Energy-aware Persistent Coverage and Intruder Interception in 3D Dynamic Environments

William Bentz and Dimitra Panagou

***Abstract*— This paper considers the persistent coverage of a 2-D manifold that has been embedded in 3-D space. The manifold is subject to continual collisions by intruders that are generated with random trajectories. The trajectories of intruders are estimated online with an extended Kalman filter and their predicted impact points contribute normally distributed decay terms to the coverage map. A formal hybrid control strategy is presented that allows for power-constrained 3-D free-flyer agents to persistently monitor the domain, track and intercept intruders, and periodically deploy from and return to a single charging station on the manifold. Guarantees on intruder interception with respect to agent power lifespans are formally proven. The efficacy of the algorithm is demonstrated through simulation.**

I. Introduction

The advent of inexpensive autonomous research platforms has spurred recent interest in teams of mobile sensors collaborating on complex surveillance and monitoring tasks. Coverage control problems have been particularly popular due to their numerous applications: e.g., environmental monitoring [1], battlefield surveillance [2], lawn mowing and vacuuming, search and rescue [3], and hull inspections [4], [5]. The latter application is actively supported by NASA whose work on the Mini AERCam paves the way for a future of extravehicular robotic (EVR) free flyers performing independent visual inspections of spacecraft exterior areas of interest [6]. Free flyer visual inspection is the primary motivating example for our work.

Coverage is often partitioned into three classes of problems: static, dynamic, and persistent. Static coverage problems (e.g., area coverage, k-coverage and point coverage) often explore the optimal arrangement of sensor nodes in a network and the agents tend to immobilize after this arrangement has been achieved [7]. Dynamic coverage problems involve the active exploration of a domain. Agents typically must sweep their sensors over all points of a domain until some desired level of coverage has been achieved [8]–[10]. Persistent coverage is often similar to dynamic coverage with the addition of information decay within the environment: i.e., agents are required to continually return to areas of interest in order to restore a deteriorating coverage level.

William Bentz is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor; wbentz@umich.edu.

Dimitra Panagou is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor; dpanagou@umich.edu.

The authors would like to acknowledge the support of the Automotive Research Center (ARC) in accordance with Cooperative Agreement W56HZV-14-2-0001 U.S. Army TARDEC in Warren, MI and the support by an Early Career Faculty grant from NASAs Space Technology Research Grants Program

The term "persistent coverage" appears as early as [11] where agents must cover all points in a 2-D convex polygonal domain every $T^\star$ time units. This was accomplished with the design of concentric polygonal trajectories with agents following closed paths in steady state. The work in [12] is similar but also introduces a linear coverage decay rate for specific points of interest. In this paper, as well as [13], controller design is akin to regulating the velocity along paths generated offline to increase observation time at select points of interest. As the decay rates are known and time invariant, optimal speed control is computed via linear programming.

Palacios-Gasós et. al have published multiple works recently on persistent coverage [14]–[16] which also consider time-invariant coverage decay rates. These works use techniques from discrete optimization and linear programming to iteratively compute optimal paths. Similar techniques are used in [17] which also considers that agents must periodically return to refueling depots.

Common themes through all of these persistent coverage works are convex 2-D domains, predictable environments, and simplified sensing and dynamic models for agents. Coverage surfaces embedded in $\mathbb{R}^3$ are considered in [18]; however, this work is closer to that of [11] in that agents also follow preplanned trajectories without any consideration of spatially dependent coverage decay maps.

Monitoring of stochastic environments is presented in [19], [20], outside of the strict persistent coverage formulation. In [19], the authors consider that agents must observe events at multiple points of interest and the precise arrival times of events are unknown *a priori*. Arrival time statistics are used to inform a multi-objective scheduling protocol that results in fixed cyclic servicing policies. In [20], the environment contains smart intruders which actively attempt to evade a camera surveillance network. Camera motion is restricted to a single pan axis and thus the the system model is essentially that of a 1-D pursuit evasion problem.

The primary contributions are: a formal hybrid control strategy for multi-agent persistent coverage of non-planar surfaces embedded in $\mathbb{R}^3$ that does not make overly simplifying assumptions with respect to agent dynamic and sensing models, a formal guarantee on agent interception of stochastic intruders, and an energy-aware agent deployment and scheduling protocol. The primary hybrid mode builds upon the authors' previous work in [21]. In addition, agents now operate with finite resources and are required to periodically return to a refueling station while observing stochastic events at locations and times that are not known *a priori*. Unlike related works, these events may occur anywhere in

the domain and the events themselves introduce non-linear time varying decay terms to the coverage map.

This paper is organized as follows: Section II describes the agents sensing and kinematic models, local coverage strategy, and intruder detection model. Section III describes our particle (e.g., intruder) interception hybrid mode. Section IV details our energy-aware domain partitioning and scheduling protocols. Section V presents our collision avoidance hybrid mode with the formal hybrid automaton presented in Section VI. Simulations and conclusions are presented in Sections VII and VIII respectively.

II. Problem Formulation

A. System Model

Consider a network of spherical autonomous agents indexed $i \in \{1, ..., N\}$, of radius $\imath_i$, whose motion is subject to 3-D rigid body kinematics [22]:

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \end{bmatrix} = \begin{bmatrix} \cos\Theta_i \cos\Psi_i & \sin\Phi_i \sin\Theta_i \cos\Psi_i - \cos\Phi_i \sin\Psi_i & \cos\Phi_i \sin\Theta_i \cos\Psi_i + \sin\Phi_i \sin\Psi_i \\ \cos\Theta_i \sin\Psi_i & \sin\Phi_i \sin\Theta_i \sin\Psi_i + \cos\Phi_i \cos\Psi_i & \cos\Phi_i \sin\Theta_i \sin\Psi_i - \sin\Phi_i \cos\Psi_i \\ -\sin\Theta_i & \sin\Phi_i \cos\Theta_i & \cos\Phi_i \cos\Theta_i \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix}, \quad (1)$$

$$\begin{bmatrix} \dot{\Phi}_i \\ \dot{\Theta}_i \\ \dot{\Psi}_i \end{bmatrix} = \begin{bmatrix} 1 & \sin\Phi_i \tan\Theta_i & \cos\Phi_i \tan\Theta_i \\ 0 & \cos\Phi_i & -\sin\Phi_i \\ 0 & \sin\Phi_i \sec\Theta_i & \cos\Phi_i \sec\Theta_i \end{bmatrix} \begin{bmatrix} q_i \\ r_i \\ s_i \end{bmatrix}, \quad (2)$$

where $p_i = [x_i \;\; y_i \;\; z_i]^T$ is the position vector and $\Omega_i = [\Phi_i \;\; \Theta_i \;\; \Psi_i]^T$ is the vector of 3-2-1 Euler angles taken with respect to a global Cartesian coordinate frame $\mathcal{G}$ with origin $\mathcal{O}$. The linear velocities $[u_i \;\; v_i \;\; w_i]^T$ and angular velocities $[q_i \;\; r_i \;\; s_i]^T$ are both presented in the body fixed frame $\mathcal{B}_i$ with origin p_i. The state vector of agent i is defined as $\eta_i = [p_i^T \;\; \Omega_i^T]^T$. In the sequel, the rotation matrices of (1) and (2) shall be denoted $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively. The agents travel within a stationary domain, $\mathcal{D} \subset \mathbb{R}^3$. Their task is to survey a two-dimensional manifold, $\mathcal{C} \subset \mathcal{D}$, known as our surface of interest. For the purpose of this work, we assume that the surface is an ellipsoid of revolution with semi-major axis $x_{\mathcal{C},r}$ and semi-minor axis $z_{\mathcal{C},r}$ aligned with the global coordinate axes $\hat{x}_\mathcal{G}$ and $\hat{z}_\mathcal{G}$ respectively with center at $\mathcal{O}$. Note that circumflex (i.e., hat) symbols denote unit vectors.

Each agent, i, is equipped with a forward facing sensor whose footprint shall be referred to as $\mathcal{S}_i$. A spherical sector model is chosen for $\mathcal{S}_i$ as it is representative of the space typically observable to a single camera lens. The center-line $\mathcal{S}_i$ provides anisotropic sensing data which degrade in quality towards the periphery of the footprint in a similar manner to that of a camera lens. Spherical sector $\mathcal{S}_i$ has vertex at p_i and a centerline parallel to the $\hat{x}_{\mathcal{B}_i}$ axis. It is also characterized by field of view $2\alpha_i > 0$ and sensing range $R_i > 0$ where $R_i > \imath_i$. See [21] for a full sensing model description.

Denote the quality of information available at each point in $\mathcal{S}_i$ as $S_i(\eta_i(\tau), \tilde{p})$ where $\tilde{p} = [\tilde{x} \;\; \tilde{y} \;\; \tilde{z}]^T$ is the position of a point in $\mathcal{D}$ with respect to $\mathcal{G}$. Define the coverage level provided by agent i at time t as $Q_i(t, \tilde{p}) = \int_0^t S_i(\eta_i(\tau), \tilde{p})\, C(\tilde{p})\, d\tau$, where C is defined as $C(\tilde{p}) = \{1 \,\forall \tilde{p} \in \mathcal{C} \wedge 0 \,\forall \tilde{p} \notin \mathcal{C}\}$ and encodes that the accumulation of information only occurs along our surface of interest, $\mathcal{C}$.

As the agents cover $\mathcal{C}$, a set of N_p high-speed particles denoted $k \in \{1, ..., N_p\}$, each of which travels at a constant linear velocity, pass through the domain. Each particle shall have an associated map decay term, $\Lambda_k(\tau, \tilde{p})$, which is defined later in Section II-C. We define the global coverage level as $Q(t, \tilde{p}) = \sum_{i=1}^{N} Q_i(t, \tilde{p}) - \sum_{k=1}^{N_p} \int_0^t \Lambda_k(\tau, \tilde{p})\, C(\tilde{p})\, d\tau$.

In this work, coverage refers to the accumulation of sensing data over time. Points, $\tilde{p}$, are said to be sufficiently covered when $Q(t, \tilde{p}) \geq C^\star$. The purpose of this work is to derive a hybrid control strategy which persistently sweeps $\mathcal{S}_i$ across $\mathcal{C}$ while emphasizing surveillance around the predicted impact points of intruders $k \in \{1, ..., N_p\}$ on $\mathcal{C}$. This must be done while avoiding collisions as defined in *Definition 1*.

Definition 1: Agent i avoids collision so long as $\|p_i(t) - p_j(t)\| > \imath_i + \imath_j$, $\forall j \neq i \in \{1, ..., N\}$ and $\|n_i\| > \imath_i$ where the vector n_i has direction normal to $\mathcal{C}$ and length equal to the Euclidean distance of its intersection point on $\mathcal{C}$ to p_i.

Furthermore, agents operate with finite power resources and are required to periodically return to a fueling station denoted $\mathcal{F}$. Thus, a scheduling protocol is derived whereby agents periodically deploy from $\mathcal{F}$ to cover within an assigned partition of $\mathcal{C}$. These partitions are bounded by latitude lines and are sorted by geodesic distance from $\mathcal{F}$ with agents transferring to increasingly closer partitions as their power resource dwindles requiring a return to $\mathcal{F}$. This partitioning scheme also has the benefit of ensuring that the network of agents is well distributed across $\mathcal{C}$ with agents nominally assigned to intercept intruders with predicted impact points within their own partition.

B. Local Coverage

Our previous work [21], considers the same environment of stochastic intruders with agents operating under a single control law. This controller, referred to as local coverage mode in the sequel, constitutes the first of five hybrid modes in our new automaton. Local coverage is gradient following in nature and essentially commands agent i to always seek to orient and translate $\mathcal{S}_i$ such that the volume of uncovered space intersecting $\mathcal{S}_i$ is increased. The control laws are derived by defining an error function of $Q(t, \tilde{p})$ with respect to its desired level $C^\star$ and then using Lyapunov-like arguments to drive this error function towards smaller values. They take the form: $\begin{bmatrix} u_i^{loc} & v_i^{loc} & w_i^{loc} & r_i^{loc} & s_i^{loc} \end{bmatrix}^T = \begin{bmatrix} \frac{k_u a_{i1}}{\sqrt{a_{i1}^2 + a_{i2}^2 + a_{i3}^2}} & \frac{k_v a_{i2}}{\sqrt{a_{i1}^2 + a_{i2}^2 + a_{i3}^2}} & \frac{k_w a_{i3}}{\sqrt{a_{i1}^2 + a_{i2}^2 + a_{i3}^2}} & \bar{r}_i sat\left(\frac{k_r a_{i4}}{\bar{r}_i}\right) & \bar{s}_i sat\left(\frac{k_s a_{i5}}{\bar{s}_i}\right) \end{bmatrix}^T + \begin{bmatrix} \rho_{l,i}^T & \rho_{a,i} \cdot \hat{y}_{\mathcal{B}_i} & \rho_{a,i} \cdot \hat{z}_{\mathcal{B}_i} \end{bmatrix}^T$, where $\rho_{l,i} = -\ln\left(\frac{1}{\gamma R_i - \imath_i}\left(\|n_i\| - \imath_i\right)\right)\mathcal{R}_1^{-1}\hat{n}_i$ and $\rho_{a,i} = \xi \mathcal{R}_2^{-1} \begin{bmatrix} 0 \\ \arcsin(\hat{n}_i \cdot \hat{z}_\mathcal{G}) - \Theta_i \\ \operatorname{atan2}(-\hat{n}_i \cdot \hat{y}_\mathcal{G}, -\hat{n}_i \cdot \hat{x}_\mathcal{G}) - \Psi_i \end{bmatrix}$.

Note that $a_{i,\ell}(t, \acute{Q}_i(t,\tilde{p})), \forall \ell \{1, ..., 5\}$ are functions of agent i's modified coverage map $\acute{Q}_i(t,\tilde{p}) = Q(t,\tilde{p}) + M_i(t,\tilde{p})$, where $M_i(t,\tilde{p})$ shall be defined in Section IV. The functional dependence in the local coverage control laws are omitted in the interest of space. $M_i(t,\tilde{p})$ essentially blocks off portions of the map as being uninteresting for agent i by adding $C^\star$ barriers to $Q(t,\tilde{p})$. Thus, $a_{i,\ell}$ tend to zero along the boundary of agent i's assigned partition. For derivations of the local coverage laws and definitions of $a_{i,\ell}$, see Section III of [21].

$\rho_{l,i}$ is a collision avoidance term with respect to the surface of interest. It takes a value of zero when agent i's normalized distance from $\mathcal{C}$ is γR_i for $\gamma \in (0,1]$ and is logarithmically repulsive and attractive from the surface when the distance is decreased or increased respectively. $\rho_{a,i}$, for $\xi << 1$, encodes that the agents should tend to align $\hat{x}_{\mathcal{B}_i}$ with $-\hat{n}_i$ if the coverage terms associated with r_i and s_i have become sufficiently small. The physical meaning of $\rho_{a,i}$ is to direct $\mathcal{S}_i$ back onto $\mathcal{C}$ if it has reached a configuration in which it no longer intersects $\mathcal{C}$. See Fig. 1 for further illustration of the effects of $\rho_{l,i}$ and $\rho_{a,i}$.

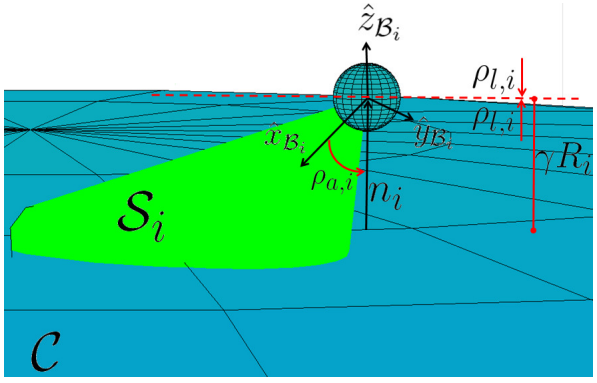

Fig. 1. As agent i explores $\mathcal{C}$, $\rho_{l,i}$ is parallel to n_i for $\|n_i\| < \gamma R_i$, antiparallel to n_i for $\|n_i\| > \gamma R_i$, and the zero vector otherwise. This term prevents collision of i with $\mathcal{C}$ and prevents i from flying away from $\mathcal{C}$. $\rho_{a,i}$ tends to direct $\mathcal{S}_i$ onto $\mathcal{C}$.

$\bar{r_i}$ and $\bar{s_i}$ are saturation limits for the coverage angular velocity inputs to the system. k_u, k_v and k_w are tuning gains which are chosen to satisfy $\sqrt{k_u^2 + k_v^2 + k_w^2} \leq U_{max}$. As $\rho_{l,i}$ is normal to the surface, it can be shown that U_{max} is an upper bound to agent velocity tangential to $\mathcal{C}$.

C. Intruder Detection Model

We assume that an omnidirectional range sensor (e.g., LiDAR) is co-located with $\mathcal{O}$ and provides measurements of each intruder's position in spherical coordinates. Define a model for the motion of intruder k:

$$\dot{\tilde{q}}_k(t) = \begin{bmatrix} 0_{3\times3} & \mathbb{I}_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tilde{q}_k(t), \tag{3}$$

$$\tilde{z}_k(t) = \begin{bmatrix} \sqrt{x_k^2 + y_k^2 + z_k^2} \\ \text{atan2}(y_k,\, x_k) \\ \arccos\left(\frac{z_k}{\sqrt{x_k^2+y_k^2+z_k^2}}\right) \end{bmatrix} + \epsilon, \tag{4}$$

where $\tilde{q}_k = [x_k,\, y_k,\, z_k,\, \dot{x}_k,\, \dot{y}_k,\, \dot{z}_k]^T$ and $\tilde{z}_k = [\rho_k,\, \theta_k,\, \psi_k]^T$ are the Cartesian state and spherical coordinate measurement vectors of intruder k resolved in $\mathcal{G}$. ρ_k, θ_k, and ψ_k are the range, azimuthal angle, and polar angle of k respectively. Intruders are assumed to be particles and shall interchangeably be referred to as such.

If intruder k is detected at time t_{dk}, we initialize the state estimate at $t_{dk'} = t_{dk} + \Delta t$ where Δt is lower bounded by the time required to record two measurements of $\tilde{z}_k(t)$. The evolution of intruder state estimate $\hat{\tilde{q}}_k(t)$ and associated covariance $\mathbf{P}_k(t)$ are then estimated online via extended Kalman Filter. For a full description of this model, see [21].

At any time t, we define our decay rate map for intruder k in terms of its predicted position and covariance evolution over a horizon $T_{H,k}(t)$. As the intruders are assumed to travel at fixed velocities, the predicted values for Cartesian position $\tilde{p}'_k(t+\tau)$ and associated covariance $\mathbf{P}_k(t+\tau)$ may be defined as $\tilde{p}'_k(t+\tau) = G(\tau)\hat{\tilde{q}}_k(t)$ and $\mathbf{P'}_k(t+\tau) = G(\tau)\mathbf{P}_k(t)G(\tau)^T$ respectively where $G(\tau) = [\mathbb{I}_{3\times3} \;\; \tau\mathbb{I}_{3\times3}]$.

We define the decay rate map associated with particle k as the integral of our predicted normal distribution $\mathcal{N}(\tilde{p}'_k(t+\tau), \mathbf{P'}_k(t+\tau))$ through horizon T_H:

$$\Lambda_k(t,\tilde{p}) = \int_0^{T_{H,k}(t)} \lambda_k \mathcal{N}\left(\tilde{p}'_k(t+\tau), \mathbf{P'}_k(t+\tau)\right) d\tau. \tag{5}$$

For $t < t_{dk'}$, define $\Lambda_k(t,\tilde{p}) = 0$, $\forall \tilde{p} \in \mathcal{D}$. Our formulation for (5) essentially takes a normal distribution for the position of particle k at time t and cumulatively sweeps it forward in time up to our horizon $T_{H,k}(t)$. The horizon is lower-bounded by an estimate of the remaining time until impact of particle k on $\mathcal{C}$. This may be computed using $\tilde{q}_k(t)$ along with the surface geometry. With this design, $Q(t,\tilde{p})$ decays along the predicted trajectory of k with tapering omnidirectional decay rates spreading out from the predicted path. This design lends itself naturally to our local coverage formulation, which is gradient following in nature, in that the agents may follow these tapered decay paths towards the predicted impact points on our surface of interest. The parameter $\lambda_k > 0$ may be adjusted to scale how rapidly the coverage level will decay in time.

III. Particle Intercept Mode

We denote the detection and state estimation of particle k as an event, ϵ_k, which concludes at $t_{dk'}$. Assuming that particle k is embedded within the surface upon impact, its position shall intersect $\mathcal{C}$ at most one time. Thus, we define particle k's estimated impact (collision) time as $t_{ck} = \min\left(t \in \mathbb{R}^+ \mid \frac{(\hat{x}_k+\dot{\hat{x}}_k t)^2}{x_{\mathcal{C},r}^2} + \frac{(\hat{y}_k+\dot{\hat{y}}_k t)^2}{x_{\mathcal{C},r}^2} + \frac{(\hat{z}_k+\dot{\hat{z}}_k t)^2}{z_{\mathcal{C},r}^2} = 1\right)$, with estimated point of impact $\tilde{p}'_k(t_{ck}) = G(t_{ck} - t)\hat{\tilde{q}}_k(t)$.

Upon conclusion of event ϵ_k, particle k is assigned to a free agent i with the minimum geodesic distance from the estimated point of impact. We define a new index, i_k, as the index of the agent assigned to intercept particle k. $i_k := i$ may occur on condition that $i_p \neq 1$ and $f_i \neq 1$ where $f_i \in \{0,1\}$ is a particle assignment flag for agent i defined

as 0 when the agent is free (i.e., not currently assigned a particle). i_p, the power index of agent i, shall be defined in Section IV.

When agent i has been assigned to intercept particle k, f_i is set to 1 and it is said to have transitioned into particle intercept mode. In this mode, agent i follows an optimal trajectory to the point $\tilde{p}'_k(t_{ck})$ and remains within some distance tolerance ε_1 until $t > t_{ck}$ at which time f_i is set to 0. The optimal trajectory is referred to as a geodesic and its computation may be executed in an iterative manner. Specifically, we use Vincenty's formulae as presented in [23]. For cases involving nearly antipodal points in which the standard inverse method does not converge, we use Vincenty's supplemental algorithm presented in [24].

As an input, Vincenty's algorithm requires two points, current and desired position, on the surface of an ellipsoid of revolution. The algorithm also requires the length of the semi-major and semi-minor axes of the ellipsoid and it returns a heading angle measured clockwise from North. This heading angle shall be referred to as χ_i. We now define the nominal heading unit vector $\hat{\nu}_i$ which lies in the plane tangent to the surface at p_i. It may be computed by rotating the North-pointing vector at p_i clockwise by an angle of χ_i within the tangent plane.

As with our local coverage strategy, it is assumed that the agent shall nominally maintain a distance γR_i normal to $\mathcal{C}$. We define an ellipsoid of revolution, $\mathcal{C}'$ which is concentric with $\mathcal{C}$ and has the property that each semi-principal axis is γR_i longer than its associated counterpart in $\mathcal{C}$, i.e., $x_{\mathcal{C}',r} = x_{\mathcal{C},r} + \gamma R_i$, $y_{\mathcal{C}',r} = y_{\mathcal{C},r} + \gamma R_i$, and $z_{\mathcal{C}',r} = z_{\mathcal{C},r} + \gamma R_i$. Note that as this is an ellipsoid of revolution we have that $x_{\mathcal{C}',r} = y_{\mathcal{C}',r}$. The nominal trajectories of i shall be attractive to $\mathcal{C}'$. Thus, χ_i and $\hat{\nu}_i$ shall be calculated at any given time with respect to this $\mathcal{C}'$.

The position controller used to guide agent i to $\tilde{p}'_k(t_{ck})$ is composed of two additive terms: one which commands velocity tangential to $\mathcal{C}'$ along $\hat{\nu}_i$ and one logarithmic term which commands velocity normal to $\mathcal{C}'$ in order to maintain the property that $\|n_i\| \approx \gamma R_i$. The particle intercept mode position control law is:

$$\begin{bmatrix} u_i^{pim} \\ v_i^{pim} \\ w_i^{pim} \end{bmatrix} = U_{max}\mathcal{R}_1^{-1} \frac{\hat{\nu}_i - \ln\left(\frac{1}{\gamma R_i - r_i}\left(\|n_i\| - r_i\right)\right)\hat{n}_i}{\left\|\hat{\nu}_i - \ln\left(\frac{1}{\gamma R_i - r_i}\left(\|n_i\| - r_i\right)\right)\hat{n}_i\right\|}. \tag{6}$$

As (6) commands the vehicle to follow the optimal length path along $\mathcal{C}'$ to $\tilde{p}'_k(t_{ck})$, we can establish a guarantee on the feasibility of particle interception. To simplify notation, define:

$$g = \left[1 + \sum_{n=1}^{\infty}\left(\frac{(2n-1)!!}{2^n n!}\right)^2 \frac{\left(\frac{x_{\mathcal{C}',r} - z_{\mathcal{C}',r}}{x_{\mathcal{C}',r} + z_{\mathcal{C}',r}}\right)^{2n}}{(2n-1)^2}\right]. \tag{7}$$

Theorem 1: Assuming that sufficient time is provided, i.e., $t_{ck} - t_{dk'} > \frac{\pi x_{\mathcal{C}',r} + \frac{\pi}{2}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g}{U_{max}}$, a continuous application of control law (6) shall guide agent i to reach impact point $\tilde{p}'_k(t_{ck})$ before t_{ck}.

Proof: Assume that at time $t_{dk'}$, agent i is at position $p_i(t_{dk'})$ which lies on $\mathcal{C}'$ and must travel to point $\tilde{p}'_k(t_{ck})$ which is also on $\mathcal{C}'$. Denote the length of this path as $S_{p_i(t_{dk'}),\tilde{p}'_k(t_{ck})}$. As (6) directs agent i to follow the shortest feasible path along $\mathcal{C}'$, we may upper bound $S_{p_i(t_{dk'}),\tilde{p}'_k(t_{ck})}$ by the sum of two paths: a path of constant latitude S_{lat} followed by a path of constant longitude S_{long}:

$$S_{p_i(t),\tilde{p}'_k(t_{ik})} \leq S_{lat} + S_{long}. \tag{8}$$

For two generic points on $\mathcal{C}'$, we have that:

$$S_{lat} \leq \pi x_{\mathcal{C}',r}, \tag{9}$$

$$S_{long} \leq \frac{\pi}{2}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g. \tag{10}$$

where the bound on S_{lat} is half of the circumference of our ellipsoid of revolution about its equator and the bound on S_{long} is half of the perimeter of our revolved ellipse. The infinite series expression term, denoted g in (10), is first presented in [25]. Assuming that agent i follows the geodesic path at velocity U_{max}, we have that:

$$S_{p_i(t_{dk'}),\tilde{p}'_k(t_{ck})} = U_{max} t_{ik}, \tag{11}$$

where t_{ik} is the time for agent i to travel to the impact point of particle k along the geodesic. Substituting (11) for $S_{p_i(t_{dk'}),\tilde{p}'_k(t_{ck})}$ in (8) and combining with inequalities (9) and (10) yields: $U_{max} t_{ik} \leq \pi x_{\mathcal{C}',r} + \frac{\pi}{2}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g$, which may be rearranged to form:

$$t_{ik} \leq \frac{\pi x_{\mathcal{C}',r} + \frac{\pi}{2}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g}{U_{max}}. \tag{12}$$

Agent i reaching $\tilde{p}'_k(t_{ck})$ before t_{ck} implies that: $t_{ik} < t_{ck} - t_{dk'}$. From (12), it is clear that this is guaranteed so long as: $\frac{\pi x_{\mathcal{C}',r} + \frac{\pi}{2}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g}{U_{max}} < t_{ck} - t_{dk'}$. This concludes the proof. ∎

As agent i travels towards $\tilde{p}'_k(t_{ck})$ along the geodesic, it is desirable that it should point $\mathcal{S}_i$ towards $\mathcal{C}$. Therefore, the orientation controller for particle intercept mode is similar to that of Section II-B:

$$\begin{bmatrix} q_i^{pim} \\ r_i^{pim} \\ s_i^{pim} \end{bmatrix} = \mathcal{R}_2^{-1} \begin{bmatrix} 0 \\ \arcsin\left(\hat{n}_i \cdot \hat{z}_{\mathcal{G}}\right) - \Theta_i \\ \operatorname{atan2}\left(-\hat{n}_i \cdot \hat{y}_{\mathcal{G}}, -\hat{n}_i \cdot \hat{x}_{\mathcal{G}}\right) - \Psi_i \end{bmatrix}, \tag{13}$$

which is essentially a proportional controller that tends to align $\hat{x}_{\mathcal{B}_i}$ with $-\hat{n}_i$.

IV. Energy-aware Scheduling Protocol

A. Domain Partitioning

As this is a persistent coverage protocol, which operates indefinitely, it is necessary to establish an agent deployment and scheduling protocol that realistically considers the agents' finite power and/or propulsive resources.

Our strategy is to periodically deploy agents from a fueling station $\mathcal{F}$ which we assume to be located at the North pole of $\mathcal{C}'$, i.e., at the point $[0 \;\; 0 \;\; z_{\mathcal{C}',r}]^T$. Define $T^\star$ as the power lifespan of each agent in the network. With our definition

for $T^\star$, it is intuitive that deployment windows from $\mathcal{F}$ will arise every $\frac{T^\star}{N}$ seconds.

In order to reduce redundancy between agents surveying $\mathcal{C}$, it is desirable to partition the domain and assign agents to monitor separate regions. Specifically, partitioning the domain by latitude, rather than longitude, ensures that agents are poised to intercept particles without the need for frequent crossings of the equator which tend to be associated with larger values of $S_{p_i(t_{dk'}),\tilde{p}'_k(t_{ck})}$ on an oblate spheroid.

Define $i_p(t) \in \{1, ..., N\}$ as the power index of agent i. Upon deployment from $\mathcal{F}$, agent i has power index $i_p = N$ and this index is reduced by one every $\frac{T^\star}{N}$ seconds until $i_p = 1$, i.e., agent i is the power critical agent. Thus, the power index obeys $i_p(t) = 1 + \mod\left(i - 2 - \left\lfloor \frac{tN}{T^\star} \right\rfloor, N\right)$, where the first argument of our modulo operation is the dividend and the second argument is the divisor. The lower-bracketed delimiters represent the floored division operation. This definition implies that agent $i = 1$ is deployed first.

We may now define our latitude partitions. The partitions are divided such that $N-1$ agents are assigned equal surface areas of $\mathcal{C}$ to explore. Each partition is characterized by an upper bound in $\hat{z}_\mathcal{G}$ denoted $\bar{z}_{i_p-2}$ and a lower bound $\bar{z}_{i_p-1}$. The surface area of our ellipsoid of revolution $\mathcal{C}$ is defined as:

$$A_\mathcal{C} = 2\pi x_{\mathcal{C},r}^2 \left(1 + \frac{1 + \left(1 - \frac{z_{\mathcal{C},r}^2}{x_{\mathcal{C},r}^2}\right)}{\left(\sqrt{1 - \frac{z_{\mathcal{C},r}^2}{x_{\mathcal{C},r}^2}}\right)} \operatorname{artanh}\left(\sqrt{1 - \frac{z_{\mathcal{C},r}^2}{x_{\mathcal{C},r}^2}}\right)\right). \tag{14}$$

The agent with $i_p = 2$ is assigned to monitor the partition characterized by upper bound at north pole of $\mathcal{C}$, i.e., $\bar{z}_0 = z_{\mathcal{C},r}$. The lower bound $\bar{z}_1$ may be computed by dividing (14) by $(N-1)$, equating with the integral of ellipse cross sectional circumferences parametrized by $\tilde{z}$, and then numerically solving for $\bar{z}_1$:

$$\frac{A_\mathcal{C}}{N-1} = \int_{z_{\mathcal{C},r}}^{\bar{z}_1} 2\pi \sqrt{\left(x_{\mathcal{C},r}^2 - \frac{x_{\mathcal{C},r}^2 \tilde{z}^2}{z_{\mathcal{C},r}^2}\right)\left(1 + \frac{\tilde{z}^2 x_{\mathcal{C},r}^4}{x_{\mathcal{C},r}^2\left(z_{\mathcal{C},r}^4 - z_{\mathcal{C},r}^2 \tilde{z}^2\right)}\right)} d\tilde{z}. \tag{15}$$

One may then iteratively solve for the remaining bounds for increasing values of i_p up to $i_p = N - 1$:

$$\frac{A_\mathcal{C}}{N-1} = \int_{\bar{z}_{i_p-2}}^{\bar{z}_{i_p-1}} 2\pi \sqrt{\left(x_{\mathcal{C},r}^2 - \frac{x_{\mathcal{C},r}^2 \tilde{z}^2}{z_{\mathcal{C},r}^2}\right)\left(1 + \frac{\tilde{z}^2 x_{\mathcal{C},r}^4}{x_{\mathcal{C},r}^2\left(z_{\mathcal{C},r}^4 - z_{\mathcal{C},r}^2 \tilde{z}^2\right)}\right)} d\tilde{z}. \tag{16}$$

The final computation of (16) for $i_p = N$ is not necessary as $\bar{z}_{N-1}$ is the south pole of $\mathcal{C}$, i.e., $\bar{z}_{N-1} = -z_{\mathcal{C},r}$, although this may be shown through numerical computation as well. Our partitioning strategy for the case where $N = 4$ is presented in Fig. 2.

Our map augmentation term $M_i(t, \tilde{p})$ for agent i, first referenced in Section II-B, is defined in terms of power index i_p:

$$M_{i_p}(t, \tilde{p}) = \begin{cases} 0, & \text{if } \bar{z}_{i_p-1} \le \tilde{z} \le \bar{z}_{i_p-2},\ \forall i_p \in \{2, ..., N\}; \\ 0, & \text{if } i_p = 1; \\ C^\star, & \text{otherwise.} \end{cases} \tag{17}$$

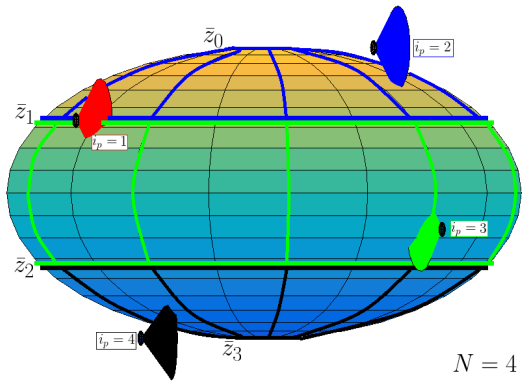

Fig. 2. Our domain partitioning scheme for $\mathcal{C}$ is illustrated above. Agents with $i_p \in \{2, 3, 4\}$ are indicated with blue, green and black $\mathcal{S}_i$ respectively. Their partitions are separated by latitude lines upper bounded at $\bar{z}_{i_p-2}$ and lower bounded at $\bar{z}_{i_p-1}$. The power critical has $\mathcal{S}_i$ indicated in red.

This map augmentation term encodes that regions of the domain outside of an agent's partition are "uninteresting" and our local coverage controller will tend to direct the $\mathcal{S}_i$ away from these regions when $\mathcal{S}_i$ intersects $\bar{z}_{i_p-1}$ or $\bar{z}_{i_p-2},\ \forall i \mid i_p \in \{2, ..., N\}$.

Note that no partition has been assigned to the agent for which $i_p = 1$. This is the power critical agent and it shall have flag $f_i := 1$ at the instant $i_p := 1$. The power critical agent cannot be assigned a new particle to intercept after $i_p := 1$ as this opens the possibility that particle assignment could occur near the end of the $\frac{T^\star}{N}$ time window during which time the agent with $i_p = 1$ should be transitioning back to $\mathcal{F}$ to exchange its power source. The power critical agent will instead spend the majority of this time window in local coverage mode assisting the other agents in gathering information. It can only be tasked with intercepting a particle if this assignment had occurred previously when $i_p = 2$. In this scenario, the agent should be capable of intercepting particle k and then transitioning back to $\mathcal{F}$ so long as a bound is established on the length of our deployment scheduling window $\frac{T^\star}{N}$.

Theorem 2: If agent power lifespan $T^\star$ satisfies $\frac{T^\star}{N} \ge t_{ck} - t_{dk'} + \frac{\pi}{2U_{max}}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g,\ \forall k$ then the agent with $i_p = 1$ shall always be capable of reaching $\mathcal{F}$ within $\frac{T^\star}{N}$ of the time at which $i_p := 1$.

Proof: Consider the worst case scenario in which the agent with $i_p = 2$ is assigned to intercept particle k at the instant before $i_p := 1$. It's remaining flight time is currently $\frac{T^\star}{N}$. The time required to intercept the particle is $t_{ck} - t_{dk'}$, after which our control strategy dictates that the agent will follow a geodesic trajectory to $\mathcal{F}$. As $\mathcal{F}$ lies at the north pole of $\mathcal{C}'$, this will be a trajectory of constant longitude which may be upper bounded by a length half the perimeter of our revolved ellipsoid: $\frac{\pi}{2}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g$ by definition. As the agent is controlled by (6) with a North-pointing $\hat{\nu}_i$, it will proceed along this geodesic at speed U_{max}. Thus the time required to complete this trajectory is $\frac{\pi}{2U_{max}}\left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right) g$ and we may bound our deployment window: $\frac{T^\star}{N} \ge t_{ck} -$

$t_{dk'} + \frac{\pi}{2U_{max}} (x_{\mathcal{C}',r} + z_{\mathcal{C}',r})\, g,\ \forall k$. This concludes the proof. ■

In summary, the appropriate design method for this surveillance system is to first ensure that the time from detection to collision of any arbitrary particle, $t_{ck} - t_{dk'}$, as governed by the omnidirectional range sensor satisfies *Theorem* 1. One must subsequently ensure that power lifespan $T^\star$, for all agents, satisfies *Theorem* 2.

B. Partition Transfer and Return to Base

If an agent with $i_p \in \{2, ..., N\}$ lies outside of its prescribed partition, and we have $f_i = 0$, then the agent shall enter partition transfer mode. This mode uses the same geodesic position and orientation controllers (6) and (13) with the destination position set to the point:

$$[x_{id}\ \ y_{id}\ \ z_{id}]^T = \begin{bmatrix} x_{\mathcal{C}',r} \cos\left(\arcsin\left(\frac{z_{id}}{z_{\mathcal{C}',r}}\right)\right) \cos\Big(\text{atan2}\big(y_i(t), x_i(t)\big)\Big) \\ y_{\mathcal{C}',r} \cos\left(\arcsin\left(\frac{z_{id}}{z_{\mathcal{C}',r}}\right)\right) \sin\Big(\text{atan2}\big(y_i(t), x_i(t)\big)\Big) \\ \bar{z}_{i_p-1}, \text{if } z_i < \bar{z}_{i_p-1}; \text{or } \bar{z}_{i_p-2}, \text{if } z_i > \bar{z}_{i_p-2} \end{bmatrix},$$

i.e., the closest point along the agent's current longitude which lies on the boundary of its assigned partition.

The return to base mode is similar to partition transfer mode but is defined for the agent with $i_p = 1$. This mode is activated when the time since agent i's last deployment from $\mathcal{F}$, denoted $t_{i\mathcal{F}} \geq T^\star - \frac{\pi}{2U_{max}} (x_{\mathcal{C}',r} + z_{\mathcal{C}',r})\, g$ as established in *Theorem* 2. The control strategy is the same as partition transfer mode with the desired position set to $\mathcal{F}$. Control laws for partition transfer mode and return to base shall be denoted with superscripts ptm and rtb respectively.

V. Collision Avoidance

To encode collision avoidance, we have designed an explicit avoidance mode which may be transitioned into from local coverage, partition transfer, or particle intercept mode. This mode is triggered for agent i when we have the condition that $\|p_i - p_j\| \leq R_i$ for $i \neq j$. Denote $\tilde{j} = i \cup j$ as the set of agents satisfying this condition. Agents in $\tilde{j}$ are ranked by $t_{\tilde{j}\mathcal{F}}$. One agent, denoted i_{pr}, whose value for $t_{\tilde{j}\mathcal{F}}$ is highest, i.e., $i_{pr} = \text{argmax}_{\tilde{j}} \left(t_{\tilde{j}\mathcal{F}}\right)$ is permitted to proceed while the remaining agents transition to collision avoidance mode. Agents in $\tilde{j} \setminus i_{pr}$, are controlled to follow $\hat{n}_i$ until they have reached a height R_i above $\mathcal{C}'$, i.e., they ascend to and hover about points on an ellipsoid concentric with $\mathcal{C}'$ whose semi-major and semi-minor axes are a factor of R_i longer than those of $\mathcal{C}'$. They remain in this hover configuration until i_{pr} passes underneath the hovering agents along its nominal trajectory and removes itself from the deadlock. At this point, a new i_{pr} is selected and that agent will descend to $\mathcal{C}'$ and return to its previous hybrid state.

The avoidance position control strategy is:

$$[u_i^{av}\ v_i^{av}\ w_i^{av}]^T = U_{max} \mathcal{R}_1^{-1} \hat{n}_i. \tag{18}$$

As the agents shall ascent to a point at which R_i does not intersect $\mathcal{C}$, sensing information is not gathered in avoidance mode and thus the avoidance orientation control is simply $[q_i^{av}\ \ r_i^{av}\ \ s_i]^T = [0\ \ 0\ \ 0]$. With an additional assumption on agent size, we may establish a collision avoidance guarantee.

Theorem 3: For agents $\{i, j\} \in \tilde{j}$, the condition that $\min(R_{\tilde{j}}) > 2\,\text{ż}_i + 2\,\text{ż}_j$, implies that i shall not collide with j.

Proof: Consider the case in which $i \neq i_{pr}$ and $j \neq i_{pr}$. Both agents operate in accordance with (18) and the agents follow trajectories along $\hat{n}_i$ and $\hat{n}_j$ respectively. Both unit vectors are normal to surface $\mathcal{C}$, an ellipsoid of revolution, and thus diverge from one another away from $\mathcal{C}$. Agents i and j shall enter avoidance mode at an instant when $\|p_i - p_j\| \geq \min\left(R_{\tilde{j}}\right)$ and their distance shall tend to increase under (18). Thus $\min(R_{\tilde{j}}) > \text{ż}_i + \text{ż}_j$ and subsequently $\min(R_{\tilde{j}}) > 2\,\text{ż}_i + 2\,\text{ż}_j$ imply that they avoid collision.

Consider the case in which $i = i_{pr}$ and thus $j \neq i_{pr}$. In the instant that j transitions to avoidance mode we have that $\|p_i - p_j\| \geq \min\left(R_{\tilde{j}}\right)$. Thus the distance for i to travel until collision is greater than or equal to $\min\left(R_{\tilde{j}}\right) - \text{ż}_i - \text{ż}_j$. This straight line path for i is a conservative estimate as the true path is curved. Collision will be avoided if agent j, whose path is normal to the surface, may cover a distance $\text{ż}_i + \text{ż}_j$ before i covers $\min\left(R_{\tilde{j}}\right) - \text{ż}_i - \text{ż}_j$. As j moves at speed U_{max} and i's tangential speed is upper bounded by U_{max}, this condition is satisfied if $\min\left(R_{\tilde{j}}\right) - \text{ż}_i - \text{ż}_j > \text{ż}_i + \text{ż}_j$. This may equivalently be written as $\min\left(R_{\tilde{j}}\right) > 2\,\text{ż}_i + 2\,\text{ż}_j$. These arguments apply to the case in which $j = i_{pr}$ and $i \neq i_{pr}$ as well. This concludes the proof. ■

VI. Hybrid Formulation

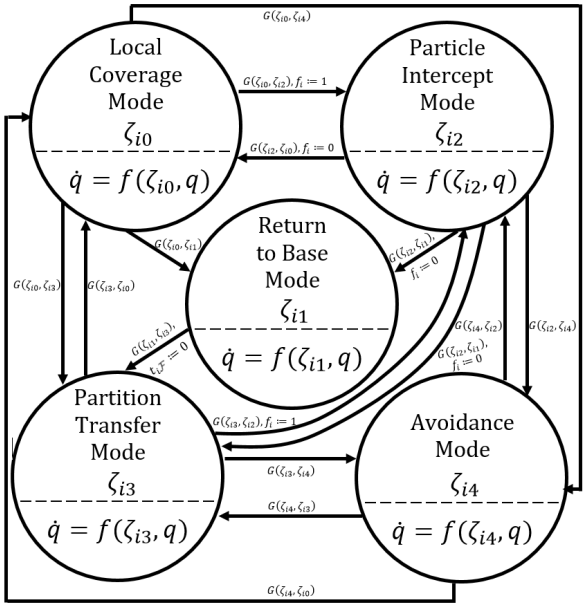

Fig. 3. Agent i operates in accordance with this automaton. For clarity, elements of the reset map are shown explicitly.

To provide a compact notation in this section, define $\acute{f}_i = \frac{x_i^2}{(x_{\mathcal{C},r} + \text{ż}_i)^2} + \frac{y_i^2}{(y_{\mathcal{C},r} + \text{ż}_i)^2} + \frac{z_i^2}{(z_{\mathcal{C},r} + \text{ż}_i)^2}$. The coverage strategy for agent i is represented by the hybrid automaton in Fig. 3, described by the following entities [26]:

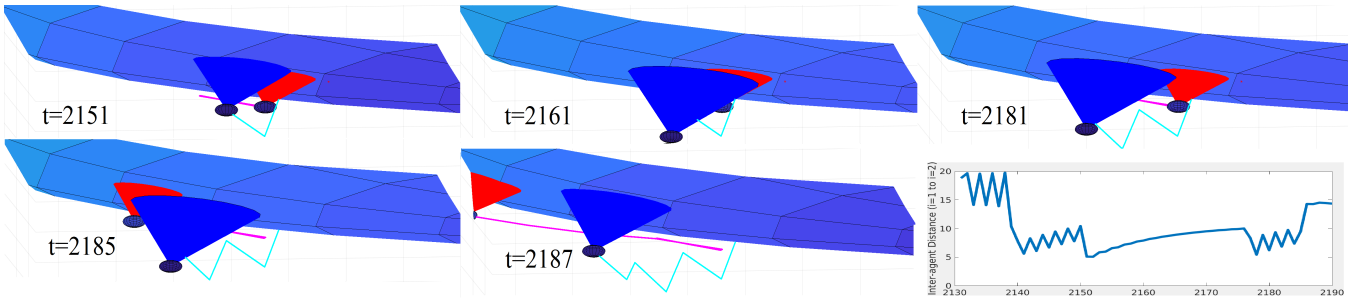

Fig. 4. Agent $i = 1$, indicated with blue $\mathcal{S}_i$, is at a distance less than $R_i = 10$ of agent $i = 2$, indicated with red $\mathcal{S}_i$, at $t = 2151$. Agent $i = 1$ enters avoidance mode and follows a trajectory which oscillates about an altitude of $(\gamma + 1) R_i$ with respect to the surface while $i = 2$ follows a nominal trajectory about the altitude of γR_i. This allows for $i = 2$ to follow its geodesic trajectory close to the surface while avoiding collision with $i = 1$. Note that $\gamma = 0.5$ and $i = 1$ and $i = 2$ trajectories are indicated in cyan and magenta respectively. Also note the height differential in trajectories at $t = 2187$.

- A set of discrete states: $Z_i = \{\zeta_{i0}, \zeta_{i1}, \zeta_{i2}, \zeta_{i3}, \zeta_{i4}\}$,
- A set of continuous states: $\eta_i = \{x_i, y_i, z_i, \Phi_i, \Theta_i, \Psi_i\}$,
- A vector field:
  $f(\zeta_{i0}, \eta_i) = \mathcal{R} \left[u_i^{loc} \; v_i^{loc} \; w_i^{loc} \; 0 \; r_i^{loc} \; s_i^{loc} \right]^T$,
  $f(\zeta_{i1}, \eta_i) = \mathcal{R} \left[u_i^{rtb} \; v_i^{rtb} \; w_i^{rtb} \; q_i^{rtb} \; r_i^{rtb} \; s_i^{rtb} \right]^T$,
  $f(\zeta_{i2}, \eta_i) = \mathcal{R} \left[u_i^{pim} \; v_i^{pim} \; w_i^{pim} \; q_i^{pim} \; r_i^{pim} \; s_i^{pim} \right]^T$,
  $f(\zeta_{i3}, \eta_i) = \mathcal{R} \left[u_i^{ptm} \; v_i^{ptm} \; w_i^{ptm} \; q_i^{ptm} \; r_i^{ptm} \; s_i^{ptm} \right]^T$,
  $f(\zeta_{i4}, \eta_i) = \mathcal{R} \left[u_i^{av} \; v_i^{av} \; w_i^{av} \; 0 \; 0 \; 0 \right]^T$ where $\mathcal{R} = \begin{bmatrix} \mathcal{R}_1 & 0 \\ 0 & \mathcal{R}_2 \end{bmatrix}$,
- A set of initial states: $\{\zeta_{i3}\} \times \{\eta_i \in \mathbb{R}^6 \mid p_i = \mathcal{F} \wedge \Phi_i \in [-\pi, +\pi] \wedge \Theta_i \in \left[\frac{-\pi}{2}, \frac{+\pi}{2}\right] \wedge \Psi_i \in [-\pi, +\pi]\}$,
- A domain: $Dom\left(\zeta_{i0}\right) = \{\eta_i \in \mathbb{R}^6 \mid \acute{f}_i \geq 1 \wedge \left(i_p \in \{2, ..., N\} \implies \bar{z}_{i_p - 1} \leq z_i \leq \bar{z}_{i_p - 2}\right)\}$,
  $Dom\left(\zeta_{i1}\right) = \{\eta_i \in \mathbb{R}^6 \mid \acute{f}_i \geq 1\}$,
  $Dom\left(\zeta_{i2}\right) = \{\eta_i \in \mathbb{R}^6 \mid \acute{f}_i \geq 1\}$,
  $Dom\left(\zeta_{i3}\right) = \{\eta_i \in \mathbb{R}^6 \mid \acute{f}_i \geq 1 \wedge \left(i_p \in \{2, ..., N\} \implies z_i < \bar{z}_{i_p - 1} \vee z_i > \bar{z}_{i_p - 2}\right)\}$,
  $Dom\left(\zeta_{i4}\right) = \{\eta_i \in \mathbb{R}^6 \mid \acute{f}_i \geq 1\}$,
- A set of edges: $E = \{\left(\zeta_{i0}, \zeta_{i1}\right), \left(\zeta_{i0}, \zeta_{i2}\right), \left(\zeta_{i0}, \zeta_{i3}\right), \left(\zeta_{i0}, \zeta_{i4}\right), \left(\zeta_{i1}, \zeta_{i3}\right), \left(\zeta_{i2}, \zeta_{i0}\right), \left(\zeta_{i2}, \zeta_{i1}\right), \left(\zeta_{i2}, \zeta_{i3}\right), \left(\zeta_{i2}, \zeta_{i4}\right), \left(\zeta_{i3}, \zeta_{i0}\right), \left(\zeta_{i3}, \zeta_{i2}\right), \left(\zeta_{i3}, \zeta_{i4}\right), \left(\zeta_{i4}, \zeta_{i0}\right), \left(\zeta_{i4}, \zeta_{i2}\right), \left(\zeta_{i4}, \zeta_{i3}\right), \}$,
- A set of guard conditions:
  $G\left(\zeta_{i0}, \zeta_{i1}\right) = \{i_p = 1 \wedge t_{i\mathcal{F}} \geq T^\star - \frac{\pi g}{2U_{max}} \left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right)\}$,
  $G\left(\zeta_{i0}, \zeta_{i2}\right) = \{\exists k \mid i = i_k\}$,
  $G\left(\zeta_{i0}, \zeta_{i3}\right) = \{i_p \neq 1 \wedge \left(z_i < \bar{z}_{i_p - 1} \vee z_i > \bar{z}_{i_p - 2}\right)\}$,
  $G\left(\zeta_{i0}, \zeta_{i4}\right) = \{\|p_i - p_j\| \leq R_i \wedge i_{pr} \neq \text{argmax}_{\tilde{j}} \left(t_{\tilde{j}\mathcal{F}}\right)\}$,
  $G\left(\zeta_{i1}, \zeta_{i3}\right) = \{\|p_i - \mathcal{F}\| \leq \varepsilon_1 \wedge t_{i\mathcal{F}} = T^\star\}$,
  $G\left(\zeta_{i2}, \zeta_{i0}\right) = \{\|p_i - \tilde{p}'_k(t_{ck})\| \leq \varepsilon_1 \wedge t \geq t_{ck} \wedge \left(\left(i_p \in \{2, ..., N\} \wedge \bar{z}_{i_p - 1} \leq z_i \leq \bar{z}_{i_p - 2}\right) \vee \left(i_p = 1 \wedge t_{i\mathcal{F}} < T^\star - \frac{\pi g}{2U_{max}} \left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right)\right)\right)\}$,
  $G\left(\zeta_{i2}, \zeta_{i1}\right) = \{\|p_i - \tilde{p}'_k(t_{ck})\| \leq \varepsilon_1 \wedge t \geq t_{ck} \wedge i_p = 1 \wedge t_{i\mathcal{F}} \geq T^\star - \frac{\pi g}{2U_{max}} \left(x_{\mathcal{C}',r} + z_{\mathcal{C}',r}\right)\}$
  $G\left(\zeta_{i2}, \zeta_{i3}\right) = \{\|p_i - \tilde{p}'_k(t_{ck})\| \leq \varepsilon_1 \wedge t \geq t_{ck} \wedge i_p \in \{2, ..., N\} \wedge \left(z_i < \bar{z}_{i_p - 1} \vee z_i > \bar{z}_{i_p - 2}\right)\}$,
  $G\left(\zeta_{i2}, \zeta_{i4}\right) = G\left(\zeta_{i0}, \zeta_{i4}\right)$,
  $G\left(\zeta_{i3}, \zeta_{i0}\right) = \{i_p = 1 \vee \left(i_p \neq 1 \wedge \bar{z}_{i_p - 1} \leq z_i \leq \bar{z}_{i_p - 2}\right)\}$,
  $G\left(\zeta_{i3}, \zeta_{i2}\right) = G\left(\zeta_{i0}, \zeta_{i2}\right)$,
  $G\left(\zeta_{i3}, \zeta_{i4}\right) = G\left(\zeta_{i0}, \zeta_{i4}\right)$,
  $G\left(\zeta_{i4}, \zeta_{i0}\right) = \{\left(\|p_i - p_j\| > R_i, \forall j \vee i_{pr} = \text{argmax}_{\tilde{j}} \left(t_{\tilde{j}\mathcal{F}}\right)\right) \wedge \left(i_p = 1 \vee \left(i_p \in \{2, ..., N\} \wedge \bar{z}_{i_p - 1} \leq z_i \leq \bar{z}_{i_p - 2}\right)\right)\}$,
  $G\left(\zeta_{i4}, \zeta_{i2}\right) = \{\left(\|p_i - p_j\| > R_i, \forall j \vee i_{pr} = \text{argmax}_{\tilde{j}} \left(t_{\tilde{j}\mathcal{F}}\right)\right) \wedge f_i = 1\}$,
  $G\left(\zeta_{i4}, \zeta_{i3}\right) = \{\left(\|p_i - p_j\| > R_i, \forall j \vee i_{pr} = \text{argmax}_{\tilde{j}} \left(t_{\tilde{j}\mathcal{F}}\right)\right) \wedge \left(i_p \neq 1 \wedge \left(z_i < \bar{z}_{i_p - 1} \vee z_i > \bar{z}_{i_p - 2}\right)\right)\}$.
- Additional parameters include a clock set: $C = \{t_{i\mathcal{F}}\}$, a flag: $f_i \in \{0, 1\}$ and,
- A reset map: $R\left(\zeta_{i0}, \zeta_{i2}, f_i\right) = \{1\}, R\left(\zeta_{i1}, \zeta_{i3}, t_{i\mathcal{F}}\right) = \{0\}, R\left(\zeta_{i2}, \zeta_{i0}, f_i\right) = \{0\}, R\left(\zeta_{i2}, \zeta_{i1}, f_i\right) = \{0\}$, $R\left(\zeta_{i2}, \zeta_{i3}, f_i\right) = \{0\}, R\left(\zeta_{i3}, \zeta_{i2}, f_i\right) = \{1\}$, and continuous states do not reset between transitions.

One should note that Zeno oscillations are avoided in $G\left(\zeta_{i0}, \zeta_{i3}\right)$ and $G\left(\zeta_{i3}, \zeta_{i0}\right)$ as (17) implies that the local coverage control laws will tend to pull $\mathcal{S}_i$ inside of the assigned partition from the boundary.

VII. Simulations

A simulation was performed in MATLAB to verify the efficacy of the algorithm. Four agents are deployed to cover the surface of an ellipsoid, $\mathcal{C}$, whose radius in the xy-plane is 80 and whose radius in the z-plane is 20. For each agent, $R_i = 10$, $\mathbf{z}_i = 1$, $\alpha_i = 30°$, $k_u = 1$, $k_v = 5$, $k_w = 1$, $k_r = 0.1$, $k_s = 0.1$, $\bar{r}_i = 0.4$, $\bar{s}_i = 0.4$. Upon initialization of the simulation, $\mathcal{C}$ was set to a fully covered level of $C^\star = 20$ which would begin decaying upon detection of the first particle $k \in \{1, ..., 4\}$ at $t = 600$ sec. Particles, which travelled in random directions at a speed of 1 distance unit per second, were generated every 25 seconds.

Agents were able to successfully intercept nearly all particles along their geodesic trajectories while actively avoiding collision (see Fig. 4); however, agents spent the majority of their time in particle intercept mode hovering near predicted impact points due to the high rate of particle generation (see Fig. 5). This negatively impacted their ability to explore away from the site of impacts thus contributing to a slow but continued growth in the coverage error. In fact, Fig. 6 illustrates precisely that only the power critical agent tended to spend any time in local coverage mode. This always preceded its return to base. Future simulations will test longer time trials under the same particle generation rate to determine if the coverage error growth tapers off. Additional simulations will test less frequent particle generation to increase the time agents spend in local coverage mode.

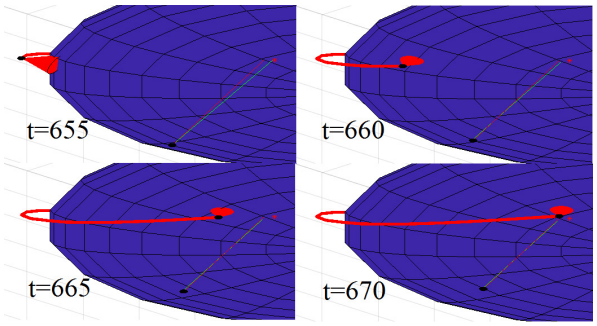

Fig. 5. Agent $i = 2$ follows its geodesic trajectory to the predicted impact point of particle k. The true trajectory of the particle is indicated in red and the estimated trajectory in green.

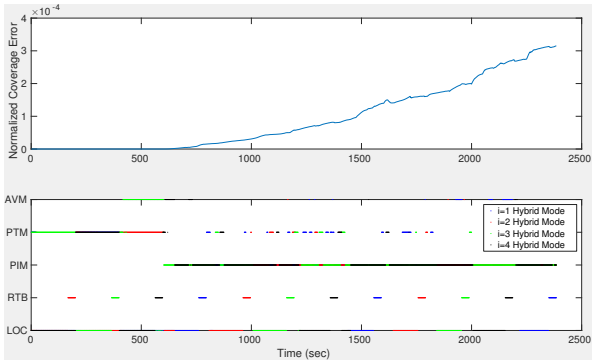

Fig. 6. The Normalized coverage error remains on the order of 10^{-4}. Agent operating modes are indicated over time with the abbreviations from top to bottom referring to avoidance mode, partition transfer mode, particle intercept mode, return to base, and local coverage mode.

VIII. Conclusions

In this paper, we presented a hybrid formulation for the persistent coverage problem in an environment subject to stochastic intruders. This formulation was motivated in part by extravehicular applications of the NASA Mini AERCam. Agents operated with finite power resources and were required to periodically return to a refueling station while patrolling assigned latitude partitions along the surface of an ellipsoid. The efficacy of the algorithm was demonstrated in simulation.

References

[1] R. N. Smith, M. Schwager, S. L. Smith, B. H. Jones, D. Rus, and G. S. Sukhatme, "Persistent ocean monitoring with underwater gliders: Adapting sampling resolution," *Journal of Field Robotics*, vol. 28, no. 5, pp. 714–741, 2011.

[2] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessell, M. Rutten, and S. Jha, "Wireless sensor networks for battlefield surveillance," in *Proceedings of the land warfare conference*, 2006, pp. 1–8.

[3] R. R. Murphy, S. Tadokoro, D. Nardi, A. Jacoff, P. Fiorini, H. Choset, and A. M. Erkmen, "Search and rescue robotics," in *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008, pp. 1151–1173.

[4] H. Choset and D. Kortenkamp, "Path planning and control for free-flying inspection robot in space," *Journal of Aerospace Engineering*, vol. 12, no. 2, pp. 74–81, 1999.

[5] G. A. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme, "Active planning for underwater inspection and the benefit of adaptivity," *The International Journal of Robotics Research*, vol. 32, no. 1, pp. 3–18, 2013.

[6] S. E. Fredrickson, S. Duran, N. Howard, and J. D. Wagenknecht, "Application of the mini AERcam free flyer for orbital inspection," in *Defense and Security*. International Society for Optics and Photonics, 2004, pp. 26–35.

[7] J. Cortes, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Trans. on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[8] I. I. Hussein and D. M. Stipanović, "Effective coverage control for mobile sensor networks with guaranteed collision avoidance," *IEEE Trans. on Control Systems Technology*, vol. 15, no. 4, pp. 642–657, Jul. 2007.

[9] B. Liu, O. Dousse, P. Nain, and D. Towsley, "Dynamic coverage of mobile sensor networks," *IEEE Transactions on Parallel and Distributed systems*, vol. 24, no. 2, pp. 301–311, 2013.

[10] D. M. Stipanović, C. Valicka, C. J. Tomlin, and T. R. Bewley, "Safe and reliable coverage control," *Numerical Algebra, Control and Optimization*, vol. 3, pp. 31–48, 2013.

[11] P. Hokayem, D. Stipanovíc, and M. Spong, "On persistent coverage control," in *Proc. of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, USA, Dec. 2007, pp. 6130–6135.

[12] C. Song, L. Liu, G. Feng, Y. Wang, and Q. Gao, "Persistent awareness coverage control for mobile sensor networks," *Automatica*, vol. 49, no. 6, pp. 1867–1873, 2013.

[13] S. L. Smith, M. Schwager, and D. Rus, "Persistent robotic tasks: Monitoring and sweeping in changing environments," *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 410–426, 2012.

[14] J. M. Palacios-Gasós, E. Montijano, C. Sagüés, and S. Llorente, "Multi-robot persistent coverage with optimal times," in *Proc. of the 55th IEEE Conference on Decision and Control*. IEEE, 2016, pp. 3511–3517.

[15] J. M. Palacios-Gasós, E. Montijano, C. Sagues, and S. Llorente, "Multi-robot persistent coverage using branch and bound," in *Proc. of the 2016 American Control Conference*, 2016, pp. 5697–5702.

[16] J. M. Palacios-Gasós, Z. Talebpour, E. Montijano, C. Sagüés, and A. Martinoli, "Optimal path planning and coverage control for multi-robot persistent coverage in environments with obstacles," in *Proc. of the 2017 IEEE International Conference on Robotics and Automation*, 2017, pp. 1321–1327.

[17] D. Mitchell, M. Corah, N. Chakraborty, K. Sycara, and N. Michael, "Multi-robot long-term persistent coverage with fuel constrained robots," in *Proc. of the 2015 IEEE International Conference on Robotics and Automation*. IEEE, 2015, pp. 1093–1099.

[18] P. Cheng, J. Keller, and V. Kumar, "Time-optimal UAV trajectory planning for 3D urban structure coverage," in *Proc. of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, Sep. 2008, pp. 2750–2757.

[19] J. Yu, S. Karaman, and D. Rus, "Persistent monitoring of events with stochastic arrivals at multiple stations," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 521–535, 2015.

[20] F. Pasqualetti, F. Zanella, J. R. Peters, M. Spindler, R. Carli, and F. Bullo, "Camera network coordination for intruder detection," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 5, pp. 1669–1683, 2014.

[21] W. Bentz and D. Panagou, "Persistent coverage of a two-dimensional manifold subject to time-varying disturbances," in *Proc. of the 56th IEEE Conference on Decision and Control, accepted*, Melbourne, Australia, Dec. 2017. [Online]. Available: http://www-personal.umich.edu/~dpanagou/assets/documents/WBentz_CDC17.pdf

[22] R. W. Beard, "Quadrotor dynamics and control," 2008, lecture notes. [Online]. Available: http://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=2324&context=facpub

[23] T. Vincenty, "Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations," *Survey review*, vol. 23, no. 176, pp. 88–93, 1975.

[24] ——, "Geodetic inverse solution between antipodal points," Aug. 1975, Scanned by Charles Karney from the copy in R.H. Rapp's library at Ohio State University. The report is a work of the U.S. Government and so is in the public domain. [Online]. Available: https://doi.org/10.5281/zenodo.32999

[25] J. Ivory, "VIII. A new series for the rectification of the ellipsis; together with some observations on the evolution of the formula $(a^2 + b^2 - 2ab\cos\theta)^n$," *Transactions of the Royal Society of Edinburgh*, vol. 4, no. 2, p. 177190, 1798.

[26] J. Lygeros, "Lecture notes on hybrid systems," 2004. [Online]. Available: https://robotics.eecs.berkeley.edu/~sastry/ee291e/lygeros.pdf