Experimental Implementation of Spectral Multiscale Coverage and Search Algorithms for Autonomous UAVs

George Mathew, Suresh Kannan, Amit Surana, Sanjay Bajekal and Konda R. Chevva*

We discuss the implementation of Spectral Multiscale Coverage (SMC) based multi-vehicle control and coordination for coverage and search missions by autonomous UAVs. The SMC algorithm gives rise to multi-scale vehicle trajectories leading to efficient coverage of a given area and thereby making it useful for search algorithms that are robust to sensor errors and terrain uncertainties. We provide a functional summary of the SMC framework and address its practical implementation. The practical feasibility of the SMC approach is demonstrated via several coverage problems using high fidelity software-in-the-loop (SIL) simulations and experimental flight tests conducted using an electric helicopter.

I. INTRODUCTION

A variety of coverage problems arise in applications involving autonomous vehicles. A few such applications where coverage control comes into play are search and tracking missions, exploration and environmental monitoring. The problem of searching a large area to find an unknown number of targets is challenging especially when the field of view of one sensor is much smaller than the entire search area and when there is uncertainty in terrain and sensor measurements. This path planning problem has been addressed by several research groups and the various solution approaches are briefly discussed below.

There are various types of coverage problems. The first type of coverage problem is that of locational optimization and algorithms that solve the locational optimization problem are referred to as *static coverage algorithms*. Static coverage algorithms are relevant when the collective sensor footprint is comparable to the domain size and the problem is to place the sensors in an optimal configuration so as to maximize the detection probability of some event. Static coverage problems are typically addressed using Voronoi partition based algorithms (e.g. Lloyd algorithm) and their numerous variants. See Ref. 1 for a review on static coverage algorithms and the distributed and asynchronous implementations of the Lloyd algorithm for mobile sensing networks.

The focus of this paper is on *dynamic coverage* problems and also their practical implementation for UAVs. Dynamic coverage problems refer to situations where the collective sensor footprint is small compared to the size of the environment and the motion of the sensors must be designed so that they visit or come close to visiting every point in the environment. This path planning problem has been addressed by both computer scientists and control theorists. For a review of results from the robotics community see Ref. 2 where the author describes both heuristic and complete algorithms based on cellular decompositions to ensure complete coverage of a domain. For control theoretical approaches to coverage problems, see Ref. 3 and our recent work in Refs. 4 and 5.

A special type of dynamic coverage problem arises when the environment to be covered changes with time. Examples are when sensors have to move to cover a moving domain or search for a stationary or mobile target. For the search of a target, the uncertain position of the target may be specified in terms of a probability distribution that evolves in time according to the sensor observations and target dynamics. Design of uniform coverage dynamics for sensors in search missions is crucial to minimize the average time to detect a target. A variety of researchers have addressed the problem of trajectory design for search missions. For an information-driven framework for coordinated control of a network of unmanned vehicles, see Ref. 6. In Ref. 7, the authors describe a model predictive approach that optimizes the routes of agents searching for a mobile target. Refs. 8 and 9 describe frameworks for cooperative search using UAV teams.

*G. Mathew is with United Technologies Research Center (UTRC Inc.), Berkeley, CA, USA. S. Kannan is with NodeIN LLC, Burlington, CT, USA. A. Surana, S. Bajekal and K. R. Chevva are with United Technologies Research Center (UTRC), East Hartford, CT, USA. mathewga@utrc.utc.com, kannan@nodein.com, suranaa@utrc.utc.com, bajekas@utrc.utc.com, chevvakr@utrc.utc.com

In this paper, we describe and implement a Spectral Multiscale Coverage (SMC) framework for multi-vehicle control and coordination for search and tracking applications. This approach differs from other related approaches in that it is based on the notion of 'uniform coverage'. By 'uniform coverage', we roughly mean that the sensor footprints are uniformly distributed or evenly spaced throughout the domain. In a classic book on the theory of search,[10] the author describes the advantages and difficulty of generating a sensor track that is uniformly distributed. To the best of our knowledge, there is no other framework that uses uniformly distributed sensor tracks to perform search missions - which is the subject of this paper.

The multiscale coverage metric that quantifies the uniformity of coverage is described in Refs. 4 and 5. The SMC algorithms leads to sensor dynamics so that large scale features are detected first followed by smaller and smaller features. This leads to uniform coverage dynamics for the mobile sensors such that the amount of time spent observing a region is proportional to the probability of finding a target in it. It has been demonstrated in Ref. 11 that in the presence of uncertainty in terrain and sensor observations, search strategies based on the SMC algorithm outperform lawnmower-type search strategies by a factor of 2. For moving targets, the probability distribution specifying the uncertainty in target state evolves in time according to the target dynamics. An extension of the SMC algorithm - Dynamic Spectral Multiscale Coverage (DSMC) - has been developed for the search and tracking of mobile targets.[12]

In our initial studies, the sensor dynamics was restricted to that of a point mass model. We recently extended the SMC approach for a Dubins vehicle model[13] which has been found to be an adequate abstraction for motion planning problems involving unmanned vehicles governed by complicated and constrained dynamics.[14,15] The work in Ref. 13 also describes an adaptive search methodology based on combining SMC control with estimation/decision theoretic methods, namely: Sequential Probability Ratio Test (SPRT) and Recursive Least Square estimation. This renders an efficient method for searching for an unknown number of stationary targets in a manner that is is robust to sensor and terrain uncertainties, and Automatic Target Recognition (ATR) algorithm errors (i.e. false alarm, missed detections).

The objective of this paper is to present the experimental implementation of the SMC approach described in Ref. 13. We review the SMC based multi-vehicle control and coordination framework, and address some of the issues related to the experimental implementation. Specifically, we show how the SMC control can be coupled with a commercially available off the shelf (COTS) autopilot. We also conduct a detailed software-in-the-loop (SIL) assessment of the SMC framework using a high fidelity simulation environment. We also present results from experimental flight tests with an electric helicopter showing the practical feasibility of the SMC based coverage and search.

The rest of the paper is structured as follows. In Section II, we summarize the SMC feedback control laws for different vehicle abstractions including the Dubins model. We also review the SMC based adaptive search methodology which accounts for sensor and ATR errors in target detection. In Section III, we describe the details of implementation of the coverage control on a commercially available off the shelf autopilot, and also present a boundary control to keep the vehicle within a prescribed domain. Section IV provides details of software-in-the-loop (SIL) and experimental implementations. We also outline the different test scenarios. In Section V we present the SIL and experimental flight test results for a Maxi Joker helicopter, demonstrating the practical feasibility of the Spectral Multiscale Coverage framework. We give some concluding remarks in Section VI, and point to some future research directions.

II. Spectral Multiscale Coverage (SMC): An Overview

In this section, we review the SMC framework that is discussed with more detail in Refs. 5, 12 and 13.

II.A. Coverage Metric

The SMC algorithm was initially developed for designing uniform coverage dynamics for the search of stationary targets in complex terrains.[4,5] A key component of the SMC algorithm is a metric for uniformity of coverage which was motivated by ergodic dynamical systems theory. Consider N sensors searching a rectangular region $U \subset \mathbf{R}^2$ with a given search prior $\mu(\mathbf{x})$. The prior represents the probability of finding a target at a given location $\mathbf{x} \in U$. Let the locations of the sensors at time t be given by $\mathbf{z}^j(t) \subset \mathbf{R}^2, j = 1, \cdots, N$. The sensor locations evolve according to some controlled dynamic model. To keep track of the points that the sensors have already visited, we use a *coverage distribution* generated by the sensor trajectories upto time t, which is defined as:

$$C_t(\mathbf{x}) \quad = \quad \frac{1}{Nt} \sum_{j=1}^{N} \int_0^t \delta(\mathbf{x} - \mathbf{z}^j(\tau)) d\tau, \tag{1}$$

where it has been assumed that sensors have an infinitesimal footprint represented by the Dirac delta distribution δ (This can be replaced by an arbitrary footprint). The coverage metric (given by a Sobolev space norm of negative

index) can be expressed as

$$\phi^2(t) = ||C_t(.) - \mu(.)||^2_{H^{-3/2}} = \sum_K \Lambda_k |s_k(t)|^2, \tag{2}$$

where,

$$s_k(t) \quad = \quad c_k(t) - \mu_k, \quad \Lambda_k = \frac{1}{(1+||k||^2)^{3/2}}, \tag{3}$$

and where c_k and μ_k are the Fourier coefficients of C_t and $\mu(.)$. i.e

$$c_k(t) \quad = \quad < C_t, f_k >= \frac{1}{Nt} \sum_{j=1}^{N} \int_0^t f_k(\mathbf{z}^j(\tau)) d\tau, \tag{4}$$

$$\mu_k \quad = \quad < \mu, f_k >, \tag{5}$$

with f_k being the Fourier basis functions that satisfy Neumann boundary conditions on the domain U, k is the wave-number vector and $< \cdot, \cdot >$ is the standard inner product (for details, see Ref. 5). $\phi(t)$ quantifies how much the time averages of the Fourier basis functions along the sensor trajectories deviate from their spatial averages, but giving more importance to large-scale modes than the small-scale modes.

II.B. SMC Control

Given a model for the sensor dynamics, we use a receding horizon approach to maximize the rate of decay of the coverage metric. The control action at any time t is obtained by solving a finite horizon optimal control problem over a short time horizon $[t, t+\Delta t]$ with the objective of attaining the highest rate of decay of the coverage metric (2) at the end of the horizon. The feedback law is derived in the limit as the size of the horizon Δt goes to zero and the resulting feedback control is applied at every time instant t. The cost functional over $[t, t+\Delta t]$ can be expressed as:

$$\mathscr{C}(t, \Delta t) = \dot{\Phi}(t+\Delta t) = \sum_K \Lambda_k S_k(t+\Delta t) W_k(t+\Delta t), \tag{6}$$

where,

$$\Phi(t) \quad = \quad \frac{1}{2} \sum_k \Lambda_k |S_k(t)|^2 = \frac{1}{2} N^2 t^2 \phi^2(t), \tag{7}$$

$$S_k(t) \quad = \quad (c_k(t) - \mu_k) N t, \tag{8}$$

$$W_k(t) \quad = \quad \dot{S}_k(t) = \sum_{j=1}^{N} f_k(\mathbf{z}^j(t)) - N\mu_k. \tag{9}$$

We review SMC control laws for different vehicle abstractions - namely point mass, first order and second order Dubins vehicle models. Trajectories generated based on a point mass model can have infinite turn rates and therefore is not suitable for practical applications as most unmanned vehicles have turn rate constraints. On the other hand, a Dubins vehicle is a vehicle moving with bounded turn rate on a plane[16] and is considered an adequate model (see Refs. 14, 15 and references therein) from the perspective of path planning for unmanned vehicles.

II.B.1. Point Mass Model

First order dynamics for a point mass model is described by

$$\dot{\mathbf{z}}^j = \mathbf{u}^j(t), \quad j = 1, \cdots, N. \tag{10}$$

A closed form control law $\mathbf{u}^j(t)$ at time t that minimizes (6), subject to the speed constraint,

$$||\mathbf{u}^j|| \leq v_{max}, \tag{11}$$

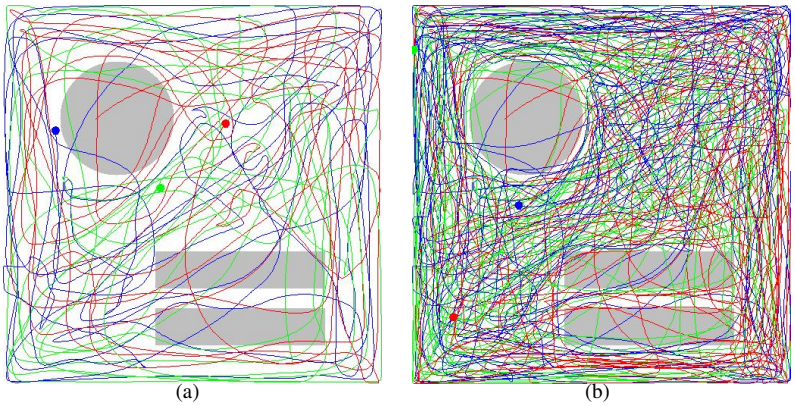
can be derived in the limit as $\Delta t \to 0$. This control law is given as

$$\mathbf{u}^j(t) = -v_{max} \frac{\mathbf{B}^j(t)}{||\mathbf{B}^j(t)||_2}, \tag{12}$$

where,

$$\mathbf{B}^j(t) = \begin{pmatrix} B_x^j(t) \\ B_y^j(t) \end{pmatrix} = \sum_K \Lambda_k S_k(t) \nabla_{\mathbf{x}} f_k(\mathbf{z}^j(t)). \tag{13}$$

Figure 1 shows samples trajectories generated for a point mass model with the SMC control (12).



(a) (b)

Figure 1. Sensor trajectories generated by the SMC algorithm for a point mass model and for a uniform prior on an irregular domain. The coverage control gives rise to multiscale sensor trajectories. The figure on the left is a snapshot of the trajectories at a time earlier than that for the figure on the right. As seen, the spacing between trajectories decreases as time proceeds. These figures are taken from Ref. 5.

II.B.2. First Order Dubins Model

Consider a first order Dubins vehicle model, whose dynamics is restricted to a plane and is governed by

$$\begin{aligned} \dot{\mathbf{z}}^j &= v^j(t) e^{i\theta^j} \\ \dot{\theta}^j &= \omega^j(t), \end{aligned} \tag{14}$$

where $e^{i\theta} = (\cos(\theta), \sin(\theta))'$ denotes the unit vector in the direction of the vehicle heading, v^j is the speed, θ^j is the yaw angle and ω^j is the yaw rate. For simplicity, we assume that all vehicles are subject to the same speed and turn rate constraints. i.e.,

$$v_{min} \leq v^j \leq v_{max}, \quad \omega_{min} \leq \omega^j \leq \omega_{max}. \tag{15}$$

The optimal control law takes the classical bang-bang form and in the limit as the horizon $\Delta t \to 0$, the control law is given as:

$$v^j(t) = \begin{cases} v_{min} & \text{if} \quad \Gamma_v^j(t) \geq 0 \\ v_{max} & \text{otherwise,} \end{cases} \tag{16}$$

and

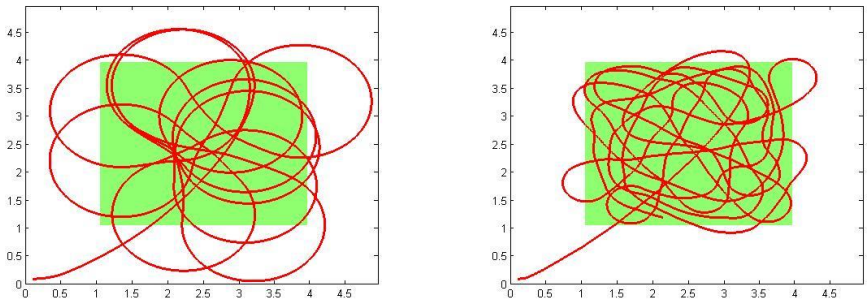
$$\omega^j(t) = \begin{cases} \omega_{min} & \text{if} \quad \Gamma_\omega^j(t) \geq 0 \\ \omega_{max} & \text{otherwise,} \end{cases} \tag{17}$$

where,

$$\Gamma_v^j(t) = B_x^j(t) \cos\theta^j(t) + B_y^j(t) \sin\theta^j(t) \tag{18}$$

$$\Gamma_\omega^j(t) = -B_x^j(t) \sin\theta^j(t) + B_y^j(t) \cos\theta^j(t), \tag{19}$$

and (B_x^j, B_y^j) is as defined before in (13). Figure 2 shows uniform coverage trajectories generated for the Dubins vehicle model with different bounds on the turn-rates. The plot on the left is that for a low turn-rate (high turn radius) and the plot on the right is that for a high turn-rate (low turn radius). Note that due to the bang-bang nature of the SMC control, the speed and angular rate commands would be discontinuous and hence difficult for a physical vehicle to follow. However, the position and yaw variables are continuous and one could issue them as commands to the vehicle autopilot. To avoid the discontinuities in the speed and angular rates, one can consider a second order Dubins model as described next.



(a) Low turn-rate (b) High turn-rate

Figure 2. Uniform coverage trajectories generated for Dubins vehicle model with different turn-rates. The search prior μ is constant within the shaded region and zero outside.

II.B.3. Second Order Dubins Model

A second order Dubins vehicle model includes speed and angular velocity dynamics as shown below:

$$\begin{aligned} \dot{\mathbf{z}}^j &= v^j(t)e^{i\theta^j} \\ \dot{v}^j &= F^j(t) \\ \dot{\theta}^j &= \omega^j(t) \\ \dot{\omega}^j &= T^j(t). \end{aligned} \tag{20}$$

In addition to the constraints in (15) we have thrust and torque constraints. i.e.,

$$F_{min} \leq F^j \leq F_{max}, \quad T_{min} \leq T^j \leq T_{max}. \tag{21}$$

We define the following constants

$$\begin{aligned} v_a &= \frac{v_{min}+v_{max}}{2}, & v_d &= \frac{v_{max}-v_{min}}{2}, \\ \omega_a &= \frac{\omega_{min}+\omega_{max}}{2}, & \omega_d &= \frac{\omega_{max}-\omega_{min}}{2}, \end{aligned}$$

and then introduce the variables α^j and β^j to eliminate the constraints on the speed and yaw rate. The vehicle dynamics is rewritten as

$$\begin{aligned} \dot{\mathbf{z}}^j &= v^j(t)e^{i\theta^j(t)} \\ \dot{v}^j &= v_d\cos(\alpha^j(t))\bar{F}^j(t) \\ \dot{\alpha}^j &= \bar{F}^j(t) \\ \dot{\theta}^j &= \omega^j(t) \\ \dot{\omega}^j &= \omega_d\cos(\beta^j(t))\bar{T}^j(t) \\ \dot{\beta}^j &= \bar{T}^j(t), \end{aligned} \tag{22}$$

with initial conditions for α^j and β^j being

$$\alpha^j(0) = sin^{-1}\left(\frac{v^j(0)-v_a}{v_d}\right), \beta^j(0) = sin^{-1}\left(\frac{\omega^j(0)-\omega_a}{\omega_d}\right).$$

For this model, the cost functional over the horizon $[t, t+\Delta t]$ that we optimize for is written as:

$$\begin{aligned} \mathscr{C}(t,\Delta t) = &\sum_{j=1}^{N}\frac{c_v}{2}\left[v^j(t+\Delta t)\right]^2 \\ &+\sum_{j=1}^{N}\int_t^{t+\Delta t}\frac{c_\omega}{2}\left[\omega^j(\tau)\right]^2 d\tau + \ddot{\Phi}(t+\Delta t), \end{aligned} \tag{23}$$

where $c_v, c_\omega > 0$ are damping coefficients that penalize the magnitudes of the speed and turn-rates respectively. The coverage control law in the limit as $\Delta t \to 0$ (as described in Ref. 13) takes the form:

$$\bar{F}^j(t) = \begin{cases} \bar{F}_{min} & \text{if} \quad \Gamma_v^j(t) \geq 0 \\ \bar{F}_{max} & \text{otherwise,} \end{cases} \tag{24}$$

and

$$\bar{T}^j(t) = \begin{cases} \bar{T}_{min} & \text{if} \quad \Gamma_\omega^j(t) \geq 0 \\ \bar{T}_{max} & \text{otherwise,} \end{cases} \tag{25}$$

where

$$\Gamma_v^j(t) = [c_v v^j + B_x^j \cos(\theta)^j) + B_y^j \sin(\theta^j)] \cos(\alpha^j), \tag{26}$$

$$\Gamma_\omega^j(t) = [c_\omega \omega^j - v^j B_x^j \sin(\theta^j) + v^j B_y^j \cos(\theta^j)] \cos(\beta^j) \tag{27}$$

In the above equations, $\bar{F}_{min} = F_{min}/v_d, \bar{F}_{max} = F_{max}/v_d, \bar{T}_{min} = T_{min}/\omega_d$ and $\bar{T}_{max} = T_{max}/\omega_d$.

II.C. SMC based Adaptive Search for Stationary Targets

In this section we summarize the SMC based adaptive search algorithm for search of an unknown number of targets in a region of interest using a team of mobile sensors. In order to make the search robust to sensor uncertainties and Automatic Target Detection algorithm errors (i.e. false alarm, missed detections), the SMC control law derived in the previous section was combined with decision and estimation theoretic techniques.[13] As new targets are discovered, the Sequential Ratio Probability Test (SPRT), Recursive Least Squares (RLS) estimation and Bayesian updates are used to quantify the current uncertainty in target detection, location and classification (as adversarial/non adversarial), respectively. Sensors should spend more time in locations where there is higher probability in ascertaining the presence/absense of a target, and where there is associated higher uncertainty in their location and classification. This uncertainty is used to update the search prior so as to balance exploitation (to reduce uncertainty in state of already discovered potential targets) and exploration (to discover new targets). More precisely, let μ^0 denote the initial search prior. We assume that the uncertainty in the location of a discovered target can be represented by a Gaussian distribution. When a targets falls within the sensor range, a random sensor measurement is generated and the search prior is adapted as

$$\mu_t(\mathbf{x}) \quad = \quad w_t^0 \mu^0(\mathbf{x}) + \sum_{i=1}^{n_t} w_t^i \mu_t^i(\mathbf{x}), \tag{28}$$

where n_t is the number of targets that have been detected so far and

$$\mu_t^i(\mathbf{x}) = \mathscr{G}(\mathbf{x}; \hat{\mathbf{x}}_t^i, P_t^i), \tag{29}$$

where $\mathscr{G}(x; \bar{x}, \Sigma)$ is a multivariate Gaussian with mean $\bar{x}$ and covariance matrix Σ. Here, $\mathbf{x}_t^i$ is the target location estimate with covariance P_t^i at time t, which is computed using the Recursive Least Squares estimator. The weights w_t^i represent target prioritization, and are chosen as

$$w_t^i = \text{tr}(P_t^i) \times \left(1 + \mathscr{H}(p_t^i)\right) \left(1 + \mathscr{H}(\pi_t^i)\right), \tag{30}$$

where $\mathscr{H}(p) = -p \log p - (1-p)\log(1-p)$ is the entropy, π_t^i is target detection probability (computed based on SPRT) and p_t^i is target classification probability (computed based on Bayesian updates). The weights w_t^i are normalized so that they add up to one at all times. Additionally, one can apply appropriate thresholds for each term above, so that the target weight is set to zero, once the uncertainty in its state falls below these thresholds. The weight $w_t^0 \geq 0$ can be used to control the tradeoff between exploration and exploitation.

III. Practical Considerations

III.A. Boundary Control

In practical implementations, due to the vehicle dynamic constraints (specifically the turning radius constraint), the vehicle is not guaranteed to remain within the prescribed region U under the SMC control described in the previous

section. Therefore, to bring back the vehicle into the prescribed search region whenever it crosses the boundary, we use the same controls as in (24) and (25), but with the variables $\Gamma_v^j(t)$ and $\Gamma_\omega^j(t)$ redefined as

$$\Gamma_v^j(t) = [c_v v^j + (z^j - z^{mid}) \cdot e^{i\theta^j}] \cos(\alpha^j), \tag{31}$$

$$\Gamma_\omega^j(t) = [c_\omega \omega^j + (z^j - z^{mid}) \cdot ie^{i\theta^j}] \cos(\beta^j), \tag{32}$$

where z^{mid} is the mid-point of the domain U. This control is derived using a cost-function similar to that in (23), but which penalizes the distance from the mid-point instead of the rate of decay of the coverage metric.

III.B. Implementation with an Autopilot

In this section we describe the implementation of the coverage control described in the previous sections on a commercially available off the shelf (COTS) autopilot. In our setting, we had the capability to command a time parameterized trajectory to the autopilot. Let the vehicle state be represented as $\mathbf{S}(t) = (X, Y, \theta, \dot{X}, \dot{Y}, \dot{\theta}, Z, \psi, \phi, \dot{Z}, \dot{\psi}, \dot{\phi})'$, where we note that we have first indicated the states which are present in the Dubins model and then the remaining ones $(Z, \psi, \phi, \dot{Z}, \dot{\psi}, \dot{\phi})'$ which correspond to altitude, pitch, roll and their rates respectively. We assume that the vehicle state estimate $\hat{\mathbf{S}}(t)$ is available as feedback from the autopilot. Using a subset of the states $(\hat{X}(t), \hat{Y}(t), \hat{\theta}, \hat{\dot{X}}, \hat{\dot{Y}}, \hat{\dot{\theta}})'$ from the estimate $\hat{\mathbf{S}}(t)$, the Dubins model (22) is initialized and the Dubins vehicle trajectory is computed under the SMC control (24) and (25) over a desired period of time $[t, t+T]$. Let $\mathbf{S}_d(t) = (\mathbf{z}(t), \theta(t), v\cos(\theta(t)), v\sin(\theta(t)), \dot{\theta}(t))', t \in [0, T]$ be the Dubins vehicle trajectory, then the full state trajectory is constructed as:

$$\mathbf{S}(\tau) = \begin{pmatrix} \mathbf{S}_d(\tau - t) \\ Z_d \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \qquad \tau \in [t, t+T], \tag{33}$$

which is commanded to the autopilot. Note that here we have assumed that the vehicles operate at a desired constant altitude Z_d which can be taken to be different for different vehicles for collision avoidance. Also the commands for the pitch, roll and their rates are set to be zero.

IV. SIMULATION AND EXPERIMENTAL SETUP

In this section we describe the setup for the SIL and flight tests. In all our tests, we consider one Maxi-joker with a simulated camera in a terrain with virtual targets. We assume that the simulated visible spectrum camera is gimbal stabilized and always looks downward despite vehicle roll/pitch. Also, the vehicle flies over obstacles, and hence there is no issue of obstacle avoidance.

The flight tests were conducted at a baseball field approximately 450 ft long and 350 ft wide, and surrounded by a canopy. Fig. 3 shows an aerial photo of the experimental test site. The terrain map for the same site was used in the SIL tests. For the convenience of the safety pilot, the search region was restricted to a 200 ft $\times$ 200 ft square region similar to that shown by the green box in Figure 3. The vehicle was brought back inside the search region using the boundary control as described in III.A whenever the vehicle exited the search region.

IV..1. Test Cases

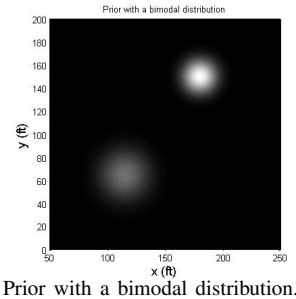
We list the different scenarios used for the SIL and flight tests. These cases represent some key canonical scenarios encountered in practice.

a) *Coverage:* We test the SMC algorithm for the following priors.

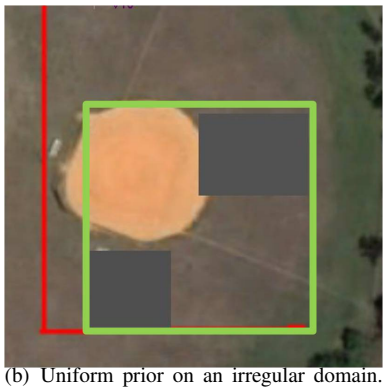
   i) Uniform prior: In this scenario, the search prior μ is uniform within the search region shown in Figure 3. This is useful when there is no prior information about the target locations.

   ii) Prior with a bimodal distribution: Here the search prior μ is an equally weighted sum of two Gaussian distributions with different peak magnitudes and spread as shown in Fig. 4a. In this example, the vehicle should spend roughly equal time close to the peaks of the two Gaussian distributions.

Figure 3. Test site used in SIL and experimental flight tests.



(a) Prior with a bimodal distribution. The prior is the average of two Gaussian distributions restricted to the green square box shown in Figure 3.



(b) Uniform prior on an irregular domain. The prior is zero within the gray shaded regions and constant outside.

Figure 4. Priors used in coverage test scenarios.

iii) Uniform prior on a non-convex domain: As seen in Figure 4b, the prior is set to zero in the gray areas which may represent buildings or foliage regions, and set to a constant non-zero value outside the gray areas within the green box.

b) *Search of stationary targets:* In this experiment, we test the adaptive search algorithm as described in II.C with virtual targets. Here, one starts with an initial prior, and the prior is adapted as new targets are detected. Target locations are drawn randomly with a uniform distribution.

We first describe the software-in-the-loop implementation. The results of these simulation studies were used to guide the selection of the algorithm parameters used in the experimental validation described in Section V.B.

IV.A. Software-in-Loop (SIL) Implementation

The SIL layout is as depicted in the lower box in Fig. 5. The Georgia Tech UAV Simulation Tool (GUST) simulation environment was used to simulate helicopter dynamics and associated navigation filters and trajectory tracking flight controller.[17,18] GUST was used in closed-loop with the SMC trajectory generation algorithm to replicate as many aspects of the actual flight test as possible. The helicopter simulator is a detailed closed-loop helicopter model that includes aerodynamics, actuator models, implemented flight controller and associated modes, software interfaces and communication delays. Trajectory commands generated by the SMC planner were transmitted via UDP packets to multiple instances of the simulator. The actual vehicle state, returned by the simulator, is fed back into the planner in order for it to adapt and update the trajectory command at regular intervals (at the replanning frequency).

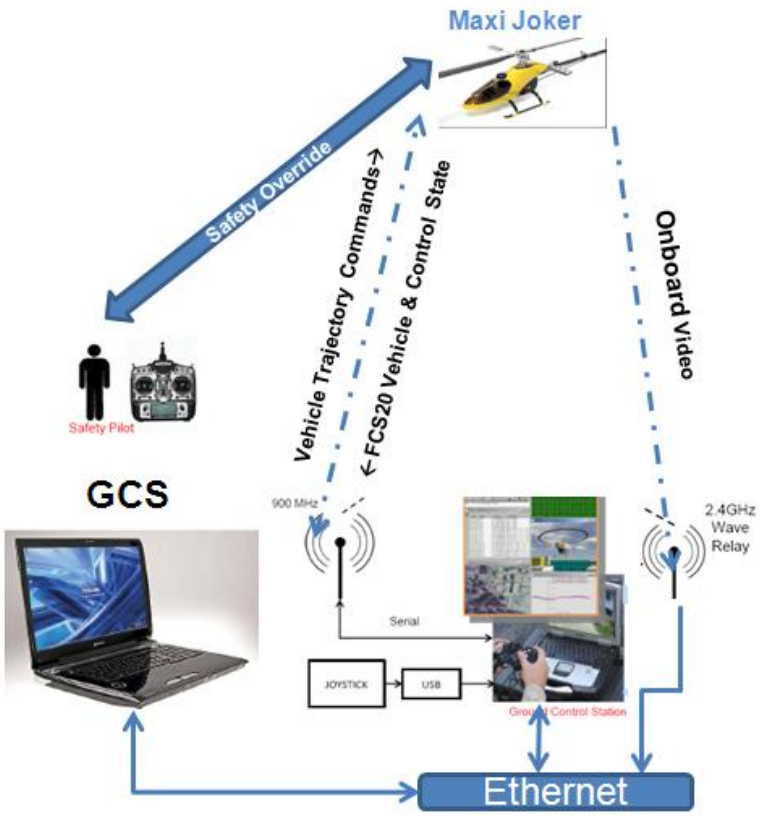


Figure 5. Experimental setup. Also shown in the box is the SIL setup.

TEST INPUT DATA Each SIL test run requires the search prior over the terrain map, the parameters for the Dubins vehicle model used to generate trajectory commands, and a number of parameters that determine the behavior of the SMC algorithm. Below we list these parameters along with their nominal values.

a) **Maximum Flyable Time**: $T_f = 10$ min and was limited by battery life.

b) **Parameters for Dubins vehicle model**:

-Max/Min Speed: $v_{min} = 5$ft/s and $v_{max} = 10$ft/s.

-Max/Min Yaw Rate: $\omega_{min} = -0.3$ rad/s and $\omega_{max} = 0.3$rad/sec.

-Max/Min Linear Acceleration: $F_{min} = -7\text{ft}/s^2$ and $F_{max} = 7\text{ft}/s^2$.

-Max/Min Angular Acceleration: $T_{min} = -0.5$ rad/s^2 and $T_{max} = 0.5\text{rad}/s^2$.

-Damping Coefficients: $c_v = 0.01$ and $c_\omega = 0.25$.

c) **Search Prior**: μ was defined on a uniform grid of size $N_x \times N_y$, so that the spatial discretization is

$$\Delta x = \frac{X_{max} - X_{min}}{N_x}, \qquad \Delta y = \frac{Y_{max} - Y_{min}}{N_y}. \tag{34}$$

The same grid would be used to represent the coverage distribution $C_t(.)$ and to compute its Fourier coefficients. For the SIL and experimental tests, we set $N_x = N_y = 50$. The Fourier coefficients were computed for $K \in \mathbf{Z}^2 = [0, 1, \cdots, 50]^2$.

f) **Sensor Parameters and Automatic Target Recognition (ATR) algorithm abstraction**: The sensor was assumed to be a gimbal stabilized visible spectrum camera, so that it always looked down despite the vehicle roll/pitch.

-Field of view (FOV): Since the camera always look straight down, we assume that the camera has a circular FOV with a radius $R = 25ft$.

-ATR algorithm abstraction: We assume that each vehicle has an onboard ATR algorithm for target detection and classification. For the purpose of this paper, we use an abstraction of the ATR algorithm in terms of its performance characterized by: probability of target detection p_d and sensor noise. When a target falls within the circular FOV of the sensor, a random sensor measurement is generated with a probability $p_d = 0.7$. To account for errors in the sensor measurements of the target location, we add a zero mean Gaussian noise with covariance $Q_s = \begin{pmatrix} 25^2 & 0 \\ 0 & 25^2 \end{pmatrix}$ to the true target location. We assume that each target is adversarial or non-adversarial. The classification probability for a target (probability of it being adversarial) is updated using Bayesian updates as described in Ref. 13.

-Overall probability of Missed Detection: The requirement on the overall probability of missed detection P_{dreq} used to set the thresholds in SPRT was set to 10^{-5}. For details on the computation of thresholds for SPRT, see Ref. 13.

TEST OUTPUT DATA For the duration of each test run we record both the actual and commanded vehicle states and the associated time stamps. For the adaptive search test case, the target detection and classification probabilities, and the target location estimate and covariance are recorded for post-processing and performance visualization.

IV.B. Experimental Implementation

The SMC approach was validated through flight tests using a Maxi Joker 3 RC electric helicopter. The Maxi Joker 3 variant used in the flight tests had a flybar with a torque tube driven tail. The helicopter has a gross weight capability of 20lb with a flight time of approximately 10 minutes. A schematic of the experimental set-up is shown in Fig. 5. An onboard embedded processing board hosts a trajectory tracking controller that receives trajectory commands for the current control horizon from the ground control station via a dedicated radio datalink. The ground control station executes the SMC planner to generate the trajectory commands. The datalink also relays the current vehicle state and actuator and sensor status as feedback to the GCS for the SMC planner to compute the new trajectory command. In the event of an emergency, a human safety pilot can override and take over control of the vehicle via a dedicated datalink.

V. RESULTS

In this section we describe the results obtained for the coverage and search demonstrations in the SIL and flight tests.

V.A. SIL Tests

For SIL testing, we mainly focused on the coverage problem. Recall that for uniform coverage of a prior μ, the SMC controller generates trajectories which sample the prior uniformly. Figures 6 and 7 show the SMC trajectories for the two priors defined in Figures 4a and 4b respectively. For the prior with a bimodal distribution, as expected, the vehicle spends most of the time close to the two peaks of the two Gaussian distributions (with few transitions). This is because the vehicle spends more time resolving areas with higher probability of finding a target. For coverage of a uniform prior on a non-convex domain, the vehicles spend most of the time in the free areas between the foliage regions, where the prior is non-zero. For SIL demonstration of adaptive search of stationary targets, we refer the reader to Ref. 13.

V.B. Flight Tests

V.B.1. Coverage Tests

The first test was for a uniform prior in the search region shown in Figure 3. Figure 8 shows the actual Maxi-Joker trajectories. Also shown are the commanded and actual values for the position of the helicopter. It is clear from these figures that the Maxi-Joker can accurately track the SMC trajectories generated based on a Dubins model with suitable parameters. Figure 9 show Maxi-Joker trajectories for the prior with a bimodal distribution as shown in Figure 4a. As can be seen, the trajectories in Figure 9 are very similar to the trajectories in Figure 6 obtained for the SIL tests [a].

[a]In the figures for the last two experimental flight tests, the search region is similar to that used in the SIL tests, but shifted to the left by 50ft.

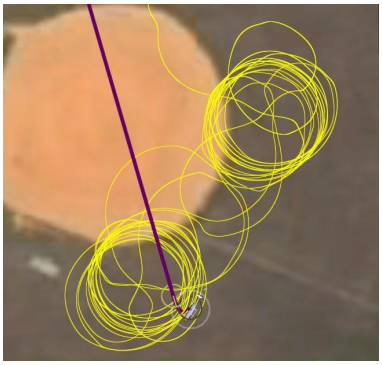


Figure 6. SIL demonstration of coverage of the prior with a bimodal distribution as shown in Figure 4a.

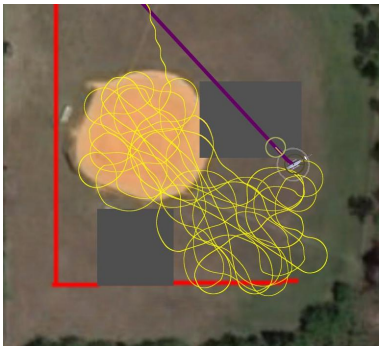
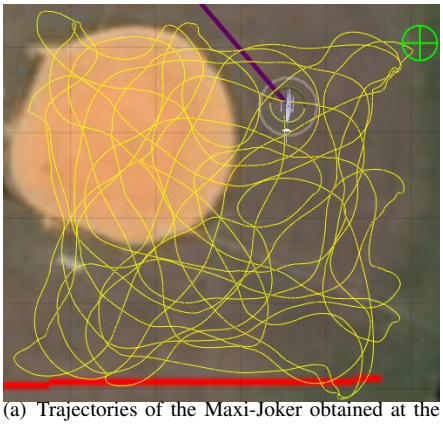
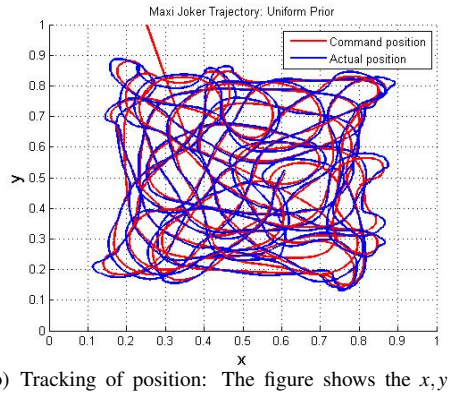


Figure 7. SIL demonstration of coverage of a uniform prior on a non-convex domain as shown in Figure 4b.

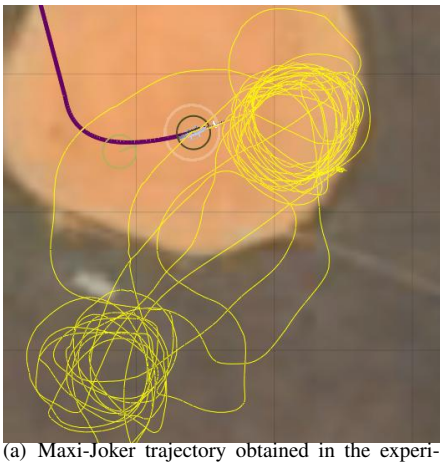


(a) Trajectories of the Maxi-Joker obtained at the flight test site for coverage of a uniform prior.

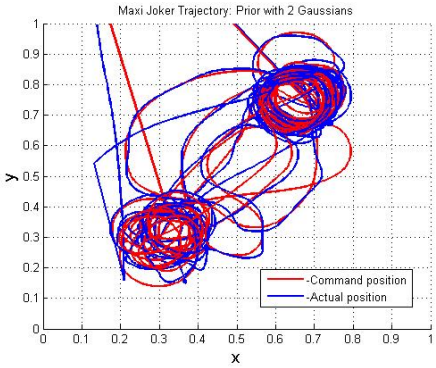


(b) Tracking of position: The figure shows the x, y coordinates of the vehicle in dimensionless units. The red curve is the commanded position and the blue curve is actual helicopter position. As can be seen, the commanded and actual positions are very close to each other.

Figure 8. Experimental demonstration of coverage for a uniform prior.



(a) Maxi-Joker trajectory obtained in the experiment for coverage of the prior with a bimodal distribution as shown in Figure 4a.

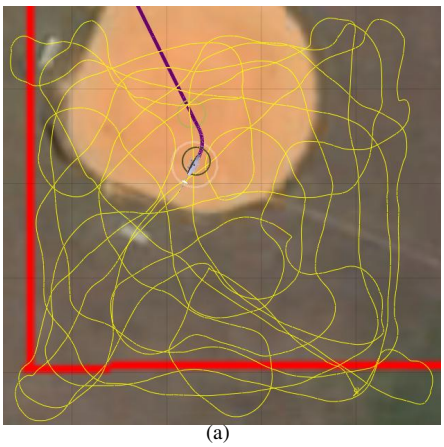


(b) Tracking of position: The figure shows the x, y coordinates of the vehicle in dimensionless units. The red curve is the commanded position and the blue curve is actual helicopter position.

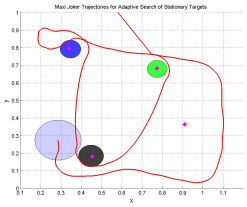
Figure 9. Experimental demonstration of coverage for a prior with a bimodal distribution.

V.B.2. Adaptive Search Tests

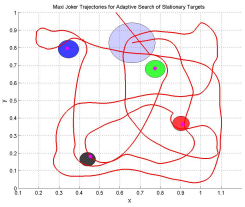
For the adaptive search test, we assumed that there were 4 stationary targets. Two of the targets are to be classified as adversarial and two of them are to be classified as non-adversarial. The initial search prior was a uniform prior within the search region. Figures 10a-e show the Maxi-Joker trajectories and the detected targets with the associated uncertainty in their position. As the Maxi-Joker explores the domain, all 4 targets are eventually discovered, and the uncertainty in all target locations shrinks to almost zero.



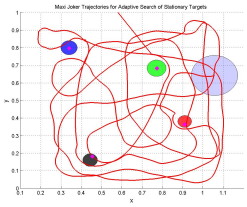
(a)



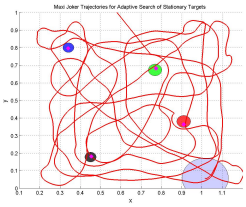
(b)



(c)



(d)



(e)

Figure 10. The left figure shows the Maxi-Joker trajectory obtained for adaptive search. The figures on the right show the detected targets together with the Maxi-Joker trajectory. The size of the ball around each discovered target is proportional to the uncertainty (trace of covariance) associated with the target location. The gray circle denotes the sensor FOV.

VI. CONCLUSIONS

In this paper we have demonstrated for the first time the practical feasibility of Spectral Multiscale Coverage and its application to search missions. We provided the algorithmic details for the implementation of SMC and illustrated the approach through SIL tests in several scenarios. We also successfully flight-tested the SMC approach on a single Maxi Joker 3 RC helicopter in an outdoor environment with virtual terrain, virtual targets and simulated sensors. This

preliminary successful experimental validation of the SMC approach provides promising evidence for the approach to be of practical value in Intelligence, Surveillance and Reconnaissance applications.

In the future it would be desirable to couple the SMC algorithm with perception algorithms for target detection and geolocalization, and test with real targets. Along similar lines, demonstrating the SMC approach in a multi vehicle setting would be desirable. This would require dealing with collision avoidance issues. Refinement and extension of the SMC approach to deal with such contingencies will be the subject of future development efforts and technical publications. Another related area of future research is to study human operator collaboration with SMC to improve the coverage/search performance and reliability.

VII. ACKNOWLEDGMENTS

The funding provided by United Technologies Research Center for this work is greatly appreciated. The authors are grateful to Andrzej Banaszuk for his feedback and suggestions. We also thank our test pilot Evan Richey for his flight test support.

References

[1]Cortes, J., Martinez, S., Karatas, T., and Bullo, F., "Coverage Control for Mobile Sensing Networks," *IEEE Transactions of Robotics and Automation*, Vol. 20, No. 2, 2004, pp. 243–255.

[2]Choset, H., "Coverage for robotics - A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, Vol. 31, 2001, pp. 113–126.

[3]Hussein, I. and Stipanović, D., "Effective Coverage Control for Mobile Sensor Networks with Guaranteed Collision Avoidance," *IEEE Transactions on Control Systems Technology, Special Issue on Multi-Vehicle Systems Cooperative Control with Applications*, Vol. 15, No. 4, 2007, pp. 642–657.

[4]Mathew, G. and Mezić, I., "Spectral Multiscale Coverage: A Uniform Coverage Algorithm for Mobile Sensor Networks," *IEEE Conf. on Decision and Control*, Shanghai, China, Dec. 2009.

[5]Mathew, G. and Mezić, I., "Metrics for ergodicity and design of ergodic dynamics for multi-agent systems," *Physica D: Nonlinear Phenomena*, Vol. 240, No. 4-5, 2010.

[6]Grocholsky, B., Keller, J., Kumar, V., and Pappas, G., "Cooperative air and ground surveillance," *Robotics Automation Magazine, IEEE*, Vol. 13, No. 3, sept. 2006, pp. 16 –25.

[7]Riehl, J. R., Collins, G. E., and Hespanha, J. P., "Cooperative Search by UAV Teams: A Model Predictive Approach Using Dynamic Graphs," *IEEE Trans. Aerospace and Electronic Syst.*, Vol. 47, Oct. 2011, pp. 2637–2656.

[8]Vincent, P. and Rubin, I., "A framework and analysis for cooperative search using UAV swarms," *Proceedings of the 2004 ACM symposium on Applied computing*, ACM, 2004, pp. 79–86.

[9]Jin, Y., Liao, Y., Minai, A., and Polycarpou, M., "Balancing search and target response in cooperative unmanned aerial vehicle (UAV) teams," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, Vol. 36, No. 3, 2005, pp. 571–587.

[10]Stone, L., *Theory of optimal search*, Academic Press New York, 1975.

[11]Hubenko, A., Fonoberov, V., Mathew, G., and Mezić, I., "Multiscale Adaptive Search," *IEEE Sytem Man and Cybernetics B*, Vol. 41, No. 4, 2011.

[12]Mathew, G., Surana, A., and Mezić, I., "Uniform Coverage Control of Mobile Sensor Networks for Dynamic Target Detection," *IEEE Conf. on Decision and Control*, Atlanta, GA, Dec. 2010.

[13]Surana, A., Mathew, G., and Kannan, S., "Coverage Control of Mobile Sensors for Adaptive Search of Unknown Number of Targets," *International Conference on Robotics and Automation, accepted to appear*, 2012.

[14]Bhatia, A., Graziano, M., Karaman, S., Naldi, R., and Frazzoli, E., "Dubins trajectory tracking using commercial off-the-shelf autopilots," *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008.

[15]Paley, D. and Warshawsky, D., "Reduced-order dynamic modeling and stabilizing control of a micro-helicopter," *Proceedings of the 47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Orlando, Florida*, 2009.

[16]Dubins, L. E., "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents," *American Journal of Mathematics*, Vol. 79, No. 3, 1957.

[17]Johnson, E. N. and Kannan, S. K., "Adaptive Trajectory Control for Autonomous Helicopters," *Journal of Guidance Control and Dynamics*, Vol. 28, No. 3, 2005, pp. 524–538.

[18]Kannan, S. K. and Johnson, E. N., "Model Reference Adaptive Control with a Constrained Linear Reference Model," *IEEE Conference on Decision and Control*, Atlanta, GA, December 2010.