convert the time-extended problem into a single step problem, by incorporating a measure of time, through distance weighted by velocity. Hence it potentially looks at an infinite time horizon. In a strict planning-based approach however, we are limited to finite time horizons.

3) *Emergence of Search Pattern and Comparison to Optimum*: For certain initial conditions, for instance when the UAV starts from one corner of the target space, the control policy results in a spiral search pattern. Basically the UAV spirals in to the center and then returns to the starting location, repeating the pattern. This pattern is not optimal but it is reasonably close, with the additional advantage of being able to react to problem dynamics or failures.

We further compare the performance of the policy for arbitrary initial conditions, to a lower bound on the optimum.⁵ Consider the same target space with $r_{\text{sensor}} = 0.02$, and $V_{\text{mission}} = 0.04$. Fig. 5 shows the maximum age over all cells as a function of number of time steps (results averaged over 50 trials). It is observed that the performance of the target-based approach is pretty close to the bound on the optimum.

III. CONTROL OF MULTIPLE UAVS

A lot of schemes for coordination among multiple vehicles have been proposed in literature. In this work, the focus has been on techniques that are robust, scalable, simple in concept, and do not require sharing of plans. We propose a technique that is an extension of the reactive policy for a single UAV. This technique, the *Multi-agent Reactive Policy* (MRP), is described next.

A. Multi-Agent Reactive Policy (MRP)

A simple two-cell two-UAV problem is analyzed to understand how the existing policy can be extended to multiple UAVs. The case shown in Fig. 6 is used for this purpose.⁶ Assuming that

⁵This equals the number of cells in the domain, and is the best that any policy can do, since the UAV moves one cell-length in one time step.

⁶Certain other arrangements of the UAVs are either antisymmetric to this case, or have trivial solutions.

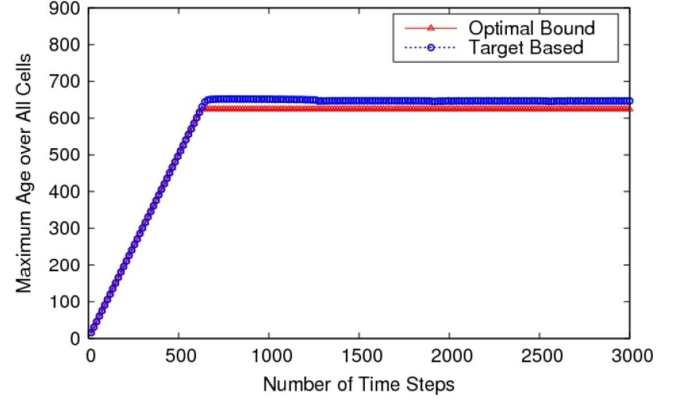


Fig. 5. Comparison of target-based approach to a bound on the optimum.

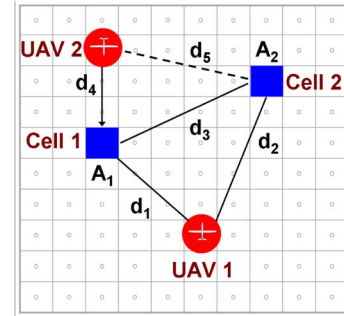


Fig. 6. Simple problem with two cells and two UAVs.

UAV 2 is going to cell 1, the control policy for UAV 1 is under question. An analysis similar to the single UAV case, results in

$$\begin{aligned}
 \text{Choose cell 1} &\Leftrightarrow A_2 + \frac{d_4}{V_{\text{mission}}} \leq A_1 + \frac{d_2 + d_3}{V_{\text{mission}}} \\
 &\Leftrightarrow \left(A_2 - \frac{d_2}{V_{\text{mission}}} + \frac{d_4}{V_{\text{mission}}} \right) \\
 &\leq \left(A_1 - \frac{d_1}{V_{\text{mission}}} + \frac{d_5}{V_{\text{mission}}} \right) + \frac{d_1 + d_3 - d_5}{V_{\text{mission}}} \quad (4)
 \end{aligned}$$

This structure motivates the policy for multiple UAVs, though it is close to optimum only for $d_1 + d_3 \sim d_5$. The value of each cell (for the i th UAV) is now given by (5), where w_1 is an additional weighting parameter for the distance of the cell to the nearest other UAV. This policy makes intuitive sense as well, since a UAV should not go to a cell that is already close to another UAV.

$$\mathcal{V}_j = \max\{A_j + w_0 \delta_{ij} + w_1 \min_{k \neq i}(\delta_{kj}), 0\}. \quad (5)$$

The control policy is defined by two weights that need to be optimized offline analogous to the single UAV case. Again, ISIS is used for optimization allowing heterogeneous control policies (i.e., different policy weights for each UAV). Ideally, the weights would need to be reoptimized when the mission specifications (e.g., the number of UAVs or size of target space) change, but the sensitivity of mission performance to such changes, using a fixed set of weights is found to be small. Also, the weights can still be found offline, and stored in a look-up table for different mission specifications.

The policy performs much better than the case of no coordination between UAVs, but before making any claims about the approach, it is compared to certain benchmarks. These include a space decomposition approach that involves optimally partitioning the target space to allocate to different UAVs for parallel surveillance, and a simplified optimal result. The reactive policy is found to exhibit an interesting emergent behavior as the number of UAVs becomes large and exhibits a character and performance similar to the space decomposition approach. The details of these results can be found in [48].

IV. INCORPORATING AIRCRAFT DYNAMICS

Kovacina *et al.* [34] claim that though devoid of terrain considerations, UAVs have to consider aircraft dynamic constraints in their control policies. Certain existing work has considered constraints imposed by vehicle dynamics [49], [50], but the problem of coupling between dynamic constraints and control policy has not been sufficiently addressed.

Previously, we have studied the effect of aircraft dynamics on the performance of the UAV. A 3-DOF aircraft dynamics simulation, similar to [51] with additional terms for thrust, was used for this purpose. It was observed that dynamic constraints, in particular the turn radius constraint, have a significant effect on the mission performance [52]. In this paper a modification to the control policy to account for dynamic constraints is described.

A. Dynamics Model With Nonholonomic Constraint

For the purpose of studying the interaction between the control policy and aircraft dynamics, the 3-DOF model is further simplified by introducing the nonholonomic constraint of constant velocity. Assuming the UAV has sufficient thrust (or can lose slight altitude while turning) to overcome drag, the system becomes a single input system, with the side force (directly related to lift coefficient, C_L , of the UAV) as the control input. The equations of motion for this system are given by (6), and the minimum radius of turn, R_{turn} , can be determined by (7).

$$\begin{aligned} \dot{x} &= V_{\text{mission}} \cos \psi \\ \dot{y} &= V_{\text{mission}} \sin \psi \\ \dot{\psi} &= \frac{F_y}{mV_{\text{mission}}} \end{aligned} \quad (6)$$

$$R_{\text{turn}} = \frac{mV_{\text{mission}}^2}{F_{y_{\text{max}}}} \quad (7)$$

These equations are equivalent to the 3-DOF system under the given assumptions. So any results we obtain using these are directly applicable to the former.

B. Minimum Length Trajectory Control

For control, the minimum length trajectories can be constructed geometrically and corresponding control inputs can be found [53]. For the case in Fig. 7, four candidate paths exist between points A and B. The shortest path is one of the four paths, and can be found analytically. Recently, Modgalya and Bhat [54] derived a feedback control algorithm for traversing these paths. But this algorithm does not cater to all the cases

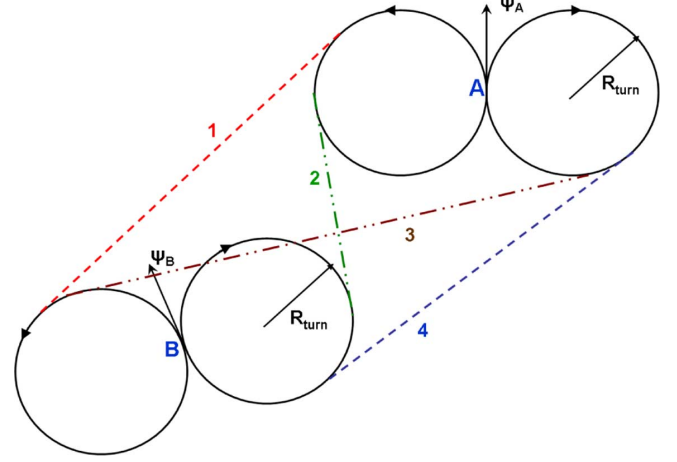


Fig. 7. Sample case showing how to find the minimum length trajectory starting from point A (heading, ψ_A), and reaching B (heading, ψ_B). There are four candidate paths, numbered 1 to 4, and the shortest path can be found by calculating the lengths for all of them.

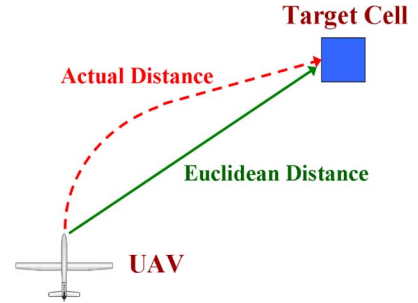


Fig. 8. Illustration of difference between the ADP and EDP policies. ADP uses the shortest (actual) path distance to a target cell, and EDP uses the Euclidean distance.

we are interested in, so we have developed our own *minimum distance controller* to traverse the shortest path trajectories.

C. Control Policy Modification

Refer to the control policies given in (3) and (5), again. The distances used in these policies so far, were Euclidean distances. But under dynamic constraints the actual distances to cells, given the positions and headings of the UAVs, are more appropriate. Hence the shortest path distance to a cell under dynamic constraints, replaces the Euclidean distance in the policy. This is illustrated in Fig. 8. The former approach is called the *Euclidean Distance Policy* (EDP), and the modified one is *Actual Distance Policy* (ADP). These are compared to see if there is some performance improvement.

For this purpose, consider UAVs with $m = 1$ kg, $V_{\text{mission}} = 5$ m/s, and control inputs of $F_{y_{\text{max}}} = 5, 8, 15$ N. These correspond to $C_{L_{\text{max}}} = 1.03, 1.18, 1.67$, and $R_{\text{turn}} = 5, 3.125, 1.67$ m respectively (for an assumed reference wing area, $S_{\text{ref}} = 0.7$ m²). We study scenarios with 1, 3, 5 and 10 UAVs on a target space, $50 \text{ m} \times 50 \text{ m}$ in dimension for a single UAV and $75 \text{ m} \times 75 \text{ m}$ in dimension for multiple UAVs, each with $r_{\text{sensor}} = 2.5$ m. MRP is used for coordination between multiple UAVs. Table I compares the maximum ages observed, averaged over 50 trials, as functions of the number of UAVs N_{UAV} for different $C_{L_{\text{max}}}$

TABLE I
SUMMARY OF RESULTS COMPARING EDP AND ADP FOR
DIFFERENT DYNAMIC CONSTRAINTS

Case	$C_{L_{max}} = 1.03$		$C_{L_{max}} = 1.18$		$C_{L_{max}} = 1.67$	
N_{UAV}	EDP	ADP	EDP	ADP	EDP	ADP
1	261.7	255.8	198.5	196.1	108.4	107.1
3	235.9	209.2	194.7	179.0	145.9	144.4
5	226.3	196.9	188.2	174.7	145.7	141.8
10	124.6	106.6	101.3	94.5	80.3	79.0

values. A graphical comparison of the results can be found in [52].

The performance of ADP with respect to EDP improves as the dynamics become more constrained, and as the number of UAVs increase. However, the improvement with N_{UAV} tends to saturate. This is because another interesting *emergent* behavior is observed. Under dynamic constraints, the UAVs tend to leave unexplored gaps in the target space. However, when other UAVs are present, they are able to fill these gaps and hence reduce the degradation in performance.

V. INCORPORATING ENDURANCE CONSTRAINTS

A solution to the *persistence* problem needs to account for the finite endurance of the UAVs, such that the UAVs can return to their base-stations periodically for recharging/refueling. Mei *et al.* [35] survey techniques for energy and time constrained problems, while addressing a deployment problem under such constraints. Sujit and Ghose [8] study a surveillance problem under endurance constraints. But, both these studies do not address the issue of refueling for a long duration mission. Albers and Kursawe [55] have studied a problem with a *piecemeal constraint* (i.e., the UAV has to visit the base-station frequently). However, they look at a single robot problem only making no claims about the efficiency of their approach.

There are two aspects to the refueling problem: one is to decide when each UAV should land, and the other is to modify the control policy to account for the endurance constraints. In general, these problems are coupled, but we do not expect the coupling to be strong. Hence, in this work each problem is analyzed separately.

A. Deriving the Health Monitoring Policy

First a *health monitoring* policy is devised to decide when each UAV should land for refueling.⁷ It is assumed that information about the refuel times of UAVs is common knowledge. Based on the remaining fuel (also referred to as *health*) of all UAVs, each UAV decides whether or not it wants to land. For the purpose of deriving the policy, the following assumptions are made: 1) the UAVs follow a more or less optimal policy, recording a maximum age of N_{cells} (number of grid cells) without endurance constraints (if they travel one cell-

length in a single time step); 2) extra fuel consumed in maneuvers is ignored; and 3) the refuel time for a UAV is T_{rf} including the set-up time, irrespective of the position of the UAV in target space.⁸ Note that these are not limitations to the application of our proposed policy.

The case of a single UAV is not very interesting, since often it does not make sense for the UAV to return to the base-station before running out of fuel—primarily because the setup time including time to return to base might be prohibitive. However, the problem with multiple UAVs has an additional degree-of-freedom—choosing the UAV that should land to refuel, allowing the refuel cycle to be split over UAVs instead of time. In this case, it is desirable to minimize the maximum number of UAVs being refueled at any point of time, $N_{UAV_{rf}}$. One way to look at this is: suppose all UAVs have infinite endurance, then all UAVs should continue to survey the space. Now if the UAVs have to land for some reason periodically (say for refueling), then they should still try to have as many UAVs surveying the space as possible while meeting the constraint of periodic landings. Actually, the optimum policy would depend on the relative values of the time to explore the domain, the refueling time, and the endurance of the vehicles. But for reasonable values of refueling time and endurance (that make this problem interesting in the first place), it can be shown that the objective of minimizing $N_{UAV_{rf}}$ is aligned with our overall mission objective (for details, refer to [48]). Moreover, practical considerations such as avoiding congestion close to the base-station, and not being able to simultaneously refuel all UAVs add support to this claim.

1) *Motivation for a Dynamic Policy*: With the objective of minimizing $N_{UAV_{rf}}$, let us consider a case of homogeneous UAVs where $N_{UAV_{rf}}^* = 1$ is the minimum possible. Further assume that the time to refuel each UAV, $T_{rf} < (N_{cells}/N_{UAV} - 1)$ and the world is deterministic.⁹ In this case, one way to ensure that $N_{UAV_{rf}} = 1$, is to change the phases of the refuel cycles of UAVs such that there is no overlap between them. So a pre-determined health monitoring policy is adequate. Also note that there are multiple solutions to this problem, since the mission performance is the same as long as there are no overlaps between the cycles. However, if there are heterogeneous UAVs with different refuel times and endurances, a predetermined scheme will not work. Apart from heterogeneity, certain stochastic elements such as inaccurate estimates of endurance, E , and T_{rf} , also require the health monitoring policy to be dynamic and adapt to changes in the environment. This holds for cases with $N_{UAV_{rf}}^* > 1$ as well. Simulation results presented in Section V.C substantiate this claim.

2) *Posing Decision Making Problem as an Optimization Problem*: In this section, the decision making problem of when to land is posed as an optimization problem. This can be solved by each UAV periodically, given the health and refuel times of other UAVs. The refueling time T_{rf} for each UAV, can be considered an interval in 1-D (see Fig. 9). The objective is to minimize the overlap between these intervals. For $N_{UAV_{rf}}^* = 1$,

⁸The setup time refers to the fixed time required to refuel, including time to return to base.

⁹Determinism here refers to the fact that the remaining endurance and refuel times of all UAVs are accurately known.

⁷This policy operates at a higher level than the control policy, that chooses the waypoints for the UAVs.

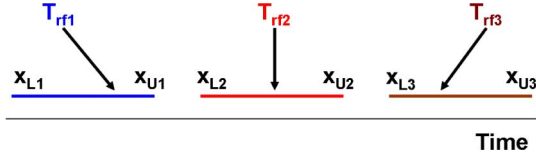


Fig. 9. Representation of the refueling cycles of UAVs as intervals in 1-D. The colored lines indicate the refueling intervals, with x_{L_i} as the start time and x_{U_i} as the end time of refueling cycle for i th UAV.

a no overlap solution exists, but for $N_{\text{UAV}_{\text{rf}}} > 1$, *minimum overlap* is not clearly defined—this is discussed later in this section.

In Fig. 9, x_{L_i} and x_{U_i} are basically the start and end times of refuel cycle for the i th UAV. Therefore, $T_{\text{rf}_i} = x_{U_i} - x_{L_i}$, where T_{rf_i} is the refuel time of the i th UAV. If we formulate an optimization problem with the objective of minimizing $N_{\text{UAV}_{\text{rf}}}$, the resulting problem is discrete and difficult to solve online. Hence better formulations need to be considered. We propose to use x_{L_i} and x_{U_i} as optimization variables. The length of overlap between the i th and j th intervals is given by: $O_{ij} = (x_{U_i} - x_{L_i}) + (x_{U_j} - x_{L_j}) - (x_{U_j} - x_{L_i})$. Consider homogeneous UAVs, i.e., equal length intervals, to begin with. This is however not a strict requirement as we will see. Our aim is to minimize the maximum value of O_{ij} over all pairs of intervals. To examine this, consider again the case where $N_{\text{UAV}_{\text{rf}}} = 1$ is possible. The above formulation results in a no overlap solution in this case, that is the same as minimizing $N_{\text{UAV}_{\text{rf}}}$. Moreover, for the case where an overlap is inevitable, it tries to minimize the length of the overlap such that the minimum time is spent with multiple UAVs on the ground. This is an intuitively better way of solving this problem than directly dealing with the objective of minimizing $N_{\text{UAV}_{\text{rf}}}$. For $N_{\text{UAV}_{\text{rf}}} \geq 2$, the case where two or more UAVs need to land for refueling at the same time, this objective still tries to avoid multiple overlaps if possible.¹⁰

In the above formulation, the i th interval should start before the j th interval, that is $x_{L_i} \leq x_{L_j}$. This is ensured by sorting the intervals according to their bounds x_{B_i} that specify the maximum time the i th UAV can fly before it has to refuel. This is the same as the remaining endurance of the UAV. It is also required that the i th interval end before the j th interval, i.e., $x_{U_i} \leq x_{U_j}$. For the case of homogeneous UAVs, the above ordering ensures this. In case the intervals are not of the same length, but approximately equal, we should still get a near-optimal solution.

The above problem can be posed as an LP, and solved using a nonlinear gradient-based optimizer, SNOPT [56]. The variable vector is: $[x_{L_1}, x_{U_1}, \dots, x_{L_{N_{\text{UAV}}}}, x_{U_{N_{\text{UAV}}}}, t]^T$, where t is the maximum overlap distance ($t = \max_{i,j}(O_{ij})$). The problem is posed mathematically in

$$\begin{aligned} \text{Minimize} \quad & t - \epsilon x_{L_1} \\ \text{subject to} \quad & -t \leq 0 \\ & x_{U_i} - x_{L_j} - t \leq 0 \\ & -x_{L_i} \leq 0 \end{aligned}$$

$$\begin{aligned} x_{L_i} - x_{B_i} &\leq 0 \\ x_{U_i} - x_{L_i} - T_{\text{rf}_i} &= 0 \\ -x_{L_i} - t + T_{\text{rf}_{\text{rem}}} &\leq 0 \end{aligned} \quad (8)$$

The second term in the objective ensures that out of the multiple optima in this problem, the solution where the time to land for the first UAV is maximum is chosen. This is desirable, since the UAVs keep surveying the space as long as it does not affect the mission performance. The weight ϵ can be made arbitrarily small for the solution of the LP to get arbitrarily close to the optimum. The first constraint ensures that only positive overlap values are considered (thus the intervals are not pushed away from each other if there is no overlap). There are $((N_{\text{UAV}}^2 - N_{\text{UAV}})/2)$ constraints of second type that enforce each overlap length to be less than t . There are N_{UAV} constraints of third and fourth kind, that are basically side bounds. The fifth set of N_{UAV} equality constraints enforce that the interval lengths equal refueling times. The last constraint accounts for the fact that there might be other UAVs already being refueled when one decides to land (for $N_{\text{UAV}_{\text{rf}}} > 1$). Denoting the time remaining to refuel the UAV on ground (maximum value in case of multiple UAVs) as $T_{\text{rf}_{\text{rem}}}$, we consider the overlap of refueling cycle of the first UAV to one on the ground as well.

This is now evaluated in simulation. The UAVs can be in one of three states: *active*, *returning to base*, or *refueling*. This information is also required by the LP solver in order to decide when and which UAV should land to refuel. The state information is currently shared among UAVs for high-level control as well, but this is not really required if the UAVs know positions of other UAVs. The LP is solved using a centralized optimizer for purposes of simulation, but for an actual implementation, the LP problem can be solved periodically by each UAV, in a distributed manner. If the information about the healths of the UAVs and the positions of the UAVs is approximately correct, they would reach approximately the same solutions. The UAVs solve the LP to find out their landing times, and land either according to these times, or if they hit *fuel reserve* before such time. It is assumed that the UAVs have sufficient fuel in reserve to be able to return to the refueling stations. As we will see in the next section, this requirement can be relaxed with our modified control policy.

B. Control Policy Modification

We now return to the second part of the problem that was assumed to be decoupled from the health monitoring policy. This section describes how the high-level control policy can be modified to obtain desirable behavior under endurance constraints. This can be done in two ways: 1) for a single UAV, trying to choose a target cell based on the remaining fuel; and 2) for multiple UAVs, trying to enhance coordination.

1) *Single UAV Policy*: We first look at the policy for a single UAV, and start with a two-cell sample problem again, as shown in Fig. 10. An analysis analogous to Section II-A shows that the policy need not be changed to account for the limited endurance. This however, is not true for more than two cells, but it is not clear how likely such pathological cases are. Also, it is not clear how to incorporate a planning-based approach here to improve upon the performance—a naive implementation (such as that

¹⁰A preliminary analysis suggests that the case of multiple interval overlap would tend to give the maximum value of O_{ij} for most cases. But to enforce that more rigorously, we need to add higher order overlaps (i.e., three UAV overlaps, four UAV overlaps, etc.). We refrain from doing that in this work, since the intent is to demonstrate the basic approach, but the extension should be very simple.

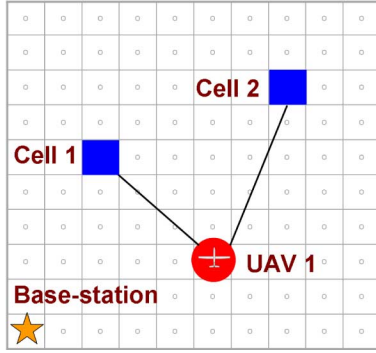


Fig. 10. Simple two-cell problem used to study need for policy modification for single UAV under endurance constraints. The UAV needs to persistently survey cell 1 and cell 2, while periodically visiting the base-station as well.

described in Section II-C.2), might actually worsen performance due to finite horizon. In this paper, we continue with our policy for a single UAV, for demonstration purposes, but this is a topic for future research.

2) *Multiple UAV Policy*: For the case of multiple UAVs, it is desirable to have UAVs low on fuel stay close to their base-stations. One way to achieve this would be an adaptive division of the target space amongst the UAVs. However, a simple modification to the control policy does the trick.

Define *reachable* cells for each UAV, as the set of cells that the UAV can visit and return to the base-station from, given its remaining endurance. In the modified approach, the UAV calculates the values of cells as before, using (5), but instead of doing so for all the cells in the target space, it considers only the reachable cells. This has several advantages.

- This ensures that the UAV will be able to reach the base-station without running out of fuel. The other alternative without this policy modification is to have enough reserve fuel to go to the base-station from anywhere in the target space. This can be a highly restrictive assumption depending on the target space dimensions and can considerably affect the performance.
- This automatically makes UAVs low on fuel stay close to the base-station. Moreover, by virtue of the MRP, the UAVs that have sufficient fuel, go to regions farther from the base-stations. This is another very interesting *emergent* behavior observed in the system.

There is a minor point to note here. Ideally, for calculation of the reachable set the shortest path distances from the cells to the base-station should be used. However, there are a few reasons not to do that: 1) in a stochastic environment, the accuracy of these distances reduces with larger time horizons; 2) the online computation required increases without significant improvement in performance; and 3) instead of using the shortest path, a weighted measure of the Euclidean distance is used, that can be tuned to get desirable performance.¹¹

This control policy modification and the health monitoring policy are incorporated in our original method, resulting in the

¹¹For a large weight, the UAV tends to explore more, with the downside of having to carry more reserve fuel. The opposite is true for low weight values, and it is not obvious where the optimum lies. In this work, the weight has been hand-tuned for reasonable performance, leaving the optimization for future work.

LP-based approach. This approach is then compared to two benchmark techniques to quantify our claims.

C. Comparison to a Pre-Determined Benchmark Policy

The first benchmark technique is a pre-determined version (does not react to changes in environment) of the health monitoring policy outlined above, and is based on the same principle of separating out the refueling cycles in time. This comparison shows the advantage of: 1) using a reactive policy (that solves the LP repeatedly) under uncertainties and 2) the control policy modification. The effect of the latter tends to be greater, so the comparison results give a good indication of the utility of the control policy modification. In particular, the refuel intervals of the UAVs (assumed homogeneous) are spread in time for minimum overlap. This is the same as our health monitoring policy (without control policy modification) in a deterministic environment. In the comparisons that follow, an uncertainty is added to the fuel consumption rate to simulate differences in actual endurance—a deviation of up to 20% from the nominal value. The UAVs are assumed homogeneous and have constant refuel time, irrespective of the amount of fuel they land with. Both policies are unaware of the real fuel consumption rate, but know the amount of fuel remaining. The LP-based approach can use this information since it is reactive in nature, but the pre-determined policy has to carry extra reserve fuel for two (practical) reasons: 1) to get back to the base-station from any point in the target space and 2) to account for the uncertainty in the endurance of the UAV.

The following parameters are varied and the mission cost is observed as a function of these parameters: 1) target space size; 2) number of UAVs; 3) target space shape; and 4) refuel time. The following baseline values are used in all these results (except for the parameter being varied): $V_{\text{mission}} = 5$ m/s, $r_{\text{sensor}} = 5$ m, $E = 100$ s, $T_{\text{rf}} = 25$ s, $N_{\text{UAV}} = 5$, and target space size = $75 \text{ m} \times 75 \text{ m}$. Fig. 11 shows the results of comparison in terms of the mission cost for various scenarios. Fig. 11(a) shows the comparison against the size of the target space—a square shaped target space is assumed and the length (same in each dimension) is varied. For studying the performance dependence on shape in Fig. 11(c), the length in X-dimension is fixed at 25 m, and the length in Y-dimension is changed.

The LP-based policy works much better than the pre-determined policy. The improvement increases in larger target spaces, where the time to return to base-station is a significant fraction of the endurance. The same trend is observed when we change the shape of the target space, making the farthest corner of the space more distant from the base-station. With increase in refueling time, the health monitoring becomes more important, so the performance improvement over the pre-determined benchmark policy increases again. As the number of UAVs increase however, the performance improvement reduces. There are two reasons for this: 1) the value of the mission cost itself is reducing, so the same deviation means a higher percentage improvement and 2) as the space gets more congested, the effect of the particular health monitoring policy reduces, by virtue of the underlying reactive policy.¹²

¹²This is the same kind of effect that was observed with the modified policy for dynamic constraints.

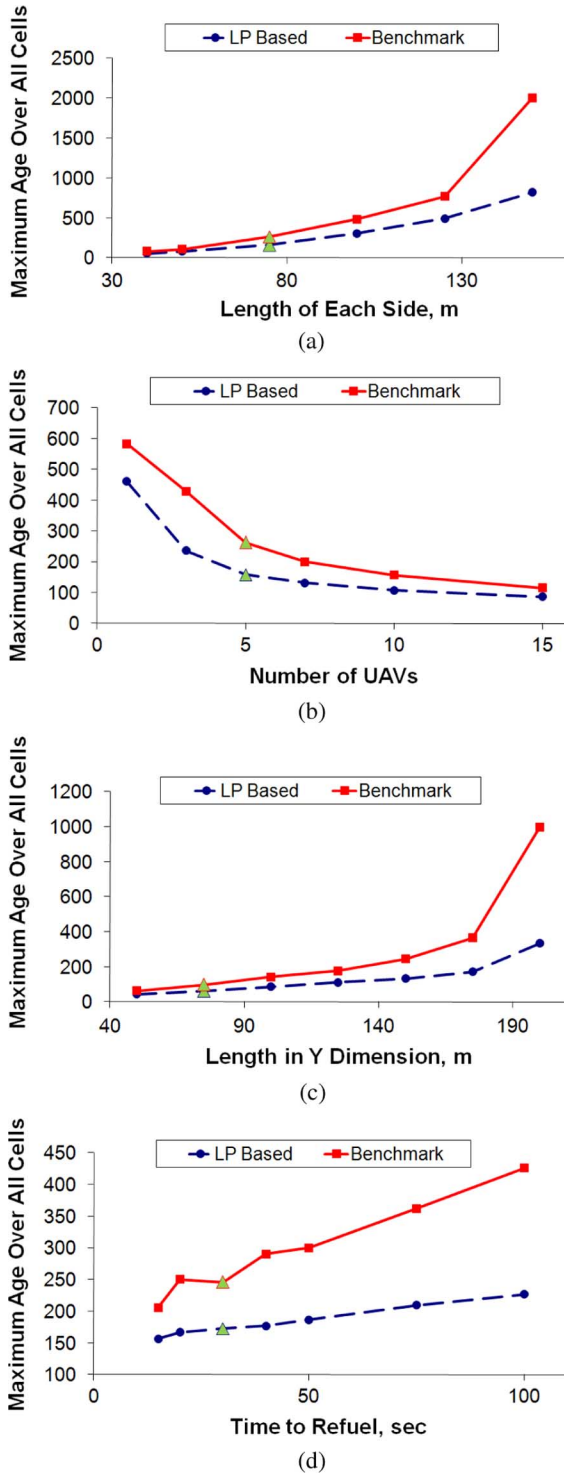


Fig. 11. Comparison of LP-based health monitoring policy to the pre-determined benchmark policy as a function of (a) length of each dimension of target space, (b) number of UAVs, (c) shape of target space, and (d) refueling time. The baseline values of the parameters being varied are shown as green triangles on both curves. (a) Size, (b) N_{UAV} , (c) Shape, (d) T_{rf} .

D. Comparison to a Reactive Benchmark Policy

The other benchmark policy used for comparison is reactive in nature. This results in reducing the reserve fuel carried by the UAVs, since it need not compensate for the uncertainty in endurance any more. The benchmark policy is conceptually

simple—it requires half the number of UAVs assigned to the mission, to survey the space at a time. The other half is either on standby, getting refueled, or returning to the base-station—this avoids the need to coordinate landing times of UAVs. The reason for not making all UAVs airborne simultaneously is to avoid all of them landing around the same time, leaving the target space unsurveyed for a long time. So basically when the UAVs need to land, the other replace them. In the implementation, a UAV on standby takes off whenever an active UAV decides to land. Note that for scenarios where it is not possible to have half the UAVs surveying the space all the time, this policy tries to put as many UAVs in air as possible. This reactive benchmark policy has also been used in the hardware results presented in Section VI-E, primarily because of its ease of implementation. In contrast to the previous set of results, comparison with this reactive technique primarily shows the utility of solving the LP to make the maximum possible number of UAVs airborne. Similar plots are generated assuming the following baseline parameters: $V_{\text{mission}} = 5$ m/s, $r_{\text{sensor}} = 5$ m, $E = 100$ s, $T_{\text{rf}} = 25$ s, and target space size = $75 \text{ m} \times 75 \text{ m}$, and $N_{\text{UAV}} = 6$. Fig. 12 shows the results of comparison. Again the LP-based approach performs better for larger space size (except for very large target space when the endurance becomes a limiting factor for any policy) and skewed domains. As T_{rf} increases, it records significant advantage over the benchmark policy. The improvement reduces slightly for congested spaces, but still remains significant. Thus we can safely conclude that the LP-based approach is promising and it would be interesting to test it on hardware in the future.

VI. FLIGHT TESTBED DEMONSTRATION

The high-level mission control architecture has also been integrated with Boeing's flight testbed to evaluate the performance of our control policies in a realistic environment.

A. Testbed Overview

Boeing Research and Technology has been developing a facility, the Vehicle Swarm Technology Laboratory, to provide an environment for testing a variety of vehicles in an indoor, controlled, safe environment [57], [58]. This type of facility provides a large number of flight test hours and the payload to fly is significantly reduced through off-board processing. It also offers a tremendous advantage in scalability and risk mitigation. Primary components of the VSTL (refer Fig. 13) include a position reference system, the vehicles and associated ground computers, and operator interface software. These elements are connected using two network busses, one carrying the position information and the other, commands and other messages. The architecture is very modular supporting rapid integration of new elements and changes to existing ones. A simulation is also available that connects directly into the two network busses and emulates the vehicles dynamics and the position reference system.

The position reference system consists of a motion capture system that emits coordinated pulses of light, that is reflected from markers placed on the vehicles. This system allows for modular addition and removal of vehicles, short calibration time, and sub-millimeter and sub-degree accuracy. Data at

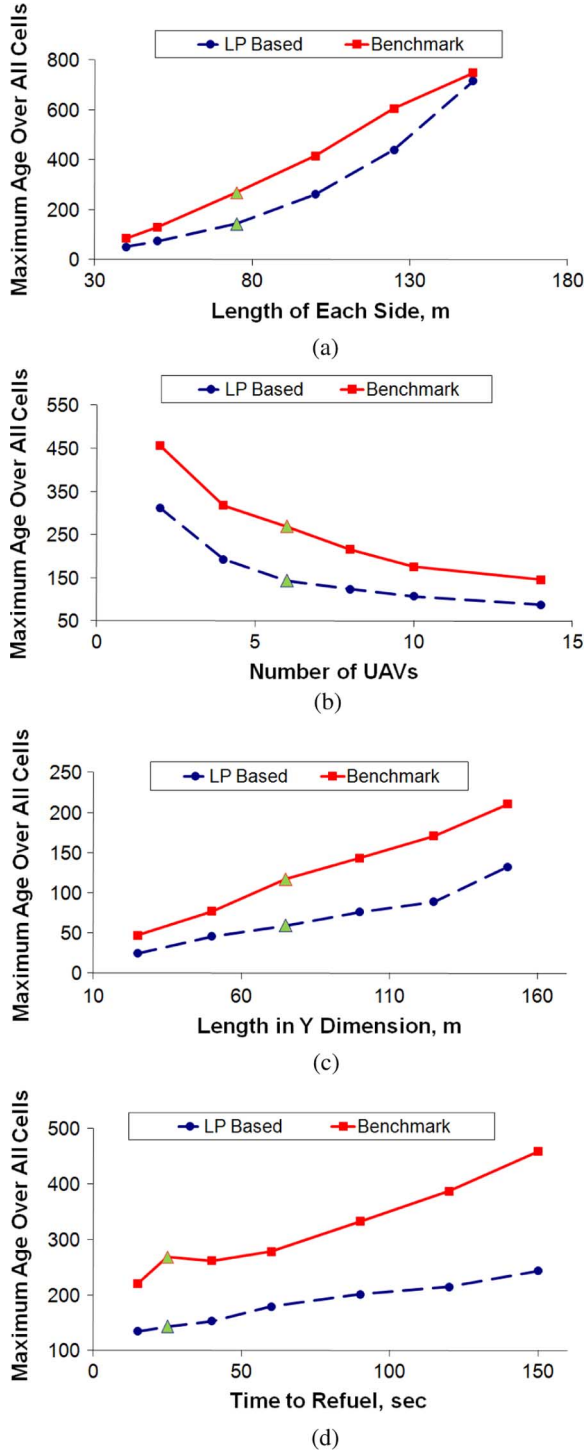


Fig. 12. Comparison of LP-based health monitoring policy to the reactive benchmark policy as a function of: (a) length of each dimension of target space; (b) number of UAVs; (c) shape of target space; and (d) refueling time. The baseline values of the parameters being varied are shown as green triangles on both curves. (a) Size. (b) N_{UAV} . (c) Shape. (d) T_{ref} .

frame rates of a 100 Hz with latency less than 40 ms are obtained for an arbitrary number of vehicles within a specified volume.

The flight vehicle used in this study is a modified remotely controlled quad-rotor helicopter. While commercially available, the on-board electronics are replaced with custom electronics

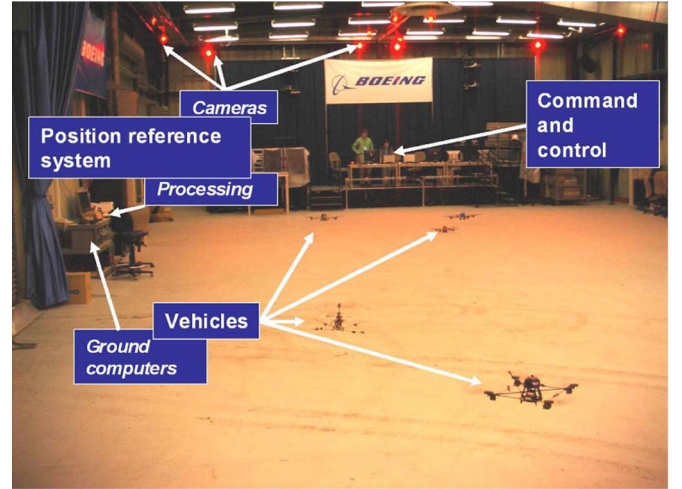


Fig. 13. Pictorial view of Boeing's Vehicle Swarm Technology Laboratory.

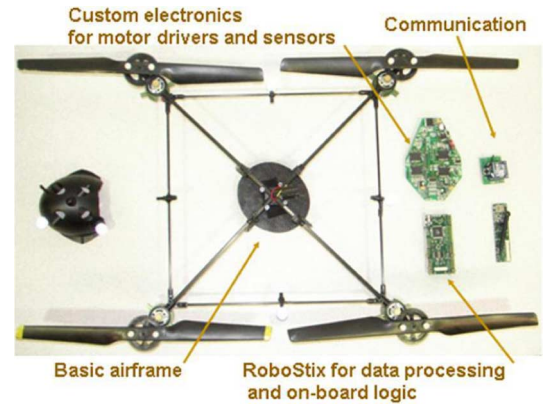


Fig. 14. Quad-rotor vehicle components.

for enhanced communication and functionality. Fig. 14 shows an image of the primary vehicle components [57]. The flight vehicle electronics are paired with a ground computer where the outer loop control, guidance, and mission management functions are executed. The performance of the control loops enables the precision coordinated flight of multiple vehicles [58]. A number of automated safety and health based behaviors have also been implemented to support simple, reliable, safe access to flight testing. Several command and control applications are used to provide an interface between the operator and the vehicles, and allow a single operator to control a large number of vehicles simultaneously [57]. To perform the missions described in this paper, an application supporting higher level mission management functionality was used, described in detail in [59].

B. Flight Test Results

We now present certain flight test results, that were obtained using our control policies on the VSTL testbed. The target space dimensions for these experiments, vary from 4 to 6 m in width, and from 4 to 10 m in length. The velocity and turn radii of UAVs are fixed at $V_{mission} = 0.2$ m/s and $R_{turn} = 0.5$ m respectively. The UAVs fly at a constant altitude of 1.5 m and

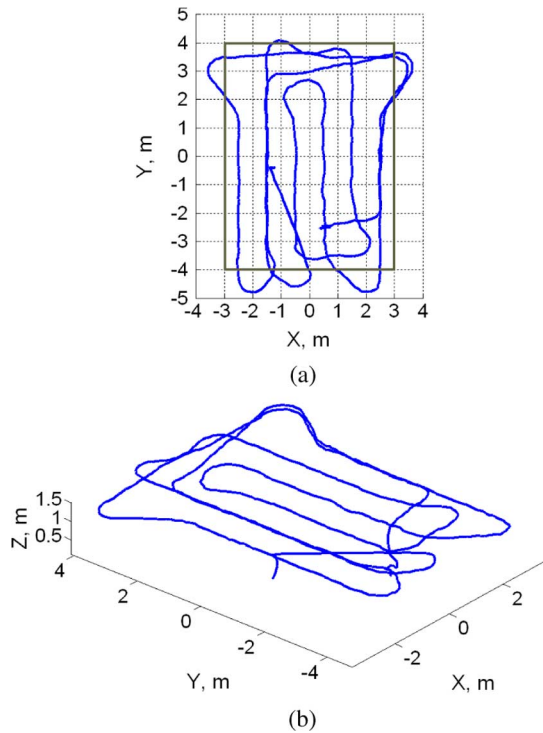


Fig. 15. Trajectory of a single UAV surveying a target space bounded by $(-3, 4, 3, -4)$. The UAV shows a roughly lawnmower search pattern. (a) 2-D trajectory, (b) 3-D trajectory.

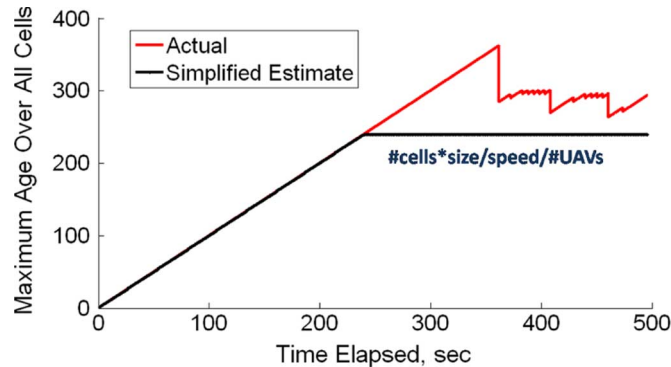


Fig. 16. Performance of a single UAV surveying the $(-3, 4, 3, -4)$ target space. The initial peaks are due to activation, takeoff, and task allocation procedures.

have a circular sensor footprint with $r_{\text{sensor}} = 1$ m. The size of the cells in the grid (used for mapping) corresponds to the sensor footprint, as earlier. The control policy weights for the MRP are found offline, as described previously.¹³ However, unlike the simulation results presented so far, the UAVs do not share maps—resulting in completely decentralized surveillance.

¹³Ideally the weights would change with the number of vehicles involved in surveillance. So in a real-world scenario where the number of the UAVs can change drastically, a simple look-up table could be employed for reading weights. However, this is not a matter of great concern, as observed through certain sensitivity studies.

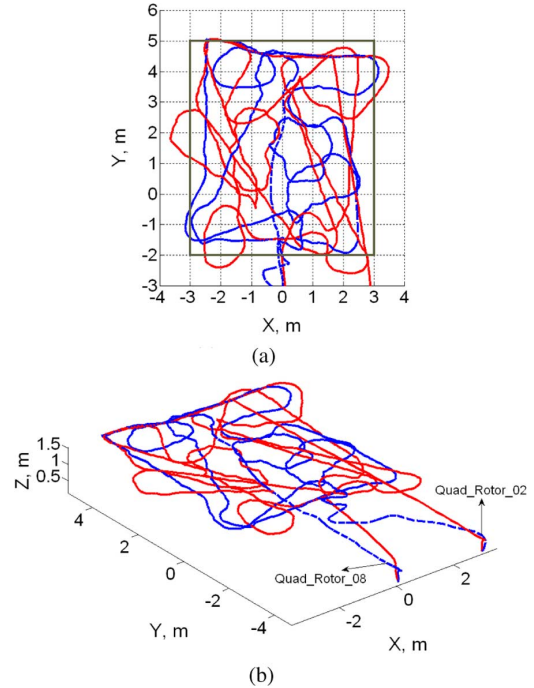


Fig. 17. Trajectories of two UAVs surveying a target space bounded by $(-3, 5, 3, -2)$. (a) 2-D trajectory. (b) 3-D trajectory.

¹⁴This has some repercussions on the performance when a UAV is replaced by a newly deployed UAV, as discussed later.

C. Results: Surveillance Using Single UAV

First, test results using one UAV are presented. Fig. 15 shows the trajectory followed by the UAV (in 2-D and 3-D) on a $6\text{ m} \times 8\text{ m}$ target space. The target space boundaries are $(-3, 4, 3, -4)$ in (west, north, east, south) notation, as shown by the gray rectangle in the 2-D plot. It can be seen that the UAV is able to effectively cover the space, respects the turn radius constraint, and follows a standard pattern (roughly lawnmower).

Fig. 16 and shows the performance of the UAV with elapsed time on the X-axis and the maximum age over all cells on the Y-axis. For this scenario, the optimum “estimate” is approximately 240.¹⁵ It is observed that while the UAV is surveying the domain, the performance stays close to this estimate. A peak is observed in the beginning because the UAV takes time to takeoff and be issued the search command. Also, after the UAV returns to base (at the end of the experiment), the maximum age starts increasing again.

D. Results: Multiple UAV Coordination

This section looks at the coordination between multiple vehicles in the flight testbed. We are able to demonstrate

¹⁴The only centralized component is the mission level architecture that allocates tasks to each UAV (tasks include activation and take off, land, search/survey, etc.). The UAVs are able to observe positions of other UAVs as well.

¹⁵Since the UAV does not cover the entire length of a cell in one time step, the bound on the optimum can not be found trivially anymore. So, in this and the following cases, we will define the estimate of optimum performance as: (number of cells)*(length of each cell)/(velocity)/(number of UAVs). This estimate is pretty close assuming no dynamic constraints, perfect coordination, and instantaneous replacement of UAVs by others (when required).

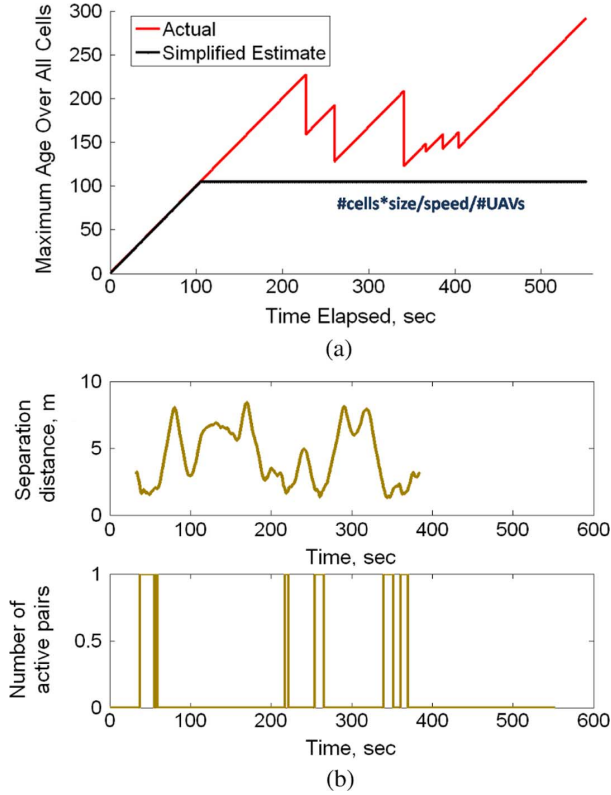


Fig. 18. Performance of two UAVs surveying the $(-3, 5, 3, -2)$ target space. Plot (a) shows the maximum age over all cells as a function of time. Plot (b) shows the separation distance between the UAVs along with the time for which collision avoidance is active. We observe that the UAVs are able to stay clear of each other most of the time.

efficient surveillance while avoiding collisions and effectively maintaining large distances between UAVs.

1) *Surveillance Using Two UAVs*: Two vehicles are commanded to survey a $6 \text{ m} \times 7 \text{ m}$ target space (bounded by $(-3, 5, 3, -2)$), in order to demonstrate the coordination aspect of the reactive policy. Fig. 17 shows the trajectories of the UAVs. The performance of the UAVs is plotted in Fig. 18(a), with the estimate of optimum being 105. The performance is close to this value when both the UAVs are surveying the domain, but when one UAV lands, the maximum age starts increasing. Though not evident from the trajectory plots, but during the experiment it was observed that the UAVs effectively spread out in space and cover different areas of the target space most of the time. Fig. 18(b) illustrates this by showing the separation distances between the UAVs as a function of time and the time periods for which the UAVs were within 2 m of each other.¹⁶ We see that the UAVs maintain a considerable separation most of the time. Also note that the UAVs are deployed at a location separated from the target space, to evaluate the situation where the base station does not lie within the target space.

2) *Surveillance Using Four UAVs*: To test the policy under congestion, the number of UAVs is increased to four—the target space dimensions are $6 \text{ m} \times 10 \text{ m}$ (bounds are $(-3, 7, 3, -3)$).

¹⁶Note that the control policy does not explicitly perform collision avoidance, but there is an underlying collision avoidance system in VSTL that is triggered when the separation distance between UAVs becomes less than 2 m.

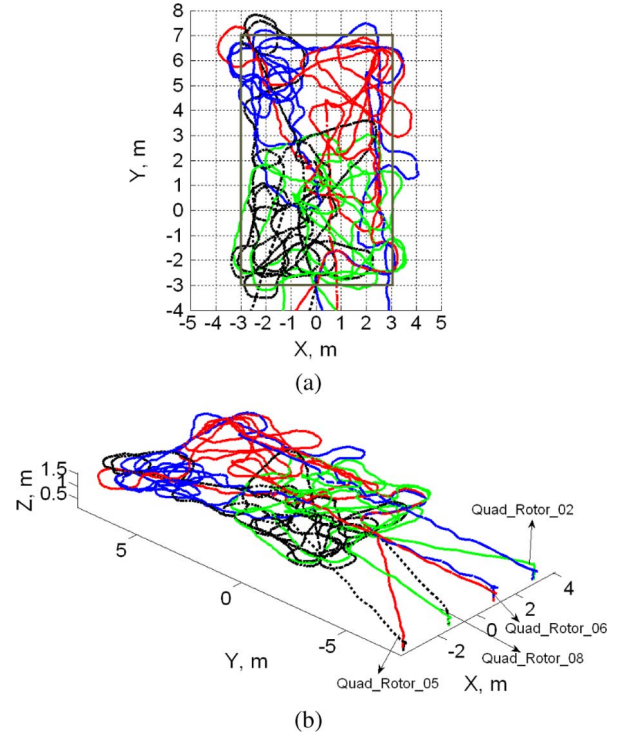


Fig. 19. Trajectories of four UAVs surveying a target space bounded by $(-3, 7, 3, -3)$. The four UAVs tend to survey different regions in the space. (a) 2-D trajectory, (b) 3-D trajectory.

The UAVs are deployed outside the target space, and the trajectories followed by the UAVs are shown in Fig. 19.

The four vehicles tend to spread to different corners in space and though there is mixing, they tend to search different locations. This can be verified by looking at the separation distance between each pair of UAVs, as shown in Fig. 20. The performance of the UAVs is also shown, with 75 as the estimate of optimum.

E. Results: Persistent Surveillance Under Endurance Constraints

This section presents results demonstrating “persistent” surveillance under endurance constraints. The policy used is the reactive benchmark policy introduced in Section V-D, except that the number of UAVs surveying the space may not be half the total number of UAVs at all times. This number is still pre-fixed however. The goal here is to demonstrate the feasibility of a health monitoring policy in hardware, and this policy is used due to its ease of implementation.

1) *Surveillance Using One UAV With Three on Standby*: First, a single UAV scenario is analyzed, with target space dimensions of $6 \text{ m} \times 6 \text{ m}$ (bounds are $(-3, 3, 3, -3)$) and 3 UAVs on standby. The mission level controller automatically decides which UAV to deploy while the others are on standby. When the active UAV fails or runs out of battery, the mission level decides to land it at its base-station while deploying another UAV to take its place. So basically, once the mission is started, there is no more human interference to carry out this persistent surveillance.

The trajectories of the UAVs (4 in total) are plotted in Fig. 21. It is interesting to observe that the UAVs end up following sim-

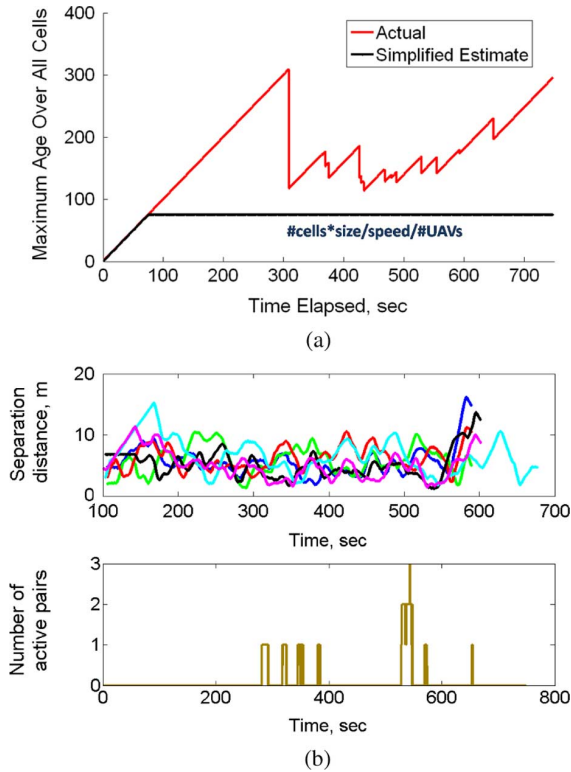


Fig. 20. Performance of four UAVs surveying the $(-3, 7, 3, -3)$ target space. (a) Plot shows the maximum age over all cells as a function of time. (b) Plot shows the separation distance between each pair and the time for which collision avoidance is active. Even in the congested space, collision avoidance is rarely triggered.

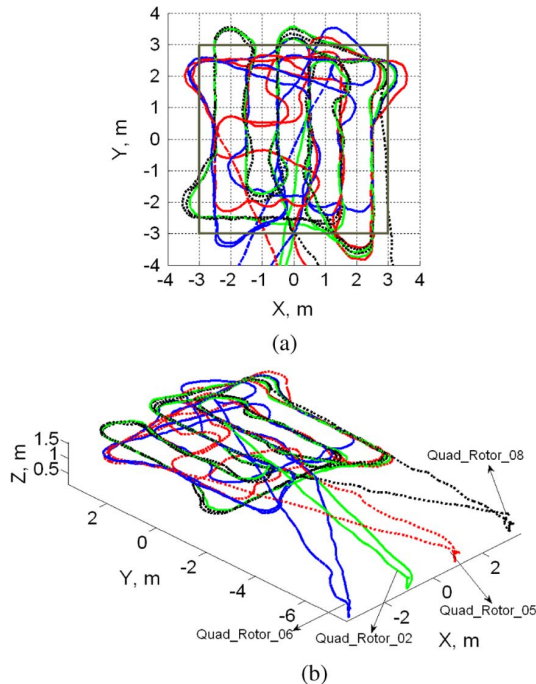


Fig. 21. Trajectories of the four UAVs in a target space bounded by $(-3, 3, 3, -3)$. A single UAV searches the domain, while three remain on standby. We observe that the UAVs end up following similar trajectories, effectively covering the domain. (a) 2-D trajectory. (b) 3-D trajectory.

ilar trajectories in spite of not being programmed to do so. The UAVs are also able to effectively demonstrate the persistence

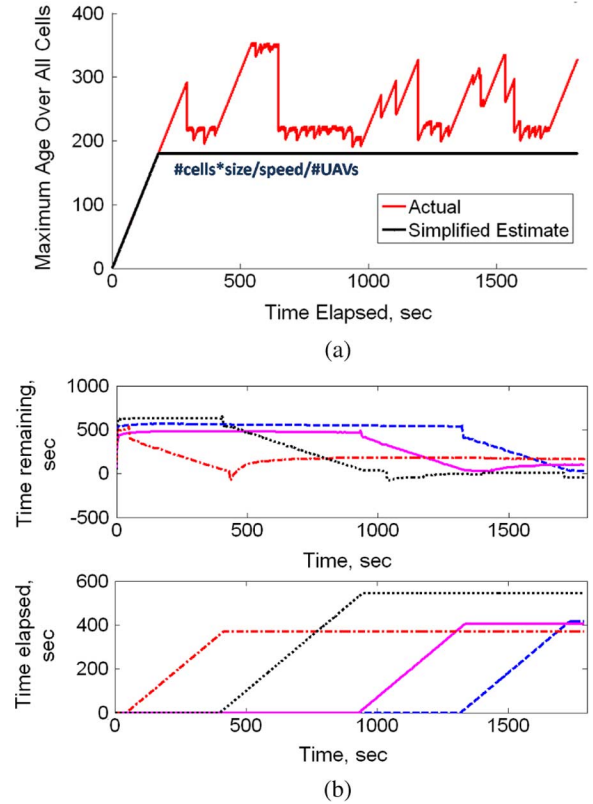


Fig. 22. Performance of single UAV surveying the $(-3, 3, 3, -3)$ target space with 3 UAVs on standby. (a) Plot shows the maximum age over all cells as a function of time. (b) Plot shows the health of the UAVs and the time they are active against experiment time (an inclined curve shows the UAVs are active). Notice the peaks in the maximum age when UAVs get replaced by others on standby.

aspect of the mission. The performance of the UAVs is shown in Fig. 22. The approximate optimum here is 180. The figure also shows the times at which the different UAVs are deployed, and the maximum age increases when a UAV gets replaced. These peaks correspond to the fact that the UAVs do not share maps of the environment, and so each UAV starts its surveillance *ab initio*.

2) *Surveillance Using Two UAVs With Three on Standby:* Next, the number of UAVs surveying the space is increased to two—i.e., two UAVs simultaneously search the domain, while three others wait to be deployed. There was a manual battery replacement on one of the UAVs during the mission—demonstrating the ability to handle human interference as well. The dimensions of the domain are 6 m \times 6 m (bounds are $(-3, 3, 3, -3)$). Fig. 23 shows the trajectories of the vehicles, demonstrating effective coverage. Fig. 24 shows the performance of the system for the complete mission, and the landing and deployment of the UAVs. The estimate of the optimum in this case is 90. There are peaks in the performance curve when the UAVs are replaced by others, though the repercussions of not sharing maps seem to be lesser.¹⁷ A final peak is observed when all the UAVs have landed. In this experiment, no UAV failed and they were able to carry on a long duration mission

¹⁷This is another interesting observation that for large groups of UAVs, map-sharing seems to be less critical. But a proper investigation of this property is work for the future.

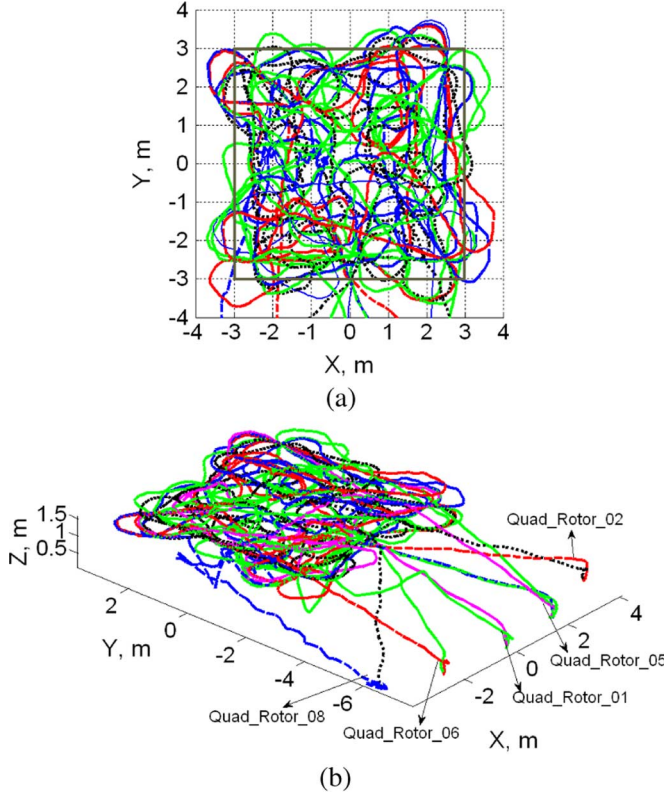


Fig. 23. Trajectories of four UAVs in a target space bounded by $(-3, 3, 3, -3)$. Two UAVs survey the domain, while three others are on standby. The system demonstrates *persistence* and good coverage. (a) 2-D trajectory. (b) 3-D trajectory.

while effectively surveying the target space. But in some other tests, we demonstrated robustness to UAV failures and achieved persistent coverage [48].

VII. VSTL SIMULATION

A simulation closely mimicking the hardware has also been developed by Boeing using Simulink in MATLAB. It accurately captures the quad-rotor dynamics of the quad-rotors and environment uncertainties. Comparison between simulation and flight tests and its use to study larger target spaces with more vehicles are detailed in [60]. Here we present a sample simulation case of 8 UAVs in a $20\text{ m} \times 20\text{ m}$ domain (see Fig. 25). The UAVs tend to spread out in space and parts of lawnmower search pattern are also apparent in the paths followed by the UAVs. This reiterates our earlier claim that the behavior of our policy approaches the space decomposition approach as the domain becomes congested.

VIII. CONCLUSION

This study has addressed the control of multiple vehicles for persistent surveillance. We develop a semi-heuristic control policy for a single UAV that is optimum for a particular case. Comparison of the policy with select benchmark techniques and a bound on the optimum shows favorable results. This policy is extended to the case of multiple UAVs using a reactive policy and comparisons are made to a bound on the optimum. An

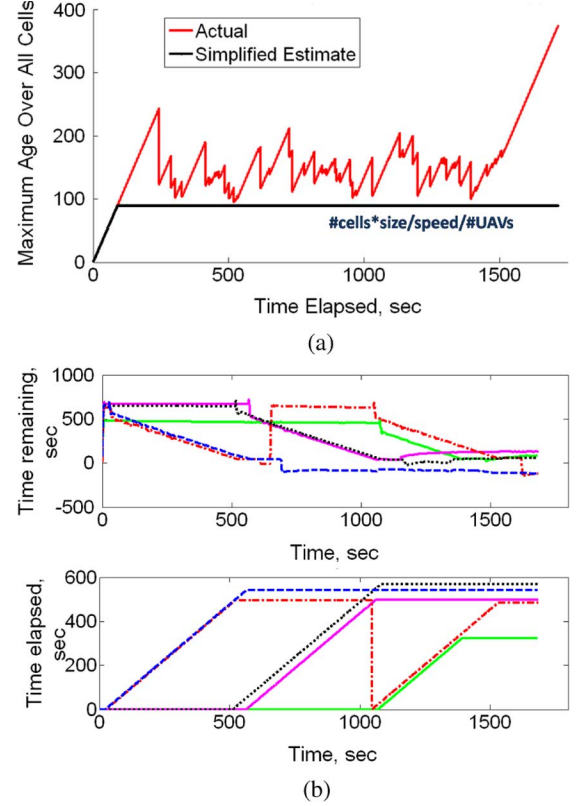


Fig. 24. Performance of two UAVs surveying the $(-3, 3, 3, -3)$ target space with 3 UAVs on standby. (a) Plot shows the maximum age over all cells as a function of time. (b) Plot shows the health of the UAVs and the time they are active against experiment time. Small peaks in maximum age are observed when new UAVs replace older ones. The final peak is observed after the UAVs have landed.

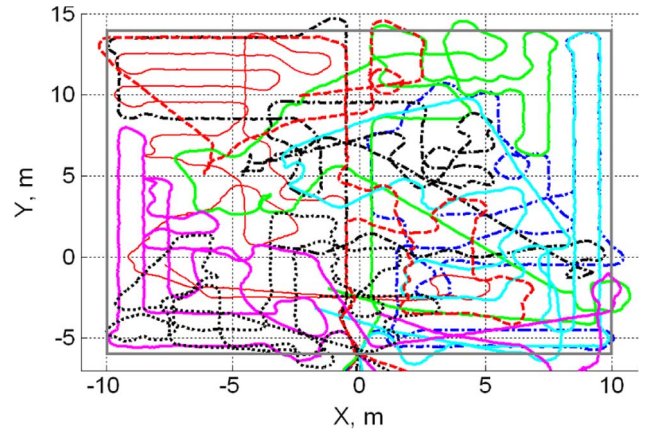


Fig. 25. Trajectories of eight UAVs surveying a target space bounded by $(-10, 14, 10, -6)$.

emergent, yet desirable and somewhat predictable behavior is observed in the policy that improves performance in congested spaces. A model to emulate dynamic constraints of UAVs is then described. A modification to the control policy is made to deal with such constraints. Interestingly, a spiral search pattern is observed for UAVs without dynamic constraints, whereas they show resemblance to lawnmower search under dynamic constraints. Next, an LP-based health monitoring policy is

developed to account for finite endurance. This can be applied in a distributed, online scenario for intelligent sequencing of landing as well as multi-vehicle coordination. It is compared to two benchmark techniques and shows clear performance improvement. Finally, the flight testbed used to evaluate these control policies is described, and experimental results are provided to validate our approach with up to 4 UAVs in several different scenarios. Although these experiments use a ground computing system, thus entailing a centralized architecture for communication, the policies themselves have been designed for distributed implementation. Through these tests we are able to demonstrate the efficiency, robustness, persistence, and reactive nature of the control policies.

Future research will include evaluation of these policies in larger target spaces with more UAVs, in the VSTL simulation environment. The LP-based approach which was not implemented on the testbed due to project time constraints, needs to be validated in the future. The simulation will also be integrated with our procedure for designing the UAVs that is presented in other work [48], [52]. For the results presented in Section VI-E, the UAVs do not share maps but know each other's position when operational (except for occasional loss of communication). It would be useful to study the performance when only limited position information exchange occurs (say due to occlusions or distance). Some preliminary investigation on usefulness of planning has been done previously [48], but a more thorough analysis, especially under dynamic and endurance constraints, is required. Using a heterogeneous group of vehicles to demonstrate persistent surveillance will also be interesting.

REFERENCES

- [1] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, "Cooperative mobile robotics: Antecedents and directions," *Auton. Robots*, vol. 4, no. 1, pp. 7–27, 1997.
- [2] L. E. Parker, "Current state of the art in distributed mobile robotics," *Distrib. Auton. Rob. Syst.*, vol. 4, pp. 3–12, Oct. 2000.
- [3] D. F. Houghton, M. D. Erickson, P. E. Rybski, S. A. Stoeter, M. Gini, and N. Papanikolopoulos, "Autonomous mobile robots and distributed exploratory missions," in *Proc. 5th Int. Symp. on Distrib. Auton. Rob. Syst.*, 2000, pp. 221–230.
- [4] H. Choset, "Coverage for robotics a survey of recent results," *Ann. Math. Artif. Intell.*, vol. 31, pp. 113–126, 2001.
- [5] W. Burgard, M. Moors, and F. Schneider, "Collaborative exploration of unknown environments with teams of mobile robots," in *Advances in Plan-Based Control of Robotic Agents*. Berlin, Germany: Springer Verlag, 2002, pp. 187–215, Lecture Notes in Computer Science.
- [6] S. B. Thrun, "Exploration and model building in mobile robot domains," in *IEEE Int. Conf. on Neural Networks*, 1993, vol. 1, pp. 175–180.
- [7] M. Flint, E. Fernandez-Gaucherand, and M. Polycarpou, "Stochastic models of a cooperative autonomous UAV search problem," *Military Oper. Res.*, vol. 8, no. 4, pp. 13–33, 2003.
- [8] P. B. Sujit and D. Ghose, "Search using multiple UAVs with flight time constraints," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 2, pp. 491–510, Apr. 2004.
- [9] J. S. Bellingham, M. Tillerson, M. Alighanbari, and J. P. How, "Cooperative path planning for multiple UAVs in dynamic and uncertain environments," in *Proc. 41st IEEE Conf. on Decision and Control*, Dec. 2002, vol. 3, pp. 2816–2822.
- [10] C. G. Cassandras and W. Li, "Sensor networks and cooperative control," in *Proc. 44th IEEE Conf. on Decision and Control*, Dec. 2005, vol. 5, pp. 4237–4238.
- [11] M. M. Polycarpou, Y. Yang, and K. M. Passino, "Cooperative control of distributed multi-agent systems," *IEEE Control Syst. Mag.*, 2001.
- [12] S. Koenig, B. Szymanski, and Y. Liu, "Efficient and inefficient ant coverage methods," *Ann. Math. Artif. Intell.*, vol. 31, no. 1–4, pp. 41–46, 2001.
- [13] M. A. Batalin and G. S. Sukhatme, "The analysis of an efficient algorithm for robot coverage and exploration based on sensor network deployment," in *Proc. IEEE Int. Conf. on Robot. Autom.*, Apr. 2005, pp. 3478–3485.
- [14] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. B. Thrun, "Collaborative multi-robot exploration," in *Proc. IEEE Int. Conf. on Robot. Autom.*, Apr. 2000, vol. 1, pp. 476–481.
- [15] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon decomposition," presented at the Int. Conf. on Field and Service Robot. (FSR'97), Canberra, Australia, 1997.
- [16] E. U. Acar and H. Choset, "Critical point sensing in unknown environments," in *Proc. IEEE Int. Conf. on Robot. Autom.*, Apr. 2000, vol. 4, pp. 3803–3810.
- [17] D. Kurabayashi, J. Ota, T. Arai, and E. Yoshida, "An algorithm of dividing a work area to multiple mobile robots," in *Proc. Int. Conf. on Intell. Robots Syst.*, Aug. 1995, vol. 2, pp. 286–291.
- [18] E. U. Acar and H. Choset, "Sensor-based coverage of unknown environments: Incremental construction of Morse decompositions," *Int. J. Robot. Res.*, vol. 21, no. 4, pp. 345–366, 2002.
- [19] Y. Gabriely and E. Rimon, "Spanning-tree based coverage of continuous areas by a mobile robot," in *Proc. IEEE Int. Conf. on Robot. Autom.*, May 2001, vol. 2, pp. 1927–1933.
- [20] S. J. Chang and B. J. Dan, "Free moving pattern's online spanning tree coverage algorithm," in *SICE-ICASE Int. Joint Conf.*, Oct. 2006, pp. 2935–2938.
- [21] T. W. Min and H. K. Yin, "A decentralized approach for cooperative sweeping by multiple mobile robots," in *Proc. IEEE/RSJ Int. Conf. on Intell. Robots Syst.*, Oct. 1998, vol. 1, pp. 380–385.
- [22] M. Berhaut, H. Huang, P. Keskinocak, S. Koenig, W. Elmaghraby, P. Griffin, and A. Kleywegt, "Robot exploration with combinatorial auctions," in *Proc. IEEE/RSJ Int. Conf. on Intell. Robots Syst.*, Oct. 2003, vol. 2, pp. 1957–1962.
- [23] D. A. Lawrence, R. E. Donahue, K. Mohseni, and R. Han, "Information energy for sensor-reactive UAV flock control," in *3rd Unmanned Unlimited Tech. Conf., Workshop and Exhibit. (AIAA-2004-6530)*, Sep. 2004.
- [24] C. A. Erignac, "An exhaustive swarming search strategy based on distributed pheromone maps," presented at the Infotech@Aerosp. Conf. Exhibit. (AIAA-2007-2822), May 2007.
- [25] A. K. Agogino and K. Tumer, "Efficient evaluation functions for multi-rover systems," in *Proc. Genetic and Evol. Computation Conf.*, Jun. 2004, pp. 1–12.
- [26] K. Tumer and A. Agogino, "Coordinating multi-rover systems: Evaluation functions for dynamic and noisy environments," in *Proc. Genetic and Evol. Computation Conf.*, Jun. 2005, pp. 591–598.
- [27] M. Flint, M. Polycarpou, and E. Fernandez-Gaucherand, "Cooperative path planning for autonomous vehicles using dynamic programming," in *Proc. 15th Triennial IFAC World Congress*, Jul. 2002, pp. 481–487.
- [28] A. Richards, J. S. Bellingham, M. Tillerson, and J. How, "Coordination and control of multiple UAVs," in *AIAA Guid., Navigation, and Control Conf. and Exhibit (AIAA-2002-4588)*, Aug. 2002.
- [29] H. V. D. Parunak, "Making swarming happen," presented at the Conf. on Swarming and Network Enabled Command, Control, Commun., Computers, Intelligence, Surveillance and Reconnaissance, 2003.
- [30] P. Gaudiano, B. Shargel, and E. Bonabeau, "Control of UAV swarms: What the bugs can teach us," presented at the 2nd AIAA Unmanned Unlimited Conf. and Workshop and Exhibit, AIAA-2003-6624, Sep. 2003.
- [31] P. F. Hokayem, D. Stipanovic, and M. W. Spong, "On persistent coverage control," in *Proc. 46th IEEE Conf. on Decision and Control*, Dec. 2007, pp. 6130–6135.
- [32] P. R. Chandler and M. Pachter, "Research issues in autonomous control of tactical UAVs," in *Proc. Amer. Control Conf.*, Jun. 1998, vol. 1, pp. 394–398.
- [33] J. Bellingham, A. Richards, and J. P. How, "Receding horizon control of autonomous aerial vehicles," in *Proc. Amer. Control Conf.*, 2002, vol. 5, pp. 3741–3746.
- [34] M. A. Kovacina, D. Palmer, G. Yang, and R. Vaidyanathan, "Multi-agent control algorithms for chemical cloud detection and mapping using unmanned air vehicles," in *IEEE/RSJ Int. Conf. on Intell. Robots Syst.*, 2002, vol. 3, pp. 2782–2788.
- [35] Y. Mei, Y. H. Lu, Y. C. Hu, and C. S. G. Lee, "Deployment of mobile robots with energy and timing constraints," *IEEE Trans. Robot.*, vol. 22, no. 3, pp. 507–522, June 2006.

- [36] D. Kurabayashi, J. Ota, T. Arai, S. Ichikawa, S. Koga, H. Asama, and I. Endo, "Cooperative sweeping by multiple mobile robots with re-locating portable obstacles," in *Proc. IEEE/RSJ Int. Conf. on Intell. Robots Syst.*, Nov. 1996, vol. 3, pp. 1472–1477.
- [37] R. G. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. B. Thrun, and H. L. S. Younes, "Coordination for multi-robot exploration and mapping," in *Proc. 17th Nat. Conf. on Artif. Intell. and 12th Conf. on Innovative Appl. of Artif. Intell.*, 2000, pp. 852–858.
- [38] N. Hazon, F. Miel, and G. A. Kaminka, "Towards robust on-line multi-robot coverage," in *Proc. IEEE Int. Conf. on Robot. Autom.*, May 2006, pp. 1710–1715.
- [39] E. W. Frew, X. Xiao, S. Spry, T. McGee, Z. Kim, J. Tisdale, R. Sengupta, and J. K. Hedrick, "Flight demonstrations of self-directed collaborative navigation of small unmanned aircraft," in *3rd AIAA Unmanned Unlimited Tech. Conf., Workshop and Exhibit (AIAA-2004-6608)*, Sept. 2004.
- [40] M. Valenti, D. Dale, J. P. How, D. P. Farias, and J. Vian, "Mission health management for 24/7 persistent surveillance operations," in *AIAA Guid., Navigation and Control Conf. Exhibit (AIAA-2007-6508)*, Aug. 2007.
- [41] B. Bethke, L. F. Bertuccelli, and J. P. How, "Experimental demonstration of adaptive MDP-based planning with model uncertainty," in *AIAA Guid., Navigation and Control Conf. and Exhibit, AIAA-2008-6322*, Aug. 2008.
- [42] N. Jodeh, M. Mears, and D. Gross, "An overview of cooperative operations in urban terrain (counter) program," in *AIAA Guid., Navigation and Control Conf. and Exhibit, AIAA-2008-6308*, Aug. 2008.
- [43] D. Enns, D. Bugajski, and S. Pratt, "Guidance and control for cooperative search," in *Proc. Amer. Control Conf.*, May 2002, vol. 3, pp. 1923–1929.
- [44] N. Nigam and I. Kroo, "Persistent surveillance using multiple unmanned air vehicles," in *Proc. IEEE Aerospace Conf.*, Mar. 2008, pp. 1–14.
- [45] D. Rajnarayan, I. Kroo, and D. Wolpert, "Probability collectives for optimization of computer simulations," in *48th AIAA/ASME/ASCE/AHS/ASC/Structures, Structural Dyn., and Mater. Conf. (AIAA-2007-1975)*, Apr. 2007.
- [46] M. Flint, E. Fernandez-Gaucherand, and M. Polycarpou, "Cooperative control for UAVs searching risky environments for targets," in *Proc. 42nd IEEE Conf. on Decision and Control*, Dec. 2003, vol. 4, pp. 3567–3572.
- [47] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. Cambridge, MA: MIT Press, 1990.
- [48] N. Nigam, "Control and design of multiple unmanned air vehicles for persistent surveillance," Ph.D. dissertation, Stanford Univ., Stanford, CA, Sep. 2009.
- [49] Y. Liu, J. B. Cruz, and A. G. Sparks, "Coordinating networked uninhabited air vehicles for persistent area denial," in *43rd IEEE Conf. on Decision and Control*, Dec. 2004, vol. 3, pp. 3351–3356.
- [50] J. Ousingasawat and M. E. Campbell, "Optimal cooperative reconnaissance using multiple vehicles," *J. Guid. Control and Dyn.*, vol. 30, no. 1, pp. 122–132, 2007.
- [51] G. Sachs, "Minimum shear wind strength required for dynamic soaring of albatrosses," *IBIS-London-British Ornithologists Union*, vol. 147, no. 1, pp. 1–10, Jan. 2005.
- [52] N. Nigam and I. Kroo, "Control and design of multiple unmanned air vehicles for a persistent surveillance task," in *12th AIAA Conf. on Multidiscip. Anal. Optimiz. (AIAA-2008-5913)*, Sept. 2008.
- [53] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *Amer. J. Math.*, vol. 79, no. 3, pp. 497–516, Jul. 1957.
- [54] M. Modgalya and S. P. Bhat, "Time-optimal feedback guidance in two dimensions under turn-rate and terminal heading constraints," presented at the Nat. Conf. on Control and Dyn. Syst., Mumbai, India, Jan. 2005.
- [55] S. Albers and K. Kursawe, "Exploring unknown environments with obstacles," in *Proc. 10th Annu. ACM-SIAM Symp. on Discrete Algorithms*, 1999, pp. 842–843.
- [56] P. Gill, W. Murray, and M. Saunders, "SNOPT: An SQP algorithm for large-scale constrained optimization," *Dep. Math., Univ. California, San Diego, Numerical Analysis Rep. 97-2*, 1997.
- [57] E. Saad, J. Vian, G. Clark, and S. R. Bieniawski, "Vehicle swarm rapid prototyping testbed," in *Proc. AIAA Infotech@Aerospace Conf. Exhibit AIAA Unmanned...Unlimited Conf. Exhibit (AIAA-2009-1824)*, 2009, pp. 1–9.
- [58] D. J. Halaas, S. R. Bieniawski, P. Pigg, and J. Vian, "Control and management of an indoor, health enabled, heterogeneous fleet," in *Proc. AIAA Infotech@Aerospace Conf. and Exhibit and AIAA Unmanned...Unlimited Conf. and Exhibit, AIAA-2009-2036*, 2009, pp. 1–19.
- [59] S. R. Bieniawski, P. E. R. Pigg, B. Bethke, J. Vian, and J. How, "Exploring health-enabled mission concepts in the vehicle swarm technology laboratory," in *Proc. AIAA Infotech@Aerospace Conf. and Exhibit and AIAA Unmanned...Unlimited Conf. and Exhibit, AIAA-2009-1918*, 2009, pp. 1–13.
- [60] N. Nigam, S. Bieniawski, I. Kroo, and J. Vian, "Control of multiple UAVs for persistent surveillance: Algorithm description and hardware demonstration," in *AIAA Infotech@Aerospace Conf. and AIAA Unmanned...Unlimited Conf., AIAA-2009-1852*, Apr. 2009, pp. 1–24.



Nikhil Nigam received the B.S. degree in aerospace engineering from the Indian Institute of Technology (IIT)—Bombay, Mumbai, India, in 2003, and the M.S. and Ph.D. degrees in aeronautics and astronautics from Stanford University, Stanford, CA, in 2004 and 2009, respectively.

He is currently a Research Scientist at Intelligent Automation Inc., Rockville, MD, where he is currently working on NextGen concepts and accurate positioning systems on-board aircraft. His primary interest is in controls and optimization applied to aircraft. His research has focused on aircraft control and design with emphasis on multi-agent systems and artificial intelligence.



Stefan Bieniawski received the B.S. and M.S. in aerospace engineering from Penn State University, University Park, in 1989 and 1992, respectively, and the Ph.D. degree in aeronautics and astronautics from Stanford University, Stanford, CA, in 2005.

He is an Associate Technical Fellow in Boeing Research and Technology, Seattle, WA, with interests in flight control systems design and analysis focusing on multi-disciplinary, unconventional, and collaborative control concepts. He has more than 10 years experience in the areas of structural dynamics, aeroelasticity, control synthesis, and optimization.



Ilan Kroo received the B.S. degree in physics in 1978, and the Ph.D. degree in aeronautics in 1983, both from Stanford University, Stanford, CA.

He is a Professor of Aeronautics and Astronautics at Stanford University, Stanford, CA. He worked in the Advanced Aerodynamic Concepts Branch at NASA's Ames Research Center for four years before returning to Stanford as a member of the Aero/Astro faculty. His research in aerodynamics and multidisciplinary design optimization includes the study of innovative airplane concepts for reduced environmental impact and efficient supersonic flight.



John Vian (M'86–SM'92) received the B.S. degree in mechanical engineering from Purdue University, West Lafayette, IN, in 1981, and the M.S. degree in aeronautical engineering, and the Ph.D. degree in electrical engineering from Wichita State University, Wichita, KS, in 1986 and 1991, respectively.

He is currently a Technical Fellow in Boeing Research and Technology, Seattle, WA, responsible for leading multi-vehicle autonomous systems research. He has 28 years experience in flight controls, vehicle health management, and autonomous systems, and has taught at Embry-Riddle Aeronautical University and Cogswell College.