

Michael J. Hirsch and Daniel Schroeder

Contents

64.1	Introduction	1578
64.2	Programming Formulation	1581
64.2.1	Problem Definition	1581
64.2.2	Parameters	1583
64.2.3	Decision Variables	1584
64.2.4	Constraints	1586
64.2.5	Objective Function (for UAV i)	1589
64.2.6	Discussion on Complexity	1590
64.3	Heuristic Algorithm	1590
64.4	Experimental Results	1591
64.4.1	Test Environment	1591
64.4.2	Experiments	1591
64.5	Conclusions and Future Research Areas	1597
	References	1598

Abstract

This chapter is concerned with dynamically determining appropriate flight patterns for a set of autonomous UAVs in an urban environment, with multiple mission goals. The UAVs are tasked with searching the urban region for targets of interest and tracking those targets that have been detected. It is assumed that there are limited communication capabilities between the UAVs and that there exist possible line of sight constraints between the UAVs and the targets. Each UAV (i) operates its own dynamic feedback loop, in a receding-horizon

M.J. Hirsch (✉)

Intelligence, Information, and Services, Raytheon Company, Maitland, FL, USA

e-mail: mhirsch@iseatek.com; www.iseatek.com

D. Schroeder

Intelligence, Information, and Services, Raytheon Company, State College, PA, USA

e-mail: Daniel.E.Schroeder@raytheon.com

framework, incorporating local information (from UAV i perspective) as well as remote information (from the perspective of the “neighbor” UAVs) to determine the task to perform and the optimal flight path of UAV i over the planning horizon. This results in a decentralized and more realistic model of the real-world situation. As the coupled task assignment and flight route optimization formulation is NP-hard, a hybrid heuristic for continuous global optimization is developed to solve for the flight plan and tasking over the planning horizon. Experiments are considered as communication range between UAVs varies.

64.1 Introduction

Military forces face ever increasing challenges to provide timely and accurate information on targets of interest, especially those targets that are mobile and elusive in nature. These tasks expand significantly in complexity when the targets are operating in an urban environment. The use of multiple unmanned vehicles to provide this target information allows our military personnel to stay out of the line of fire. However, many remotely controlled UAVs (i.e., swarms) require as many skilled pilots as there are swarm members, and these pilots must be able to deconflict airspace demands, mission requirements, and situational changes in near real time (Bamburger et al. 2006). On the other hand, autonomous unmanned vehicles allow military personnel to focus on more important issues like interpreting the gathered information, as opposed to determining how to acquire the information (Hirsch et al. 2007). Hence, there is a need to build intelligent unmanned vehicles that can plan and adapt autonomously to the environment they perceive. The clear benefit is shortened mission-critical decision chains.

There has recently been much research done in the area of autonomous vehicle control for surveillance-type missions. Almost all of the research has dealt with centralized cooperative control, with little research concerned with the more realistic decentralized problem (Shima and Rasmussen 2009). Steinberg (2006) provided an overview on research and limitations of autonomous technologies for the control of heterogeneous unmanned naval vehicles. Experiments in this paper examined aspects such as multi-vehicle task allocation, dynamic replanning of vehicle tasks, as well as human-in-the-loop management. Constraints considered in the experiments included pop-up threats, adverse weather conditions, and communication issues between the autonomous vehicles, among others. Liang et al. (2010) present a decision support framework to accurately reconstruct the current picture of the battlespace, by correctly identifying target tracks (i.e., target location, movement, identity) as seen by multiple wide area search munitions. Ahmadzadeh et al. (2006) described their Time Critical Coverage planner as a component of the Office of Naval Research Autonomy program, ICARUS. Each autonomous vehicle was modeled as a Dubin’s vehicle (Dubins 1957), whereby the vehicles were assumed to be point masses with constant speed and a prescribed minimum turning radius. The vehicles also had prescribed starting and ending spatial-temporal locations, as well as polygonal obstacles to be avoided throughout flight. The objective was to

determine the flight path of the UAVs to maximize the total sensor footprint over the region of interest. The algorithm utilized to solve this problem was based on sampling a discretized search graph (LaValle 2006).

Ma and Miller (2005) implemented a receding-horizon approach to solve a mixed-integer linear program modeling a trajectory planning problem. They limited their focus to a single vehicle navigating through three-dimensional obstacles and terrain. They considered threat avoidance and made use of the commercial software CPLEX (ILOG CPLEX 2008) to solve their problem. Shetty et al. (2008) considered the strategic routing of multiple unmanned combat vehicles to service multiple potential targets in space. They formulated this as a mixed-integer linear program, and through a decomposition scheme looked at solving the target assignment problem (vehicles to targets) and then determining the tour that each vehicle should take to service their assigned targets (a classical vehicle routing problem). They implemented a tabu-search heuristic to find solutions to their problem. However, they assumed the vehicles were holonomic, which enabled the mixed-integer linear program formulation. Kenefic (2008) utilized techniques from particle swarm optimization (Kennedy and Eberhart 1995) to efficiently define tours for a Dubin's vehicle, visiting multiple ground targets in two-dimensional space. Schumacher and Shima (2009) considered the problem of wide area search munitions, which are capable of searching for, identifying, and attacking targets. Whenever a new target is found, or a new task needs to be assigned, a capacitated transshipment assignment problem is solved, to determine the optimal assignment of munitions to tasks. Note that from one solution to the next solution, the assignment can change significantly.

Sylvester et al. (2004) presented a path-planning algorithm for two non-holonomic vehicles engaged in a docking maneuver. A bijection search method was used to determine the trajectory of the vehicles, such that the minimum turning radius constraint of each vehicle was satisfied. Santilli et al. (1995) and Bicchi et al. (1995) considered planning a path for a non-holonomic autonomous vehicle in the presence of fixed obstacles. Jiang et al. (1999) also presented a path planning algorithm for a non-holonomic vehicle subject to obstacles. This algorithm employed a global and local strategy to find the shortest path in two dimensions that was free of collisions with the obstacles. Schouwenaars et al. (2001) considered fuel-optimal paths for multiple vehicles. They formulated the problem as a mixed-integer linear program. The vehicles needed to move from an initial to final state, while avoiding vehicular collisions, as well as stationary and moving obstacles. Obstacle positions were assumed known a priori.

Kingston and Beard (2007) presented an algorithm to keep moving UAVs equally spaced (angularly) about a stationary target. The UAVs adapted their spacing based on local communication with other UAVs. Velocity bounds were derived so that the UAVs satisfied their kinematic constraints. They extended this approach to the case when the target was moving, but the path and velocity of the target was assumed known throughout. Gu et al. (2006) discussed the problem of cooperative estimation using a team of UAVs. They considered the goal of organizing the team's configuration to achieve optimal position and velocity estimates of a moving ground target. Sinclair et al. (2007) and Doucette et al. (2009) considered a problem

of cooperative control of aerial munitions during the attack phase of a stationary ground target, in order to refine the estimates of the target location. A dynamic programming solution was implemented to determine the paths for the aerial munitions. McLain et al. (2000) presented a cooperative control problem for UAVs to rendezvous at given locations, at the same time. The UAVs considered threat and fuel constraints, in determining their trajectories. A decomposition approach was employed, whereby at the team level, an arrival time was determined, and at the individual UAV level, an optimal path to reach the specified location at the arrival time was determined.

Rafi et al. (2006) presented an approach to have an autonomous vehicle follow a moving ground target. Their technique specified positional updates for the vehicle, in terms of the next time-step turning rate as well as orientation updates for the gimbaled camera mounted on the vehicle. They tested their approach using one vehicle and one moving ground target. Sinha et al. (2005) considered cooperative control for tracking of targets by multiple UAVs. An information theoretic criterion was used to determine the future path of each UAV, so that the total information gain (corresponding to the detected targets) was maximized. The UAVs were assumed to have fixed speed, and each UAV determined its angular turn rate to maximize the expected information gain for the next-time step. Akselrod et al. (2007) presented an approach for multi-target tracking of ground targets using UAVs via an hierarchical Markov decision process framework and dynamic programming. Their objective was to maximize the information gained by the UAVs. Sinha et al. (2004) presented a method to determine the location of sensors for tracking of moving ground targets, so as to maximize the resultant information gain. A genetic algorithm was implemented to determine the sensor positions. Of note here is that it was assumed that the targets were moving in the same direction as the sensors and their relative position remained constant over time.

Close-range maneuvering and following of a moving target continue to pose significant research challenges (Rafi et al. 2006). Close-range maneuvering requires real-time dynamic replanning and decision making as well as optimization of many parameters (taking into account the physical constraints of the vehicle). The problem becomes even more complex when the goal is to track a moving target for which the target dynamics is not known. In Hirsch et al. (2012, 2011) and Hirsch and Ortiz-Pena (2010), Hirsch et al. developed a decentralized approach for the autonomous cooperative control of UAVs, with the goal of persistent and accurate tracking of moving ground targets in an urban domain. The UAVs were able to share limited information with neighboring UAVs (other UAVs in their communication region) and had to dynamically replan their flight paths, incorporating predicted target movements and replanned flight paths of their neighbors into their own decision-making process. Line of sight also plays a big role in the flight plan of the UAVs and was incorporated into the problem through the use of Plücker coordinates (Semple and Kneebone 1952; Stolfi 1991). However, it was assumed that the targets were already known.

In this chapter, the approach of Hirsch et al. (2012, 2011) is extended to include, in a decentralized framework, the UAVs actively searching for and detecting

the targets, before tracking of the targets can commence. Since the UAVs have no knowledge of how many targets are present in the environment, at each decision-making step, the UAVs need to determine both the task they will perform (either searching for new targets or tracking those targets already detected) as well as determining the flight route to optimally accomplish their task. This results in a coupled task assignment/route planning mathematical formulation, which is highly nonlinear. A GRASP-SA heuristic is developed to efficiently solve the coupled problem. The solution quality is examined to determine its sensitivity to the communication capability of the UAVs. This chapter is organized as follows. Section 64.2 rigorously defines the problem of interest, discusses the dynamic feedback loop, and formulates the coupled task assignment/route planning problem as a nonlinear mathematical program. A hybrid heuristic for continuous global optimization is described in Sect. 64.3. This heuristic is utilized to solve the optimization problem at each time-step, for each UAV. Experimental results as the communication radius of the UAVs is varied are presented in Sect. 64.4. Section 64.5 provides some concluding remarks and future research directions.

64.2 Programming Formulation

64.2.1 Problem Definition

The overall problem addressed here is to provide (i) continual surveillance over a region of interest and (ii) accurately track all targets detected in this region. Hence, there are two tasks that a UAV can be performing at any given time: searching for targets or tracking those targets already detected. For this problem, there are N homogeneous autonomous vehicles flying at a fixed altitude. The vehicles are modeled as non-holonomic point masses on a two-dimensional plane with a minimum turning radius (i.e., a Dubin's vehicle (Dubins 1957)). The vehicles have minimum and maximum speed restrictions, as well as a maximum communication range, beyond which they cannot share information. Each vehicle has an independent internal representation of the surveillance region as a continuous search uncertainty map. For each location \vec{Y} in the region, the search uncertainty map for a particular UAV stores the uncertainty (from this UAV's perspective) of a target currently at location \vec{Y} . While in search mode, the UAV can update its search uncertainty map using information from its own onboard sensor as well as information from other UAVs currently in search mode (that can communicate with this UAV, i.e., its "neighbor" UAVs). While in tracking mode, the UAV can only update its search uncertainty map using information from neighbor UAVs that are in search mode. Similarly, each UAV has its own internal representation of the targets in the region of interest. When in search mode, the UAV can only update its target map from those neighbor UAVs in track mode. When in track mode, the UAV can update its target map based upon its local sensors as well as information provided from neighbor UAVs in track mode. The one exception to this is if in search mode, it is possible

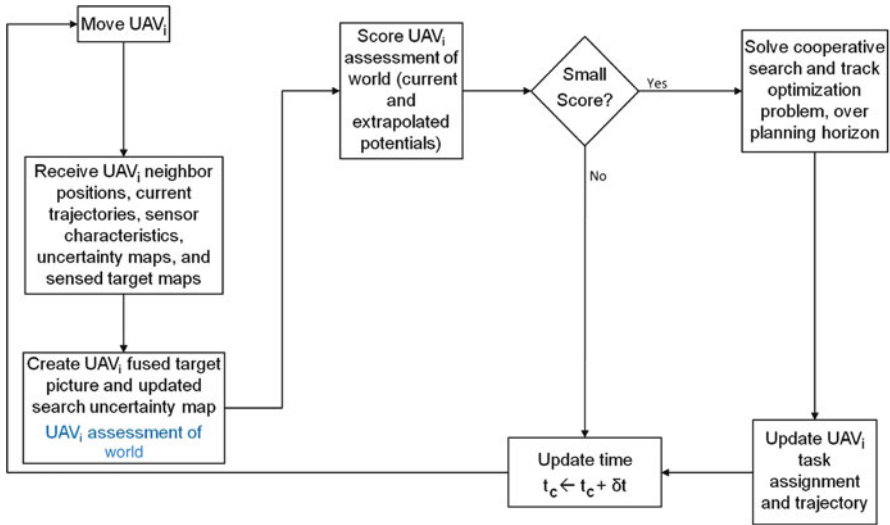


Fig. 64.1 Feedback control loop for UAV_{*i*}

(and hoped) that a UAV will detect a “new” target. When this happens, this target is added to the UAV’s stored target map, and shared with neighbor UAVs.

The vehicles operate in a decentralized manner; at each time-step, every UAV processes an equivalent version of the same dynamic feedback loop, as presented in Fig. 64.1. The feedback loop is discussed from the standpoint of UAV *i*. At the current time, t_c , the first step is for UAV *i* to move according to its planned route and continue to execute its current task (e.g., searching or tracking). Using onboard sensors, UAV *i* either searches its current field of view for new targets or attempts to sense the targets (known to UAV *i*) that are within its line of sight. The neighbors of UAV *i* send information relating to their (a) current positions, trajectories, and tasking; (b) sensor characteristics; (c) search uncertainty maps; and (d) sensed target maps. At this point, UAV *i* creates its assessment of the world, represented as a fused search uncertainty map and a fused target location map. This fused picture is scored in some qualitative manner, based upon the current positions of UAV *i* and its neighbor UAVs, their current planned trajectories and tasking, as well as the extrapolated search uncertainty map and target tracks, over the planning horizon. If the resultant score is small, or if the elapsed time since the last replan for UAV *i* is large, then UAV *i* solves a coupled tasking/route planning optimization problem for itself and its neighbor UAVs, over the planning horizon. The solution to this problem is a tasking for each UAV and route for each UAV to fly in order to accomplish their tasking. The mathematical formulation for this optimization problem is detailed in Sects. 64.2.2–64.2.5. It is worth noting that UAV *i* updates its own flight path and tasking, but does not share any of this computed information with its neighbor

UAVs. The reason for this is that the neighbors are going through this same feedback loop and may be incorporating additional knowledge not available to UAV i . The time is then incremented, and the dynamic loop continues.

64.2.2 Parameters

This section introduces the parameters incorporated into the dynamic tasking and flight plan optimization formulation. For the mathematical formulation to follow, the parameters, decision variables, and constraints are defined for all $\hat{i} \in \mathcal{N}_i(t)$, $t \in [t_c, t_c + \Delta_t]$, $\ell \in \{1, \dots, 5\}$, $h \in \{1, \dots, \mathcal{H}\}$, $j \in \{1, \dots, \mathcal{J}\}$, $u \in \{1, \dots, 4\}$, as appropriate:

t_c = The current time;

Δ_t = The current planning horizon;

C_d = The maximum distance at which any two UAVs can “communicate”, i.e., share information;

$\mathcal{N}_i(t)$ = The neighbors of UAV i , at time t , i.e., $\hat{i} \in \mathcal{N}_i(t) \implies \|P_i(t) - P_{\hat{i}}(t)\|_2 \leq C_d$. N.B.: $i \in \mathcal{N}_i(t)$;

The UAVs of interest (from UAV i 's perspective) are indexed by $\hat{i} \in \mathcal{N}_i(t)$;

The moving ground targets are indexed by $j \in \{1, \dots, \mathcal{J}\}$;

The buildings (3D, rectangular, opaque obstacles) are indexed by $h \in \{1, \dots, \mathcal{H}\}$;

$\Omega_{\hat{i}}$ = The minimum turning radius of UAV \hat{i} ;

$P_{\hat{i}}^c$ = The current position of UAV \hat{i} (N.B.: $P_{\hat{i}}^c = \{x_{\hat{i}}^c, y_{\hat{i}}^c, z_{\hat{i}}^c\}$);

$S_{\hat{i}}^{\min}$ = The minimum speed of UAV \hat{i} ;

$S_{\hat{i}}^{\max}$ = The maximum speed of UAV \hat{i} ;

ϵ_1^{\min} = The minimum distance allowed between any two UAVs (to avoid UAV collisions);

ϵ_2^{\min} = The minimum distance allowed between any UAV and any known building.

Assume this is set to 0, i.e., so that UAVs are constrained to not be within a building, but they can get infinitesimally close to the perimeter of each building.

N.B.: this assumption is valid, as one can always artificially increase the size of the buildings, as appropriate;

$Z_{\hat{i}}$ = The fixed altitude of UAV \hat{i} ;

$B_h = \{\mathcal{P}_{h\ell}\}_{\ell=1}^5$ = The representation of building h , as a set of 5 rectangular regions (i.e., faces) in \mathbb{R}^3 , all oriented in the same counterclockwise fashion;

$\mathcal{P}_{h\ell} = \{\rho_{h\ell 1}, \dots, \rho_{h\ell 4}\}$ = The ℓ -th oriented rectangle (i.e., face) of building h ;

$\rho_{h\ell u} = \{\bar{x}_{h\ell u}, \bar{y}_{h\ell u}, \bar{z}_{h\ell u}\}$ = The u -th point of the ℓ -th oriented rectangle (i.e., face) of building h ;

$\xi_{h\ell} = \{v_{h\ell}; d_{h\ell}\}$ = The plane containing face $\mathcal{P}_{h\ell}$, represented as a normal vector $v_{h\ell}$ and distance to the origin $d_{h\ell}$ ($v_{h\ell} = [\rho_{h\ell 3} - \rho_{h\ell 2}] \times [\rho_{h\ell 1} - \rho_{h\ell 2}]$, $d_{h\ell} = v_{h\ell} \cdot \rho_{h\ell 1}$);

$\gamma(x, y)$ = A function representing the “importance” level of position (x, y) . The higher this value, the more important it is to provide accurate track of targets near (x, y) ;

$\Lambda_{\hat{j}}(t)$ = The Plücker coordinate representation (Semple and Kneebone 1952; Stolfi 1991) of the oriented line from the position of UAV \hat{i} at time t , $P_{\hat{i}}(t)$, to the estimated position of target j , at time t , $X_j^i(t)$;

$\Lambda_{\hat{i}}(\vec{Y}, t)$ = The Plücker coordinate representation of the oriented line from the position of UAV \hat{i} at time t , $P_{\hat{i}}(t)$, to position \vec{Y} ;

$\hat{\Lambda}_{h\ell u, u+1}$ = The Plücker coordinate representation of the oriented line from $\rho_{h\ell u}$ to $\rho_{h\ell, u+1}$, with $u+1 \equiv 1$ when $u=4$;

$\mathcal{G}(\alpha, \hat{\alpha}) = \alpha_1 \hat{\alpha}_6 - \alpha_2 \hat{\alpha}_5 + \alpha_3 \hat{\alpha}_4 + \alpha_4 \hat{\alpha}_3 - \alpha_5 \hat{\alpha}_2 + \alpha_6 \hat{\alpha}_1$ where $\alpha = \{\alpha_1, \dots, \alpha_6\}$ and $\hat{\alpha} = \{\hat{\alpha}_1, \dots, \hat{\alpha}_6\}$ are the Plücker coordinate representations of two oriented lines;

$R_{\hat{i}}(t)$ = The measurement noise for the sensor onboard UAV \hat{i} , at time t . N.B.:
 $R_{\hat{i}}(t) = \begin{pmatrix} \sigma_{r_{\hat{i}}} & 0 \\ 0 & \sigma_{b_{\hat{i}}} \end{pmatrix}$, where $\sigma_{r_{\hat{i}}}$ = the range measurement error for the radar platform on UAV \hat{i} and $\sigma_{b_{\hat{i}}}$ = the bearing measurement error for the radar platform on UAV \hat{i} ;

$F_{\hat{i}}(t)$ = The system dynamics matrix for the sensor onboard UAV \hat{i} , at time t ;

$\underline{Q}_{\hat{i}}(t)$ = The process noise for the sensor onboard UAV \hat{i} , at time t ;

$H_{\hat{i}}(t)$ = The observation matrix for the sensor onboard UAV \hat{i} , at time t ;

$\bar{d}_{\hat{i}}$ = The probability of detection for the sensor onboard UAV \hat{i} ;

χ = The factor by which to increase the search uncertainty map if there is no line of sight to a given position (in Constraint (64.9));

$\mathcal{M}, \bar{\mathcal{M}}$ = Large enough constants (needed for Constraints (64.7), (64.12), (64.13), (64.15), (64.16), (64.18), (64.19), (64.21), (64.22), (64.28), (64.29), (64.31), (64.32), (64.34), (64.35), (64.37), and (64.38)).

64.2.3 Decision Variables

This section defines the main decision variables for the mathematical formulation:

$P_{\hat{i}}(t)$ = The position of UAV \hat{i} , at time t (N.B.: $P_{\hat{i}}(t) = \{x_{\hat{i}}(t), y_{\hat{i}}(t), Z_{\hat{i}}\}$);

$S_{\hat{i}}(t)$ = The speed of UAV \hat{i} , at time t ;

$\theta_{\hat{i}}(t)$ = The heading of UAV \hat{i} , at time t ;

$\{X_{\hat{j}}(t), \Sigma_{\hat{j}}(t)\}$ = The estimated position and covariance of target j by UAV \hat{i} , at time t ;

$\{X_j^i(t), \Sigma_j^i(t)\}$ = The fused position and covariance of target j , from UAV i 's perspective, at time t ;

$$\bar{\Phi}_{\hat{i}jh\ell u, u+1}(t) = \begin{cases} 1 & \text{if line segment } \Lambda_{\hat{j}}(t) \text{ is oriented counterclockwise} \\ & \text{with respect to line segment } \hat{\Lambda}_{h\ell u, u+1}, \text{ at time } t \\ 0 & \text{o.w.} \end{cases};$$

$\bar{\Phi}_{\hat{i}jh\ell}(t)$ = The number of line segments representing the ℓ -th face of building h that line segment $\Lambda_{\hat{j}}(t)$ is oriented counterclockwise with respect to, at time t ;

$$\beta_{\hat{i}jh\ell}(t) = \begin{cases} 1 & \text{if line segment } \Lambda_{\hat{j}}(t) \text{ does not intersect the } \ell\text{-th face} \\ & \text{of building } h \text{ at time } t \\ 0 & \text{o.w.} \end{cases};$$

$\bar{\beta}_{\hat{i}jh}(t)$ = The number of faces representing building h that line segment $\Lambda_{\hat{i}j}(t)$ does not intersect, at time t ;

$$\bar{\Psi}_{\hat{i}jh}(t) = \begin{cases} 1 & \text{if there is line of sight from UAV } \hat{i} \text{ to target } j \text{ with respect to} \\ & \text{building } h, \text{ at time } t \\ 0 & \text{o.w.} \end{cases} ;$$

$\hat{\Psi}_{\hat{i}j}(t)$ = The number of buildings that do not restrict line of sight from UAV \hat{i} to target j , at time t ;

$$\Psi_{\hat{i}j}(t) = \begin{cases} 1 & \text{if UAV } \hat{i} \text{ has line of sight to target } j, \text{ at time } t ; \\ 0 & \text{o.w.} \end{cases}$$

$$b_{\hat{i}} = \begin{cases} 1 & \text{if UAV } \hat{i} \text{ assigned to search mode over the planning horizon} \\ & [t_c, t_c + \delta_t] \\ 0 & \text{if UAV } \hat{i} \text{ assigned to track mode over the planning horizon} \\ & [t_c, t_c + \delta_t] \end{cases} ;$$

$\Gamma_i(\vec{Y}, t)$ = Search uncertainty map for UAV i at time t and position \vec{Y} ;

$$\bar{\Phi}_{\hat{i}h\ell u, u+1}(\vec{Y}, t) = \begin{cases} 1 & \text{if line segment } \Lambda_{\hat{i}}(\vec{Y}, t) \text{ is oriented counterclockwise} \\ & \text{with respect to line segment } \hat{\Lambda}_{h\ell u, u+1}, \text{ at time } t \\ 0 & \text{o.w.} \end{cases} ;$$

$\bar{\Phi}_{\hat{i}h\ell}(\vec{Y}, t)$ = The number of line segments representing the ℓ -th face of building h that line segment $\Lambda_{\hat{i}}(\vec{Y}, t)$ is oriented counterclockwise with respect to, at time t ;

$$\beta_{\hat{i}h\ell}(\vec{Y}, t) = \begin{cases} 1 & \text{if line segment } \Lambda_{\hat{i}}(\vec{Y}, t) \text{ does not intersect the } \ell\text{-th face} \\ & \text{of building } h, \text{ at time } t \\ 0 & \text{o.w.} \end{cases} ;$$

$\bar{\beta}_{\hat{i}h}(\vec{Y}, t)$ = The number of faces representing building h that line segment $\Lambda_{\hat{i}}(\vec{Y}, t)$ does not intersect, at time t ;

$$\bar{\Psi}_{\hat{i}h}(\vec{Y}, t) = \begin{cases} 1 & \text{if there is line of sight from UAV } \hat{i} \text{ to position } \vec{Y} \text{ with respect} \\ & \text{to building } h, \text{ at time } t \\ 0 & \text{o.w.} \end{cases} ;$$

$\hat{\Psi}_{\hat{i}}(\vec{Y}, t)$ = The number of buildings that do not restrict line of sight from UAV \hat{i} to position \vec{Y} , at time t ;

$$\Psi_{\hat{i}}(\vec{Y}, t) = \begin{cases} 1 & \text{if UAV } \hat{i} \text{ has line of sight to position } \vec{Y}, \text{ at time } t ; \\ 0 & \text{o.w.} \end{cases}$$

$$\Psi_o(\vec{Y}, t) = \begin{cases} 1 & \text{if neither UAV } i \text{ nor any neighbors of UAV } i \text{ has line of sight} \\ & \text{to position } \vec{Y}, \text{ at time } t \\ 0 & \text{o.w.} \end{cases} ;$$

64.2.4 Constraints

$$P_{\hat{i}}(t_c) = P_{\hat{i}}^c \quad \forall \hat{i} \tag{64.1}$$

$$\frac{\partial x_{\hat{i}}(t)}{\partial t} = S_{\hat{i}}(t) \cos [\theta_{\hat{i}}(t)] \quad \forall \hat{i}, t \tag{64.2}$$

$$\frac{\partial y_{\hat{i}}(t)}{\partial t} = S_{\hat{i}}(t) \sin [\theta_{\hat{i}}(t)] \quad \forall \hat{i}, t \tag{64.3}$$

$$\left| \frac{\partial \theta_{\hat{i}}(t)}{\partial t} \right| \leq \Omega_{\hat{i}} \quad \forall \hat{i}, t \tag{64.4}$$

$$S_{\hat{i}}^{\min} \leq S_{\hat{i}}(t) \leq S_{\hat{i}}^{\max} \quad \forall \hat{i}, t \tag{64.5}$$

$$\| P_{\hat{i}_1}(t) - P_{\hat{i}_2}(t) \|_2 \geq \varepsilon_1^{\min} \quad \forall \hat{i}_1, \hat{i}_2, t \tag{64.6}$$

$$P_{\hat{i}}(t) \cdot v_{h\ell} + d_{h\ell} \geq -\mu_{\hat{i}h\ell}(t) \mathcal{M} \quad \forall \hat{i}, h, \ell, t \tag{64.7}$$

$$\sum_{\ell=1}^5 \mu_{\hat{i}h\ell}(t) \leq 4 \quad \forall \hat{i}, h, \ell, t \tag{64.8}$$

$$\begin{aligned} \frac{\partial \Gamma_{\hat{i}}(\vec{Y}, t)}{\partial t} = & \left[\prod_{\hat{i} \in \mathcal{N}_{\hat{i}}(t)} (1 - \bar{d}_{\hat{i}})^{b_{\hat{i}} \psi_{\hat{i}}(\vec{Y}, t)} \right] \cdot \Gamma_{\hat{i}}(\vec{Y}, t) \\ & + \psi_o(\vec{Y}, t) \cdot \chi \cdot \Gamma_{\hat{i}}(\vec{Y}, t) \quad \forall t, \vec{Y} \end{aligned} \tag{64.9}$$

$$\psi_o(\vec{Y}, t) \leq 1 - b_{\hat{i}} \psi_{\hat{i}}(\vec{Y}, t) \quad \forall \hat{i}, t, \vec{Y} \tag{64.10}$$

$$\psi_o(\vec{Y}, t) + \sum_{\hat{i} \in \mathcal{N}_{\hat{i}}(t)} b_{\hat{i}} \psi_{\hat{i}}(\vec{Y}, t) > 0 \quad \forall t, \vec{Y} \tag{64.11}$$

$$\mathcal{G}(\Lambda_{\hat{i}}(\vec{Y}, t), \hat{\Lambda}_{h\ell u, u+1}) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} \leq \bar{\phi}_{\hat{i}h\ell u, u+1}(\vec{Y}, t) \mathcal{M} \quad \forall \hat{i}, \vec{Y}, h, \ell, u, t \tag{64.12}$$

$$-\mathcal{G}(\Lambda_{\hat{i}}(\vec{Y}, t), \hat{\Lambda}_{h\ell u, u+1}) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} < (1 - \bar{\phi}_{\hat{i}h\ell u, u+1}(\vec{Y}, t)) \mathcal{M} \quad \forall \hat{i}, \vec{Y}, h, \ell, u, t \tag{64.13}$$

$$\bar{\phi}_{\hat{i}h\ell}(\vec{Y}, t) = \sum_{u=1}^4 \bar{\phi}_{\hat{i}h\ell u, u+1}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, h, \ell, t \tag{64.14}$$

$$\beta_{\hat{i}h\ell}(\vec{Y}, t) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} \leq 4 - \bar{\phi}_{\hat{i}h\ell}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, h, \ell, t \tag{64.15}$$

$$\beta_{\hat{i}h\ell}(\vec{Y}, t) \mathcal{M} + (1 - b_{\hat{i}}) \bar{\mathcal{M}} \geq 4 - \bar{\phi}_{\hat{i}h\ell}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, h, \ell, t \tag{64.16}$$

$$\bar{\beta}_{\hat{i}h}(\vec{Y}, t) = \sum_{\ell=1}^5 \beta_{\hat{i}h\ell}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, h, t \tag{64.17}$$

$$\bar{\psi}_{\hat{i}h}(\vec{Y}, t) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} \leq \beta_{\hat{i}h\ell}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, h, t \tag{64.18}$$

$$1 - \bar{\Psi}_{\hat{i}h}(\vec{Y}, t) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} \leq 5 - \bar{\beta}_{\hat{i}h}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, h, t \quad (64.19)$$

$$\hat{\Psi}_{\hat{i}}(\vec{Y}, t) = \sum_{h=1}^{\mathcal{H}} \bar{\Psi}_{\hat{i}h}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, t \quad (64.20)$$

$$\Psi_{\hat{i}}(\vec{Y}, t) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} \leq \bar{\Psi}_{\hat{i}h}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, t \quad (64.21)$$

$$1 - \Psi_{\hat{i}}(\vec{Y}, t) - (1 - b_{\hat{i}}) \bar{\mathcal{M}} \leq \mathcal{H} - \hat{\Psi}_{\hat{i}}(\vec{Y}, t) \quad \forall \hat{i}, \vec{Y}, t \quad (64.22)$$

$$\Sigma_j^i(t) = \left[\sum_{\hat{i} \in \mathcal{N}_i(t)} (1 - b_{\hat{i}}) \Psi_{\hat{i}j}(t) \Sigma_{\hat{i}j}^{-1}(t) \right]^{-1} \quad \forall j, t \quad (64.23)$$

$$X_j^i(t) = \Sigma_j^i(t) \cdot \left[\sum_{\hat{i} \in \mathcal{N}_i(t)} (1 - b_{\hat{i}}) \Psi_{\hat{i}j}(t) \Sigma_{\hat{i}j}^{-1}(t) X_{\hat{i}j}(t) \right] \quad \forall j, t \quad (64.24)$$

$$\frac{\partial X_{\hat{i}j}(t)}{\partial t} = (1 - b_{\hat{i}}) [F_{\hat{i}}(t) X_{\hat{i}j}(t) + K_{\hat{i}}(t) [\bar{z}_{\hat{i}j}(t) - H_{\hat{i}}(t) X_{\hat{i}j}(t)]] \quad \forall \hat{i}, j, t \quad (64.25)$$

$$\begin{aligned} \frac{\partial \Sigma_{\hat{i}j}(t)}{\partial t} &= (1 - b_{\hat{i}}) [F_{\hat{i}}(t) \Sigma_{\hat{i}j}(t) + \Sigma_{\hat{i}j}(T) F_{\hat{i}}^T(t) + \bar{Q}_{\hat{i}}(t) \\ &\quad - K_{\hat{i}}(t) R_{\hat{i}}(t) K_{\hat{i}}^T(t)] \quad \forall \hat{i}, j, t \end{aligned} \quad (64.26)$$

$$K_{\hat{i}}(t) = \Sigma_{\hat{i}j}(t) H_{\hat{i}}^T(t) R_{\hat{i}}^{-1}(t) \quad \forall \hat{i}, j, t \quad (64.27)$$

$$\mathcal{G}(\Lambda_{\hat{i}j}(t), \hat{\Lambda}_{h\ell u, u+1}) - b_{\hat{i}} \bar{\mathcal{M}} \leq \bar{\phi}_{\hat{i}jh\ell u, u+1}(t) \mathcal{M} \quad \forall \hat{i}, j, h, \ell, u, t \quad (64.28)$$

$$-\mathcal{G}(\Lambda_{\hat{i}j}(t), \hat{\Lambda}_{h\ell u, u+1}) - b_{\hat{i}} \bar{\mathcal{M}} < (1 - \bar{\phi}_{\hat{i}jh\ell u, u+1}(t)) \mathcal{M} \quad \forall \hat{i}, j, h, \ell, u, t \quad (64.29)$$

$$\bar{\phi}_{\hat{i}jh\ell}(t) = \sum_{u=1}^4 \bar{\phi}_{\hat{i}jh\ell u, u+1}(t) \quad \forall \hat{i}, j, h, \ell, t \quad (64.30)$$

$$\beta_{\hat{i}jh\ell}(t) - b_{\hat{i}} \bar{\mathcal{M}} \leq 4 - \bar{\phi}_{\hat{i}jh\ell}(t) \quad \forall \hat{i}, j, h, \ell, t \quad (64.31)$$

$$\beta_{\hat{i}jh\ell}(t) \mathcal{M} + b_{\hat{i}} \bar{\mathcal{M}} \geq 4 - \bar{\phi}_{\hat{i}jh\ell}(t) \quad \forall \hat{i}, j, h, \ell, t \quad (64.32)$$

$$\bar{\beta}_{\hat{i}jh}(t) = \sum_{\ell=1}^5 \beta_{\hat{i}jh\ell}(t) \quad \forall \hat{i}, j, h, t \quad (64.33)$$

$$\bar{\Psi}_{\hat{i}jh}(t) - b_{\hat{i}} \bar{\mathcal{M}} \leq \beta_{\hat{i}jh\ell}(t) \quad \forall \hat{i}, j, h, t \quad (64.34)$$

$$1 - \bar{\Psi}_{\hat{i}jh}(t) - b_{\hat{i}} \bar{\mathcal{M}} \leq 5 - \bar{\beta}_{\hat{i}jh}(t) \quad \forall \hat{i}, j, h, t \quad (64.35)$$

$$\hat{\Psi}_{\hat{i}j}(t) = \sum_{h=1}^{\mathcal{H}} \bar{\Psi}_{\hat{i}jh}(t) \quad \forall \hat{i}, j, t \quad (64.36)$$

$$\Psi_{\hat{i}j}(t) - b_{\hat{i}} \bar{\mathcal{M}} \leq \bar{\Psi}_{\hat{i}jh}(t) \quad \forall \hat{i}, j, t \quad (64.37)$$

$$1 - \psi_{\hat{i}j}(t) - b_{\hat{i}}\bar{M} \leq \mathcal{H} - \hat{\psi}_{\hat{i}j}(t) \quad \forall \hat{i}, j, t \quad (64.38)$$

$$P_{\hat{i}}(t) \in \mathbb{R}^3 \quad \forall \hat{i}, t \quad (64.39)$$

$$S_{\hat{i}}(t) \in \mathbb{R}_+ \quad \forall \hat{i}, t \quad (64.40)$$

$$\theta_{\hat{i}}(t) \in [-\pi, \pi] \quad \forall \hat{i}, t \quad (64.41)$$

$$\bar{\phi}_{\hat{i}h\ell}(\vec{Y}, t), \hat{\psi}_{\hat{i}}(\vec{Y}, t), \bar{\beta}_{\hat{i}h}(\vec{Y}, t) \in \mathbb{Z}_+ \quad \forall \hat{i}, \vec{Y}, h, \ell, t \quad (64.42)$$

$$\bar{\phi}_{\hat{i}j h\ell}(t), \hat{\psi}_{\hat{i}j}(t), \bar{\beta}_{\hat{i}jh}(t) \in \mathbb{Z}_+ \quad \forall \hat{i}, j, h, \ell, t \quad (64.43)$$

$$\bar{\phi}_{\hat{i}h\ell u, u+1}(\vec{Y}, t), \psi_{\hat{i}}(\vec{Y}, t), \psi_o(\vec{Y}, t), \beta_{\hat{i}h\ell}(\vec{Y}, t) \in \{0, 1\} \quad \forall \hat{i}, \vec{Y}, h, \ell, u, t \quad (64.44)$$

$$\bar{\phi}_{\hat{i}j h\ell u, u+1}(t), \psi_{\hat{i}j}(t), \beta_{\hat{i}jh\ell}(t), \bar{\psi}_{\hat{i}jh}(t) \in \{0, 1\} \quad \forall \hat{i}, j, h, \ell, u, t \quad (64.45)$$

$$\mu_{\hat{i}h\ell}(t) \in \{0, 1\} \quad \forall \hat{i}, j, h, \ell, t \quad (64.46)$$

$$b_{\hat{i}} \in \{0, 1\} \quad \forall \hat{i} \quad (64.47)$$

Interpretation of Constraints

Constraints (64.1) prescribe initial positions for each of the UAVs in $\mathcal{N}_i(t)$;

Constraints (64.2) and (64.3) define the motion of UAV \hat{i} , as a function of the speed and heading of the UAV;

Constraints (64.4) restrict the instantaneous change in heading of UAV \hat{i} to be less than $\Omega_{\hat{i}}$ (modeling the UAVs as Dubin's vehicles);

Constraints (64.5) define minimum and maximum speed restrictions UAV \hat{i} ;

Constraints (64.6) enforce no two UAVs being closer than ε_1^{\min} ;

Constraints (64.7) and (64.8) prescribe that no UAV trajectory can intersect with the boundary of any building;

Constraints (64.9)–(64.11) determine the updated search uncertainty map for UAV i at time t ;

Constraints (64.12)–(64.22) are only enforced if UAV \hat{i} is assigned to a search task over the planning horizon $[t_c, t_c + \delta_t]$:

Constraints (64.12) and (64.13) force $\bar{\phi}_{\hat{i}h\ell u, u+1}(\vec{Y}, t)$ to be 1 when line segment $\Lambda_{\hat{i}}(\vec{Y}, t)$ is oriented counterclockwise with respect to the line segment $\hat{\Lambda}_{h\ell u, u+1}$, at time t , and to be 0 otherwise;

Constraints (64.14) count the number of line segments representing the ℓ -th face of building h that line segment $\Lambda_{\hat{i}}(\vec{Y}, t)$ is oriented counterclockwise with respect to, at time t ;

Constraints (64.15) and (64.16) force $\beta_{\hat{i}h\ell}(\vec{Y}, t)$ to be 1 when line segment $\Lambda_{\hat{i}}(\vec{Y}, t)$ does not intersect the ℓ -th face of building h , at time t , and to be 0 otherwise;

Constraints (64.17) count the number of faces of building h that do not inhibit line of sight from UAV \hat{i} to position \vec{Y} , at time t ;

Constraints (64.18) and (64.19) force $\bar{\psi}_{\hat{i}h}(\vec{Y}, t)$ to be 1 when there is line of sight from UAV \hat{i} to position \vec{Y} , at time t , with respect to building h , and to be 0 otherwise;

Constraints (64.20) count the number of buildings that do not inhibit line of sight from UAV \hat{i} to position \vec{Y} , at time t ;

Constraints (64.21) and (64.22) force $\psi_{\hat{i}}(t)$ to be 1 when UAV \hat{i} has line of sight to position \bar{Y} , at time t , and to be 0 otherwise;

Constraints (64.23) and (64.24) determine the fused covariance and state for target j at time t , from UAV i 's perspective, based on those neighbors of UAV i that have line of sight to target j ;

Constraints (64.25)–(64.27) are the continuous time Kalman filter (Kalman-Bucy filter) equations for the state of target j , as seen from UAV \hat{i} , where $\bar{z}_{ij}(t)$ represents the measurement of target j by UAV \hat{i} , at time t (N.B.: a detailed explanation of the Kalman filter equations can be found in Blackman and Popoli (1999), Gelb (1974), and Maybeck (1994));

Constraints (64.28)–(64.38) are only enforced if UAV \hat{i} is assigned to a track task over the planning horizon $[t_c, t_c + \delta_t]$:

Constraints (64.28) and (64.29) force $\bar{\phi}_{i_j h \ell u, u+1}(t)$ to be 1 when line segment $\Lambda_{ij}(t)$ is oriented counterclockwise with respect to the line segment $\hat{\Lambda}_{h \ell u, u+1}$, at time t , and to be 0 otherwise;

Constraints (64.30) count the number of line segments representing the ℓ -th face of building h that line segment $\Lambda_{ij}(t)$ is oriented counterclockwise with respect to, at time t ;

Constraints (64.31) and (64.32) force $\beta_{i_j h \ell}(t)$ to be 1 when line segment $\Lambda_{ij}(t)$ does not intersect the ℓ -th face of building h , at time t , and to be 0 otherwise;

Constraints (64.33) count the number of faces of building h that do not inhibit line of sight from UAV \hat{i} to target j , at time t ;

Constraints (64.34) and (64.35) force $\bar{\psi}_{i_j h}(t)$ to be 1 when there is line of sight from UAV \hat{i} to target j , at time t , with respect to building h , and to be 0 otherwise;

Constraints (64.36) count the number of buildings that do not inhibit line of sight from UAV \hat{i} to target j , at time t ;

Constraints (64.37) and (64.38) force $\psi_{i_j}(t)$ to be 1 when UAV \hat{i} has line of sight to target j , at time t , and to be 0 otherwise;

Constraints (64.23)–(64.27) prescribe domain restrictions on all of the decision variables.

64.2.5 Objective Function (for UAV i)

The objective function, F , as defined in Eq. (64.48), minimizes the sum of two components. The first component computes the expectant volume of UAV i 's search uncertainty map over the planning horizon, $[t_c, t_c + \delta_t]$. The second component computes the weighted sum of the determinants of the fused covariances for the targets that are known to UAV i , over the planning horizon.

$$F = \min \left[a_s \int_{t_c}^{t_c + \Delta_t} \int_{\mathbb{R}^2} \Gamma_i(\mathcal{Y}, \tau) d\mathcal{Y} d\tau + a_t \int_{t_c}^{t_c + \Delta_t} \sum_{j=1}^{\mathcal{J}} \left[\gamma(X_j^i(\tau)) \left| \Sigma_j^i(\tau) \right| \right] d\tau \right] \quad (64.48)$$

64.2.6 Discussion on Complexity

It is easy to see that our problem of determining the flight plan for a set of autonomous UAVs is a generalization of the standard vehicle routing problem (VRP). While there are many variants to the standard VRP, the general form considers m vehicles, starting at a depot, tasked with delivering goods or services to a set of n customers. Each customer has a certain required demand, and each vehicle has a capacity. In addition, each vehicle must start and end their route at the depot (a tour).

A solution to this problem is a set of tours for which each customer is visited only once while respecting capacity and demand constraints. The objective can be to minimize the total distance traveled by all vehicles, to minimize the maximum distance traveled by any one vehicle, or to minimize the number of vehicles required to service the customers. The VRP is known to be NP-hard (Garey and Johnson 1979), and thus our problem is NP-hard as well.

64.3 Heuristic Algorithm

There are many papers dealing with UAV routing (Shetty et al. 2008; Ma and Miller 2005; Schouwenaars et al. 2001; Sinha et al. 2005) that formulate the problem statement as a linear or mixed-integer linear program (or a formulation that can be linearized in the standard way). This allows the authors to use various proven heuristics to find good routes for the UAVs. However, for UAV routing, incorporating more realistic constraints on the movement of the UAVs necessitates the introduction of nonlinear constraints that cannot be linearized without approximation of the nonlinear constraints. Hence, the need to utilize more generalized heuristics (or meta-heuristics) in the search for near-optimal solutions.

To efficiently solve the optimization problem presented in Sects. 64.2.2–64.2.5, a hybrid heuristic was developed, combining Greedy Randomized Adaptive Search Procedures (GRASP) and Simulated Annealing (SA). GRASP is a multi-start local search procedure, where each iteration consists of two phases, a construction phase and a local search phase (Feo and Resende 1989, 1995; Festa and Resende 2002). In the construction phase, interactions between greediness and randomization generate a diverse set of quality solutions. The local search phase improves upon the solutions found in the construction phase. In this GRASP implementation, only the construction portion is utilized. Simulated Annealing is a heuristic to find good-quality solutions to optimization problems by approximating the cooling process of metals (Kirkpatrick et al. 1983). At each step, a current solution is perturbed. If the perturbation results in a better solution, then the current solution is replaced. If the perturbed solution is worse than the current solution, it still might replace the current solution; replacement will occur with a probability based on the distance between the current and perturbed solution values and the current temperature in the annealing process. As the heuristic progresses, the temperature is lowered, making it more and more unlikely to replace the current solution with a worse perturbed solution.

For our hybrid GRASP-SA heuristic, the GRASP heuristic determines the task assignment for the UAVs, interacting directly with a Simulated Annealing heuristic which determines the routes that the UAVs should fly to accomplish their tasks. The best solution over all of the GRASP multi-start iterations is retained as the final solution. It is important to note that a solution, for a given UAV, is the tasking and flight path for the UAV, as well as the neighbors of the UAV, over the planning horizon. Figure 64.2 provides pseudocode for the GRASP-SA hybrid heuristic, with the function Ξ representing the SA heuristic function.

64.4 Experimental Results

64.4.1 Test Environment

The experiments to follow were carried out on a Hewlett-Packard with an Intel(R) Core(TM) 2 Duo E8400 CPU at 3.00 GHz and 3.71 GB of RAM running Microsoft Windows 7 Enterprise (64-bit). The implementation of the GRASP and Simulated Annealing heuristics as written in the C++ programming language and compiled with Microsoft Visual C++ 2008. The algorithm used for random-number generation is an implementation of the Mersenne Twister algorithm described in Matsumoto and Nishimura (1998).

64.4.2 Experiments

While no specific UAV was modeled, the experiments utilized characteristics of the Raven UAV. The simulations in this study model the UAV sensors as omnidirectional radars measuring range and bearing to the targets. Conversions of the measurement and measurement uncertainty from spherical to cartesian coordinates were done utilizing the standard transformations (Lerro and Bar-Shalom 1993). Table 64.1 displays the parameter values used during the simulation. Thirty-two buildings were modeled in the simulation, each with a height of 100 m, while the altitude of the 4 UAVs was fixed to 10 m. Eight moving ground targets were simulated, with minimum and maximum speeds of 0 and 30 miles h^{-1} , respectively. Three different experiments were performed, varying the communication range of the UAVs. In the “no-comms” case, each UAV had an effective communication radius of 0 m, making them truly independent of all the other UAVs. For the “half-comms” case, each UAVs communication radius was set to 979.5 m, half the length of the simulated region of interest. And in the “unlimited comms” case, each UAV had a communication radius larger than the simulated region of interest, meaning as long as there were no obstructions (e.g., buildings) between two UAVs, they could effectively communicate.

To highlight a small sample of the simulation, Figs. 64.3–64.5 show the results for time steps 44–46 of the unlimited communications case. This is a top-down view,

```

procedure GRASP-SA Heuristic (Problem Instance)
1  UAVsAssigned  $\leftarrow \emptyset$ ;
2  UAVAssignment  $\leftarrow \emptyset$ ;
3  UAVRoutes  $\leftarrow \emptyset$ ;
4   $\alpha \leftarrow \text{UnifRand}[0,1]$ ;
5  while |UAVsAssigned| < NumUAVs do
6      min  $\leftarrow +\infty$ ;
7      max  $\leftarrow -\infty$ ;
8      for  $\hat{i} = 1, \dots, \text{NumUAVs}$  do
9          if  $\hat{i} \notin \text{UAVsAssigned}$  then
10              $[g_{\hat{i}}, z_{\hat{i}}] \leftarrow \Xi(\hat{i}, \text{UAVAssignment}, \text{UAVRoutes}, \text{Search})$ ;
11             if min >  $g_{\hat{i}}$  then min  $\leftarrow g_{\hat{i}}$ ;
12             if max <  $g_{\hat{i}}$  then max  $\leftarrow g_{\hat{i}}$ ;
13              $[\bar{g}_{\hat{i}}, \bar{z}_{\hat{i}}] \leftarrow \Xi(\hat{i}, \text{UAVAssignment}, \text{UAVRoutes}, \text{Track})$ ;
14             if min >  $\bar{g}_{\hat{i}}$  then min  $\leftarrow \bar{g}_{\hat{i}}$ ;
15             if max <  $\bar{g}_{\hat{i}}$  then max  $\leftarrow \bar{g}_{\hat{i}}$ ;
16         end if
17     end for
18     RCL  $\leftarrow \emptyset$ ;
19     Threshold  $\leftarrow \text{min} + \alpha * (\text{max} - \text{min})$ ;
20     for  $\hat{i} = 1, \dots, \text{NumUAVs}$  do
21         if  $\hat{i} \notin \text{UAVsAssigned}$  then
22             if  $g_{\hat{i}} \leq \text{Threshold}$  then
23                 RCL  $\leftarrow \text{RCL} \cup \{[\hat{i}, z_{\hat{i}}, \text{Search}]\}$ ;
24             if  $\bar{g}_{\hat{i}} \leq \text{Threshold}$  then
25                 RCL  $\leftarrow \text{RCL} \cup \{[\hat{i}, \bar{z}_{\hat{i}}, \text{Track}]\}$ ;
26         end if
27     end for
28      $j \leftarrow \text{RandomlySelectElement}(\text{RCL})$ ;
29     UAVsAssigned  $\leftarrow \text{UAVsAssigned} \cup \{j_1\}$ ;
30     UAVAssignment  $\leftarrow \text{UAVAssignment} \cup \{j_3\}$ ;
31     UAVRoutes  $\leftarrow \text{UAVRoutes} \cup \{j_2\}$ ;
32 return(UAVAssignment, UAVRoutes);
end GRASP-SA Heuristic;

```

Fig. 64.2 Pseudo-code for the hybrid GRASP-SA heuristic

Table 64.1 Parameter settings

Name	Value
Δ_t	10
T_s	0
T_e	120
q	7
N	4
Ω_i	$\frac{\pi}{6}$
S_i^{\min}	30
S_i^{\max}	170
ϵ_1^{\min}	15
\mathcal{H}	32
z	10
\bar{d}_i	0.9

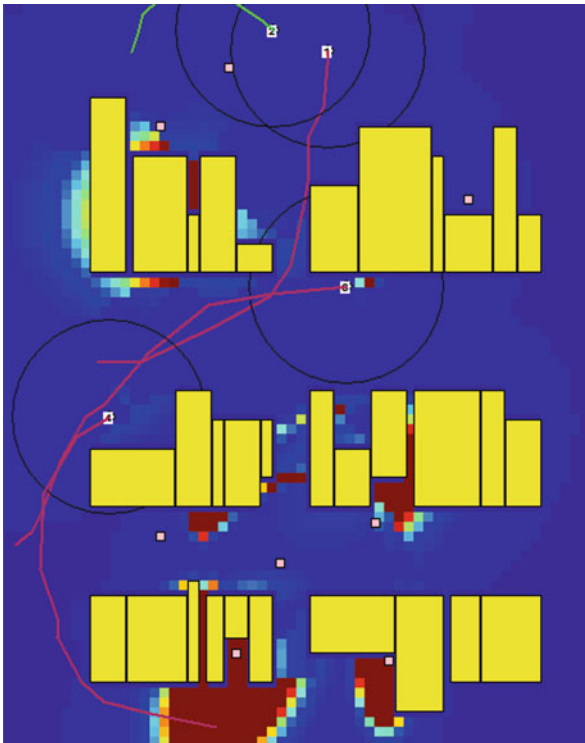


Fig. 64.3 Simulation results for unlimited communications case, at time-step 44. UAVs 1, 3, and 4 are engaged in the search task, while UAV 2 is tracking the target within its sensor coverage (black circle around UAV)

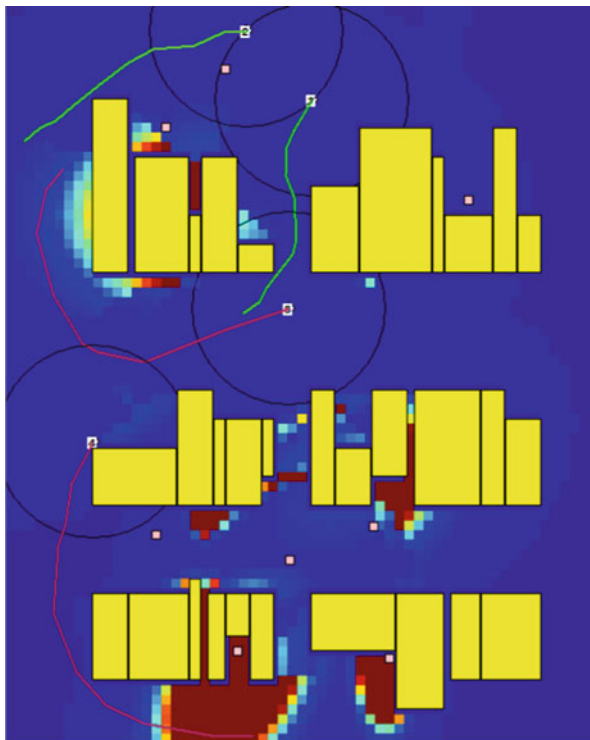


Fig. 64.4 Simulation results for unlimited communications case, at time-step 45. UAVs 3 and 4 are engaged in the search task, while UAVs 1 and 2 are tracking the target within their sensor coverage

with the yellow rectangles representing the buildings, the small white rectangles representing the targets, the UAVs numbered, and the planned routes of the UAVs emanating from each UAV. UAVs engaged in a search task have routes colored red, while those engaged in a tracking task have routes colored green. The background represents the search uncertainty map, with blue corresponding to a low level of uncertainty and red a high level of uncertainty. The sensor coverage for each UAV is represented by a black circle about each UAV. At time-step 44, UAVs 1, 3, and 4 are engaged in a search task, while UAV 2 is engaged in tracking the target within its sensor coverage. At this time-step, UAV 1 and 2 are close enough together to share information, so UAV 1 knows about the target UAV 2 is currently tracking. At time-step 45, UAVs 3 and 4 continue searching, while UAV 1 switches to a tracking task, to help track the target UAV 2 has been tracking. Time-step 46 sees UAV 1 with a new trajectory, to better keep the target it is tracking within its sensor coverage. UAV 2 is still engaged in tracking, and UAVs 3 and 4 are still engaged in a search task. Note also that UAV 4 has its route planned to reduce the uncertainty shown in the bottom left region of the scenario.

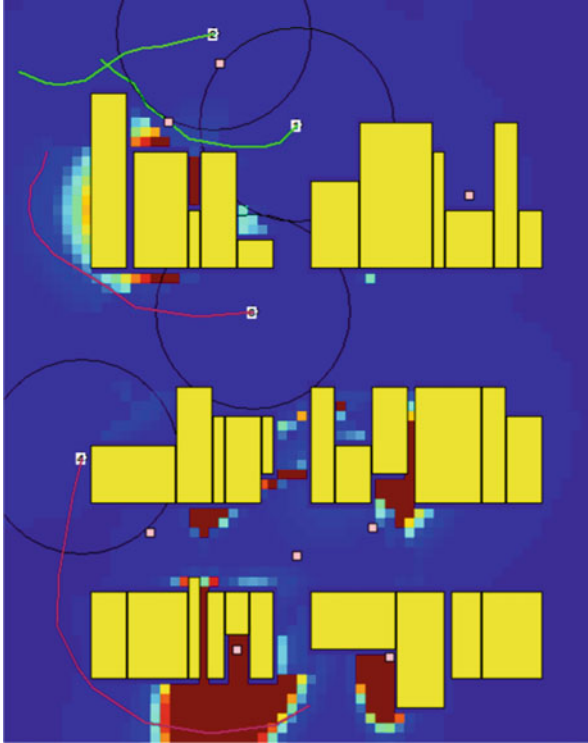


Fig. 64.5 Simulation results for unlimited communications case, at time-step 46. UAVs 3 and 4 are engaged in the search task, while UAV 1 and 2 are continuing to track the target within its sensor coverage. In addition, UAV 1 has changed its trajectory to keep the target within its coverage

One method to measure how the UAV communication range affects the searching algorithm is to observe the minimum search uncertainty value across all UAVs at each location for the simulated region of interest. Figure 64.6 displays the resultant mean minimum search uncertainty map value of the UAVs for the simulated region of interest, at each time-step. The mean search plots tend to show spike and dips. What happens in these cases is that a UAV's search uncertainty map has a group of values that become large in one particular area of the search region. When enough of these values become large, and as long as the UAV is not tasked to track a target, the UAV will eventually plan a route that leads it to search that area of interest. This sudden decrease in many search uncertainty values drives the mean sharply downward. The zero-comms case does not have as many sharp spikes because it has fewer opportunities to switch to tracking (since those UAVs cannot share their state with each other). However, they must conduct their search on their own – this leads to the general upward trend in the zero-comms case in the mean search plot, since they cannot leverage the search information of their neighbors.

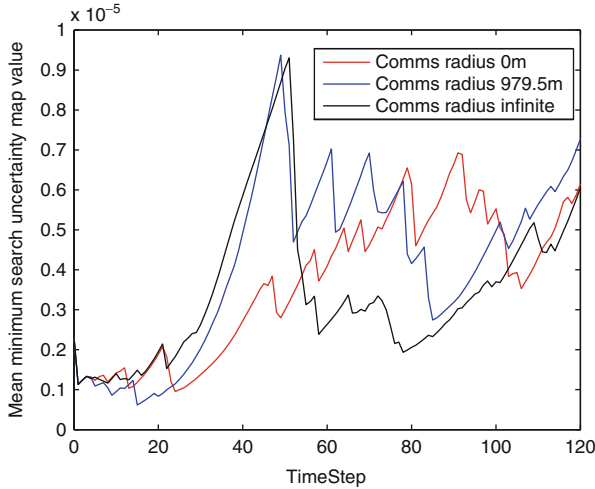


Fig. 64.6 The mean minimum search uncertainty map value over all UAVs at each time-step, for the three UAV communication cases

Figure 64.7 displays the number of UAVs engaged in tracking each target, at each time-step. Of note here is that the mean weighted percentage of targets tracked increases from 48.854 when the communications radius is 0 m to 70.208 when the communication radius is 979.5 m and finally to 80 when the communication radius is infinite. Figure 64.8 displays the mean Mahalanobis distance (Eq. 64.49), between the fused and truthful positions, for each target, for each time-step. As can be seen here, the mean Mahalanobis distance decreases as the communication radius increases. This is because as communication among the UAVs increases, more information on the target kinematics can be shared among the UAVs (and hence fused into each UAVs target map):

$$M_d(x, \bar{x}, \Sigma, \bar{\Sigma}) = (x - \bar{x}) (\Sigma + \bar{\Sigma})^{-1} (x - \bar{x})^T \tag{64.49}$$

As can be seen in the result plots, a push-and-pull effect is occurring. During the search phase, the UAVs want to fly to areas of high uncertainty (and hence reduce the uncertainty there), but as the UAVs detect targets in these high uncertainty regions, they switch their tasking to tracking (because the initial uncertainty of a target contributes more to the objective function than does the search uncertainty). This can be partially alleviated by modifying the decision variable b_i to include a time dimension, $b_i(t)$. This would have the effect of allowing each UAV to switch between tasks over the planning horizon, rather than forcing each UAV to make a decision on a task for the whole planning horizon. This can also be achieved by artificially reducing the size of the planning horizon. Both approaches will increase the complexity and computational throughput and are out of the current scope.

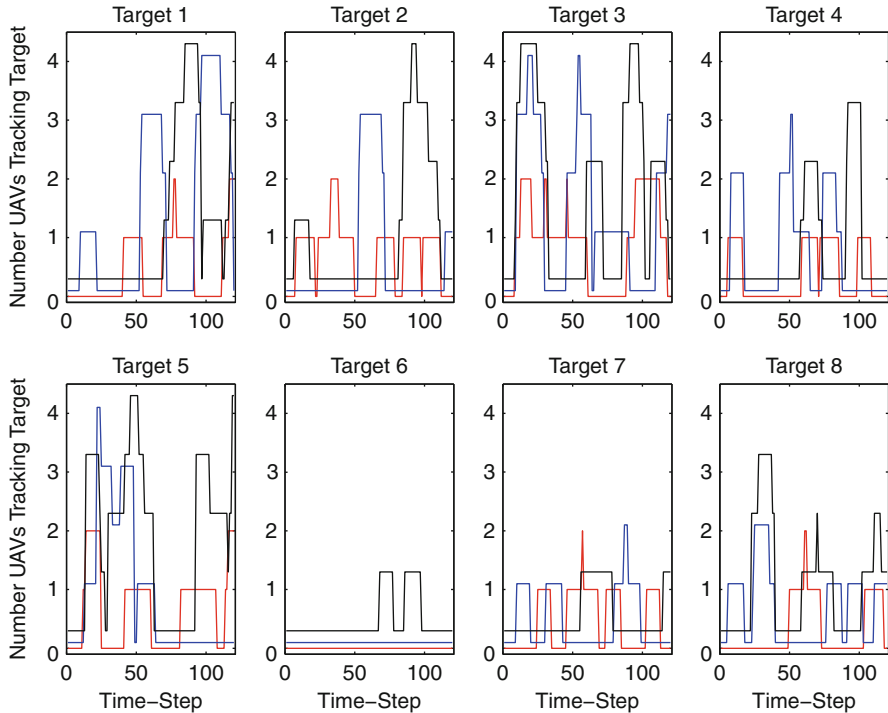


Fig. 64.7 Number of UAVs engaged in tracking each target, at each time-step. The *red curve* denotes the communications radius of 0 m, the *blue curve* denotes the communications radius of 979.5 m, and the *black curve* denotes the infinite communications radius

64.5 Conclusions and Future Research Areas

This chapter considered dynamically updating the tasking and flight paths for UAVs in an urban domain. The UAVs were assumed to be Dubins-like vehicles (i.e., point masses with kinematic constraints) with limited communication capabilities. This resulted in a decentralized approach where each UAV dynamically updated flight plans for itself and its neighbors. As it was an urban domain, the UAVs needed to incorporate obstacle avoidance constraints into the flight plan generation as well as taking into account anticipated future line of sight obstructions to the targets and search region. A dynamic feedback control loop was implemented for each UAV, and a rigorous mathematical formulation was presented for the coupled task assignment/flight path optimization problem each UAV had to solve at each time-step.

This approach was successfully applied to a simulated scenario utilizing a hybrid GRASP-SA heuristic. The influence of communication radius among the UAVs on the resultant objective function value was studied. While it is in general expected

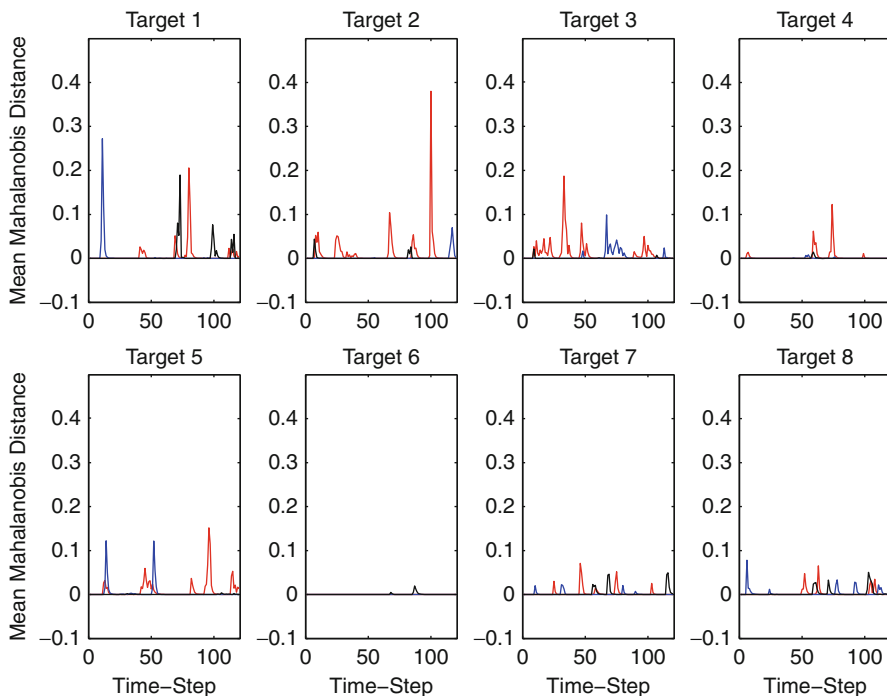


Fig. 64.8 Mean Mahalanobis distance (between the fused position and the truthful position) for each target, for each time-step. The *red curve* denotes the communications radius of 0 m, the *blue curve* denotes the communications radius of 979.5 m, and the *black curve* denotes the infinite communications radius

that the larger the communication radius, the better the performance of the UAVs, as seen from the results, many other factors also influence performance.

It is expected that UAV utilization will continue to increase in scope, both for military and nonmilitary operations. Hence, there is the need to continue to advance the state of the art in autonomous UAV research. Future research directions include applying our general approach to UAVs with heterogeneous sensors (e.g., directional radars, passive sensors) as well as to heterogeneous vehicles, from different domains (e.g., UAVs and UGVs working synergistically). A comparison of the efficiency of our approach, from a computational and functional perspective, to other centralized and decentralized approaches is currently underway. The end goal is geared toward testing these algorithms on real UAVs.

References

- A. Ahmadzadeh, G. Buchman, P. Cheng, A. Jadbabaie, J. Keller, V. Kumar, G. Pappas, Cooperative control of UAVs for search and coverage, in *Proceedings of the AUVSI Conference on Unmanned Systems*, (Orlando, Florida, 2006)

- D. Akselrod, A. Sinha, T. Kirubarajan, Collaborative distributed sensor management to multitarget tracking using hierarchical Markov decision processes, in *Proceedings of SPIE: Signal and Data Processing of Small Targets*, San Diego, vol. 6699, ed. by O.E. Drummond, R.D. Teichgraber (SPIE, Bellingham, 2007), pp. 1–14
- R.J. Bamburger, D.P. Watson, D.H. Scheidt, K.L. Moore, Flight demonstrations of unmanned aerial vehicle swarming concepts. *Johns Hopkins APL Tech. Dig.* **27**(1), 41–55 (2006)
- A. Bicchi, G. Casalino, C. Santilli, Planning shortest bounded-curvature paths for a class of nonholonomic vehicles among obstacles, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Nagoya, 1995, pp. 1349–1354
- S. Blackman, R. Popoli, *Design and Analysis of Modern Tracking Systems* (Artech House, Boston, 1999)
- E.A. Doucette, A.J. Sinclair, D.E. Jeffcoat, Simultaneous localization and planning for cooperative air munitions via dynamic programming, in *Optimization and Cooperative Control Strategies*, ed. by M.J. Hirsch, C.W. Commander, P.M. Pardalos, R. Murphey (Springer, Berlin/London, 2009), pp. 69–79
- L.E. Dubins, On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *Am. J. Math.* **79**(3), 497–516 (1957)
- T.A. Feo, M.G.C. Resende, A probabilistic heuristic for a computationally difficult set covering problem. *Oper. Res. Lett.* **8**, 67–71 (1989)
- T.A. Feo, M.G.C. Resende, Greedy randomized adaptive search procedures. *J. Global Optim.* **6**, 109–133 (1995)
- P. Festa, M.G.C. Resende, GRASP: an annotated bibliography, in *Essays and Surveys in Meta-heuristics*, ed. by C.C. Ribeiro, P. Hansen (Kluwer, Boston, 2002), pp. 325–367
- M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W.H. Freeman, San Francisco, 1979)
- A. Gelb, *Applied Optimal Estimation* (MIT, Cambridge, 1974)
- G. Gu, P.R. Chandler, C.J. Schumacher, A. Sparks, M. Pachter, Optimal cooperative sensing using a team of UAVs. *IEEE Trans. Aerosp. Electron. Syst.* **42**(4), 1446–1458 (2006)
- M.J. Hirsch, H. Ortiz-Pena, Autonomous network connectivity and cooperative control for multiple target tracking, in *Proceedings of the 27th Army Science Conference*, Orlando, 2010
- M.J. Hirsch, H. Ortiz-Pena, N. Sapankevych, R. Neese, Efficient flight formation for tracking of a ground target, in *Proceedings of the National Fire Control Symposium*, San Diego, 2007, pp. 1–16
- M.J. Hirsch, H. Ortiz-Pena, M. Sudit, Decentralized cooperative urban tracking of multiple ground targets by a team of autonomous UAVs, in *Proceedings of the 14th International Conference on Information Fusion*, Chicago, 2011, pp. 1196–1202
- M.J. Hirsch, H. Ortiz-Pena, C. Eck, Cooperative tracking of multiple targets by a team of autonomous UAVs. *Int. J. Oper. Res. Inf. Syst.* **3**(1), 53–73 (2012)
- ILOG CPLEX, <http://www.ilog.com/products/cplex>. Accessed Oct 2008
- K. Jiang, L.D. Seneviratne, S.W.E. Earles, A shortest path based path planning algorithm for nonholonomic mobile robots. *J. Intell. Rob. Syst.* **24**, 347–366 (1999)
- R.J. Kenefic, Finding good Dubins tours for UAVs using particle swarm optimization. *J. Aerosp. Comput. Inf. Commun.* **5**, 47–56 (2008)
- J. Kennedy, R. Eberhart, Particle swarm optimization, in *Proceedings of the IEEE International Conference on Neural Networks*, Australia, 1995, pp. 1942–1948
- D. Kingston, R. Beard, UAV splay state configuration for moving targets in wind, in *Advances in Cooperative Control and Optimization*, ed. by M.J. Hirsch, P.M. Pardalos, R. Murphey, D. Grundel (Springer, Berlin, 2007), pp. 109–129
- S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing. *Science* **220**(4598), 671–680 (1983)
- S.M. LaValle, *Planning Algorithms* (Cambridge University Press, Cambridge, 2006)
- D. Lerro, Y. Bar-Shalom, Tracking with debiased consistent converted measurements versus EKF. *IEEE Trans. Aerosp. Electron. Syst.* **29**(3), 1015–1022 (1993)

- Z. Liang, W.A. Chaovaitwongse, A.D. Rodriguez, D.E. Jeffcoat, D.A. Grundel, J.K. O'Neal, Optimization of spatiotemporal clustering for target tracking from multi-sensor data. *IEEE Trans. Syst. Man Cybern. (C)* **40**(2), 176–188 (2010)
- C.S. Ma, R.H. Miller, Mixed integer linear programming trajectory generation for autonomous nap-of-the-earth flight in a threat environment, in *Proceedings of the IEEE Aerospace Conference, Big Sky*, 2005, pp. 1–9
- M. Matsumoto, T. Nishimura, Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Model. Comput. Simul.* **8**(1), 3–30 (1998)
- P.S. Maybeck, *Stochastic Models, Estimation, and Control*, vol. I (Navtech Press, Arlington, 1994)
- T.W. McLain, P.R. Chandler, M. Pachter, A decomposition strategy for optimal coordination of unmanned air vehicles, in *Proceedings of American Control Conference*, Chicago, 2000, pp. 369–373
- F. Rafi, S. Khan, K. Shafiq, M. Shah, Autonomous target following by unmanned aerial vehicles, in *Proceedings of SPIE: Unmanned Systems Technology VIII*, Orlando, vol. 6230, ed. by G.R. Gerhart, C.M. Shoemaker, D.W. Gage (SPIE, Bellingham, 2006), pp. 1–8
- C. Santilli, A. Bicchi, G. Casalino, A. Balestrino, Nonholonomic, bounded curvature path planning in cluttered environments, in *Proceedings of the Conference on Emerging Technologies and Factory Automation*, Paris, vol. 2, 1995, pp. 363–372
- T. Schouwenaars, B. De Moor, E. Feron, J. How, Mixed-integer programming for multi-vehicle path planning, in *Proceedings of the European Control Conference*, Porto, 2001, pp. 2603–2608
- C. Schumacher, T. Shima, Single-task tours, in *UAV Cooperative Decision and Control*, ed. by T. Shima, S. Rasmussen (SIAM, Philadelphia, 2009), pp. 15–36
- J.G. Semple, G.T. Kneebone, *Algebraic Projective Geometry* (Oxford University Press, New York, 1952)
- V.K. Shetty, M. Sudit, R. Nagi, Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles. *Comput. Oper. Res.* **35**(6), 1813–1828 (2008)
- T. Shima, S. Rasmussen, *UAV Cooperative Decision and Control* (SIAM, Philadelphia, 2009)
- A.J. Sinclair, R.J. Prazenica, D.E. Jeffcoat, Simultaneous localization and planning for cooperative air munitions, in *Advances in Cooperative Control and Optimization*, ed. by M.J. Hirsch, P.M. Pardalos, R. Murphey, D. Grundel (Springer, Berlin, 2007), pp. 81–94
- A. Sinha, T. Kirubarajan, Y. Bar-Shalom, Optimal cooperative placement of UAVs for ground target tracking with doppler radar, in *Proceedings of SPIE: Signal processing, Sensor Fusion, and Target Recognition XIII*, Orlando, vol. 5429, ed. by I. Kadar (SPIE, Bellingham, 2004), pp. 95–104
- A. Sinha, T. Kirubarajan, Y. Bar-Shalom, Autonomous ground target tracking by multiple cooperative UAVs. *Proceedings of the IEEE Aerospace Conference, Big Sky*, 2005, pp. 1–9
- M. Steinberg, Intelligent autonomy for unmanned naval vehicles, in *Proceedings of SPIE: Unmanned Systems Technology VIII*, Kissimmee, vol. 230, ed. by G.R. Gerhart, C.M. Shoemaker, D.W. Gage (SPIE, Bellingham, 2006), pp. 1–12
- J. Stolfi, *Oriented Projective Geometry: A Framework for Geometric Computations* (Elsevier, Palo Alto, California, 1991)
- C.A. Sylvester, G.J. Wiens, N.G. Fitz-Coy, Control of collaborative mobile robots subject to nonholonomic constraints, in *Proceedings of the International Symposium on Collaborative Technologies and Systems*, San Diego, 2004