51st IEEE Conference on Decision and Control
December 10-13, 2012. Maui, Hawaii, USA

A Static Coverage Algorithm for Locational Optimization

George Mathew[1] and Amit Surana[2]

Abstract— **We propose multiscale metrics to capture the quality of coverage by a static configuration of agents. This metric is used for the locational optimization of sensor networks. Agent configurations that minimize the multiscale coverage metric are an alternative to the well-known *centroidal voronoi tesselations*. Other applications include quantization and clustering analysis. We demonstrate the performance of the algorithm on various examples.**

I. Introduction

Sensor deployment and placement is an important problem in many applications like surveillance and environmental monitoring. For surveillance problems, it is desirable to have the sensors uniformly cover an area under surveillance so that a threat (or target) can be detected with a high probability. For environmental monitoring problems, it is desirable to have the sensors uniformly placed so that various spatially distributed elements like temperature and pressure fields can be monitored with a high degree of resolution. Sensor placement problems for these sort of applications fall under the subject of locational optimization. Locational optimization has been studied under a variety of contexts from the spatial distribution of resources to the territorial behavior of animals (see [1] and [2]). Locational optimization also plays a role in applications like quantization and clustering analysis.

The term 'coverage' can have slightly different interpretations. We differentiate between 'static' coverage and 'dynamic' coverage. Static coverage problems refer to problems where we are interested in finding a stationary configuration of sensors that are optimal in terms of the quality of service they provide. Dynamic coverage problems refer more to the path-planning problem for mobile agents so that they explore an area completely. The focus of this paper is on 'static' coverage. The 'dynamic' coverage problem has been addressed in [3] and [4].

The Lloyd algorithm [5] is a classical approach to the locational optimization problem. The Lloyd algorithm is an approach to generate centroidal voronoid tesselations (CVTs). See [1] for an extensive review on the applications and analysis of CVTs. Also see [6] and [7] for a review on the application of Voronoi-based methods for coverage control of sensor networks. For distributed and asynchronous implementations of the Lloyd algorithm see [8] and [9].

In this paper, we borrow ideas from our previous work in [3], [4], [10] and [11] to capture the quality of coverage by a static configuration of agents . We use the simple idea that, for uniform coverage, the fraction of the number of agents in any set must be proportional to the probability of detecting an event (or finding a target) in that set. This concept of uniform coverage can be related to concepts in sampling theory where discrepancy measures are used to check whether a finite number of samples are a good representation of a probability distribution [12].

The rest of the paper is structured as follows. In Section II, we discuss the metrics used to quantify the quality of coverage of a network of agents. In Section III, we talk about configurations that minimize the coverage metric and the corresponding necessary conditions for optimality. In Section IV, we talk about a continuous time descent law for sensor deployment and show its application to a variety of examples. In Section V, we extend the coverage control algorithm for agents with Dubins vehicle dynamics. In Section VI, we demonstrate how the same coverage algorithm can be effectively used for uniform placement of agents on the boundary of an environment. In Section VI, we conclude with a discussion of some open issues related to the proposed coverage algorithm.

II. Multiscale Coverage Metrics

The agent positions are given as $x_j \in \mathbb{R}^n$ for $j = 1,2,...,N$, where n is the dimension of the space. In this paper, we focus on two-dimensional coverage problems (i.e., $n = 2$). The set of agent locations is denoted by $\{x_j\} = \{x_j :\ \text{for}\ j = 1,2,...,N\}$. The objective is to sample (or cover) a probability distribution μ with the N available agents. We assume that μ is zero outside a rectangular domain $U \subset \mathbb{R}^n$. Note that there are no restrictions on the shape of the support of μ within U. First, we need to define appropriate metrics to capture the quality of coverage by an agent configuration $\{x_j\}$. The locational optimization problem is solved by optimizing for these coverage metrics.

A. Coverage metric based on spherical integrals

Let $B(y,r)$ be the spherical set of radius r with center at the location y. i.e., $B(y,r) = \{z : \|z-y\| \le r\}$. Let $d(y,r)$ be the fraction of the number of agents in the set $B(y,r)$.

$$d(y,r) = \frac{1}{N}\sum_{j=1}^{N} \chi_{B(y,r)}(x_j), \qquad (1)$$

where $\chi_{B(y,r)}$ is the indicator function on the set $B(y,r)$. For spheres $B(y,r)$ that lie entirely within the rectangular domain U, the fraction $d(y,r)$ is computed as in (1). For spheres $B(y,r)$ that do not lie entirely within the domain U, $d(y,r)$ is computed as if each agent has a mirror image

*This work was supported by United Technologies Research Center

[1]G. Mathew is with United Technologies Research Center, Berkeley, CA. mathewga@utrc.utc.com

[2]A. Surana is with United Technologies Research Center, East Hartford, CT. suranaa@utrc.utc.com

about the boundaries of the domain U. Equivalently $d(y,r)$ is computed as the spherical integral of the even extension of the distribution $C(\{x_j\})$ defined in (4). The measure of the set $B(y,r)$ is given as

$$\bar{\mu}(y,r) = \int_{B(y,r)} \mu(z)dz. \tag{2}$$

The quantity $\bar{\mu}(y,r)$ can be interpreted as the probability of detecting an event in the set $B(y,r)$. For uniform coverage, we require that $d(y,r) \approx \bar{\mu}(y,r)$ for all points $y \in U$ and for all radii r. This motivates defining the following metric for uniform coverage:

$$E^2(\{x_j\}) = \int_0^R \int_U (d(y,r) - \bar{\mu}(y,r))^2 dydr, \text{ where } R > 0. \tag{3}$$

This metric is difficult to numerically compute as it involves the computation of spherical integrals at all scales. But it turns out that this metric is equivalent to another metric based on Fourier coefficients which is much easier to compute. This metric is discussed in the next subsection.

B. *Coverage metric based on Fourier coefficients*

For convenience, we use a *coverage distribution* defined as

$$C(\{x_j\}) = \frac{1}{N}\sum_{j=1}^{N} \delta_{x_j}, \tag{4}$$

where δ_{x_j} is the Dirac delta distribution with support at the location x_j. Note that the distribution $C(\{x_j\})$ can be thought of as a probability distribution. Also note that we have

$$d(y,r) = \langle C(\{x_j\}), \chi_{B(y,r)} \rangle. \tag{5}$$

Let $\{f_k\}$ be the Fourier basis functions that satisfy Neumann boundary conditions on the rectangular domain U and k is the corresponding wave-number vector. For instance, on a rectangular domain $U = [0,L_1] \times [0,L_2]$, we have,

$$\begin{aligned} & f_k(x) = \frac{1}{h_k} cos(k_1x_1)cos(k_2x_2), \text{where} \\ & \quad k_1 = \frac{K_1\pi}{L_1} \text{ and } k_2 = \frac{K_2\pi}{L_2}, \\ & \quad \text{for } K_1, K_2 = 0,1,2.... \text{ and where} \\ & \quad h_k = \left(\int_0^{L_1}\int_0^{L_2} cos^2(k_1x_1)cos^2(k_2x_2)dx_1dx_2 \right)^{1/2}. \end{aligned} \tag{6}$$

The division by the factor h_k ensures that f_k has L^2 norm equal to one. Therefore f_k is an orthonormal basis. The Fourier coefficients of the *coverage distribution* are given as

$$c_k = \langle C(\{x_j\}), f_k \rangle = \frac{1}{N}\sum_{j=1}^{N} f_k(x_j). \tag{7}$$

Also, the Fourier coefficients of the probability distribution μ are written as:

$$\mu_k = \langle \mu, f_k \rangle, \tag{8}$$

where $< .,. >$ is the standard inner product. We use the Sobolev space norm of negative index (H^{-s}, for $s = \frac{(n+1)}{2}$ and where n is the dimension of the space) to measure the distance between the coverage distribution $C(\{x_j\})$ and the probability distribution μ. This is given as

$$\begin{aligned} & \phi^2(\{x_j\}) = \|C(\{x_j\}) - \mu\|_{H^{-s}} = \sum_{K \in \mathbb{Z}^{*n}} \Lambda_k |s_k|^2, \\ & \quad \text{where } s_k = c_k - \mu_k, \quad \Lambda_k = \frac{1}{(1+\|k\|^2)^s} \\ & \quad \text{and } \mathbb{Z}^{*n} = [0,1,2,......]^n. \end{aligned} \tag{9}$$

The quantity c_k can be thought of as the sample average of the Fourier basis function f_k over the agent positions. The quantity μ_k is the spatial average of the Fourier basis function f_k with respect to the probability measure μ. The metric ϕ is a weighted sum of the discrepancy between the sample averages and the true spatial averages of the Fourier basis functions over all wavenumbers, but with more weight given to the small wavenumbers (or large scale modes). The two metrics E and ϕ described before are equivalent. i.e., there exist bounded constants c_1, c_2 such that

$$c_1\phi^2(\{x_j\}) \leq E^2(\{x_j\}) \leq c_2\phi^2(\{x_j\}). \tag{10}$$

The proof for the equivalence of these two metrics goes along the same lines as described in [4]. Since the metric ϕ is much easier to compute, we will use the metric ϕ for locational optimization in the rest of the paper. Also the metric ϕ will be referred to as the *Static Spectral Multiscale Coverage (Static SMC)* metric. The qualifier term *Static* in *Static SMC* is used to distinguish it from the SMC metric that was used to capture the quality of dynamic coverage by a network of agents ([3] and [4]).

III. Static SMC configurations

The critical points of the metric ϕ will be referred to as *Static SMC Configurations*. For convenience we define the cost-function $\Phi(\{x_j\}) = \frac{1}{2}\phi^2(\{x_j\})$. Obviously Φ and ϕ have the same critical points. Also note that the cost-function Φ is a non-convex function of the agent locations x_j. The gradient of the cost-function Φ with respect to the agent locations are given as:

$$B_j(\{x_j\}) = \frac{\partial \Phi}{\partial x_j} = \frac{1}{N}\sum_K \Lambda_k s_k \nabla f_k(x_j). \tag{11}$$

Here, $\nabla f_k(x_j)$ is the gradient of the Fourier basis function evaluated at the agent location x_j. For instance, for the basis functions in (6), we have

$$\nabla f_k(x) = \frac{1}{h_k}\begin{bmatrix} -k_1 sin(k_1x_1)cos(k_2x_2) \\ -k_2 cos(k_1x_1)sin(k_2x_2) \end{bmatrix}. \tag{12}$$

Clearly, the gradient of the cost-function Φ must be zero at the critical points. Therefore the *Static SMC configurations* $\{x_j^*\}$ should satisfy the following set of nonlinear equations:

$$\sum_K \Lambda_k s_k \nabla f_k(x_j^*) = 0, \text{ for all } j = 1,2,...,N, \tag{13}$$

or

$$B_j(\{x_j^*\}) = 0, \text{ for all } j = 1,2,...,N. \tag{14}$$

Clearly, the Static SMC configurations depend on the probability distribution μ. Also there could be multiple Static SMC configurations for the same μ.

IV. Gradient Descent Algorithm for Coverage Control

In this section, we describe how the Static SMC configurations can be computed using a simple gradient descent algorithm. We discuss only centralized implementations of coverage algorithms in this paper. The algorithm referred to as *Static SMC* can be used to deploy agents so that they converge to one of the Static SMC configurations. Let the agents have first-order dynamical behavior described by

$$\dot{x}_j(t) = u_j(t). \tag{15}$$

Computing the time-derivative of the coverage measure Φ, we get

$$\dot{\Phi}(\{x_j\}) = \sum_{j=1}^{N} B_j(\{x_j(t)\}) \cdot u_j(t) \tag{16}$$

We consider the coverage measure Φ as a Lyapunov function and set the control to be

$$u_j(t) = -K_f B_j(\{x_j(t)\}), \tag{17}$$

where $K_f > 0$ is a feedback gain. With this choice of control, the time-derivative of the coverage measure Φ is guaranteed to be non-positive. In particular

$$\dot{\Phi}(\{x_j\}) = \Psi(\{x_j\}) = -K_f \sum_{j=1}^{N} \|B_j(\{x_j(t)\})\|_2^2. \tag{18}$$

By LaSalle's principle [13], the agent configuration $\{x_j\}$ converges to the largest invariant set contained in the set $\Psi^{-1}(0)$. Note that the set $\Psi^{-1}(0)$ is the set of Static SMC configurations. Since all the elements of the set $\Psi^{-1}(0)$ are invariant to the dynamics in (15) subject to the control (17), the agent configuration asymptotically converges to the set of Static SMC configurations. If the set of Static SMC configurations is finite, the solution $\{x_j(t)\}$ converges asymptotically to one of the Static SMC configurations. But we are guaranteed convergence only to a local minimum of the coverage measure Φ.

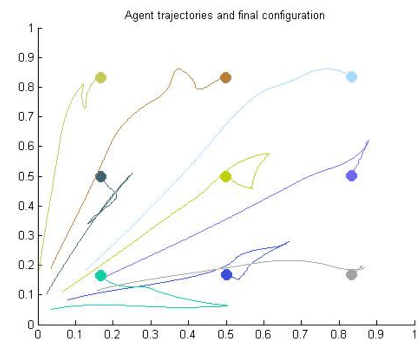
Different variations of the control law in (17) can also be used for coverage control. One such useful variation is the control law where

$$u_j(t) = \begin{cases} -u_{max} \frac{B_j(\{x_j(t)\})}{\|B_j(\{x_j(t)\})\|_2} \text{ if } \|B_j(\{x_j(t)\})\|_2 > \varepsilon \\ -K_f B_j(\{x_j(t)\}) \text{ if } \|B_j(\{x_j(t)\})\|_2 \leq \varepsilon, \end{cases} \tag{19}$$

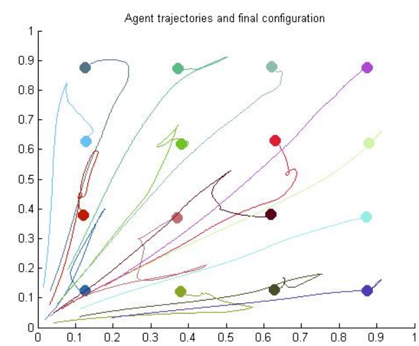
where u_{max} is the maximum speed of the agent, ε is a threshold that can be chosen and $K_f = u_{max}/\varepsilon$. By similar arguments as before, the control law in (19) would result in convergence to one of the Static SMC configurations.

A. Examples

In this section, we demonstrate the application of the Static SMC algorithm for various scenarios. First, we make an interesting observation that when the number of agents is $N = P^2$ where P is an integer and when the probability distribution μ is a uniform distribution on a rectangular domain U, the solution to the coverage problem is a configuration with the agents placed on a rectangular grid. This is shown by simulations shown in Figure 1 for $N = 3^2$ and $N = 4^2$. These simulations were for the domain $U = [0,1]^2$ and the control law in (19) was used with $u_{max} = 1.0$ and $\varepsilon = 10^{-3}$. Analytically proving that these rectangular grid configurations are indeed minimizers of the coverage metric remains an open problem and will be the subject of future work.



(a) $N = 3^2$



(b) $N = 4^2$

Fig. 1. Agent trajectories and Static SMC configurations obtained for a uniform prior on a rectangular domain $[0,1] \times [0,1]$ and with $N = P^2$. The initial agent locations were chosen randomly.

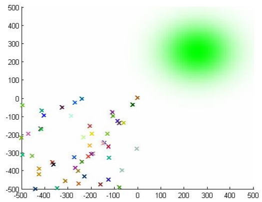
Figure 2 illustrates the performance of the Static SMC algorithm for a Gaussian distribution restricted to the domain $U = [-500, 500] \times [-500, 500]$, and is defined as

$$\mu(x) = \frac{0.5}{\pi\sigma^2} e^{-0.5\left(\frac{(x_1-\bar{x}_1)^2}{\sigma^2} + \frac{(x_2-\bar{x}_2)^2}{\sigma^2}\right)} \tag{20}$$

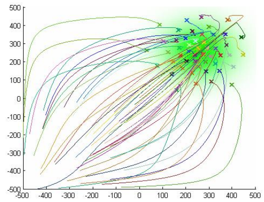
where $(\bar{x}_1, \bar{x}_2) = (250, 250)$ and $\sigma = 100$. In this simulation, the control law in (19) was used with $u_{max} = 15.0$ and $\varepsilon = 10^{-3}$.

Finally we illustrate an example where the objective is to uniformly cover a rectangular domain excluding regions covered by foliage which are represented as shaded regions in Figure 3. The shaded regions can be thought of as areas where there is no value in placing an agent or where there is zero probability of detecting an event (or target). The probability distribution μ is setup as follows. First, we define a Terrain function as:

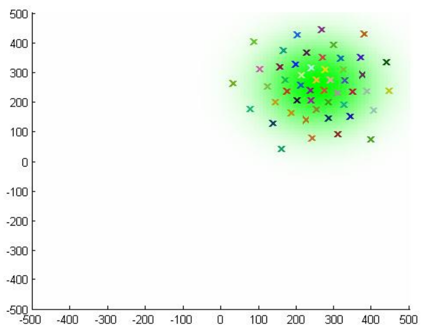
$$Ter(x) = \begin{cases} 1, & \text{if } x \text{ is outside foliage} \\ 0, & \text{if } x \text{ is inside foliage.} \end{cases} \tag{21}$$



(a) Initial agent locations



(b) Agent trajectories



(c) Final agent configuration

Fig. 2. Agent trajectories and Static SMC configuration obtained for a Gaussian probability distribution and with $N = 50$. The green coloring represents the Gaussian probability distribution.

Now, μ is defined as

$$\mu(x) = \frac{Ter(x)}{\int_U Ter(y)dy}. \tag{22}$$

Figure 3 shows the optimal agent configuration obtained for an irregular domain.

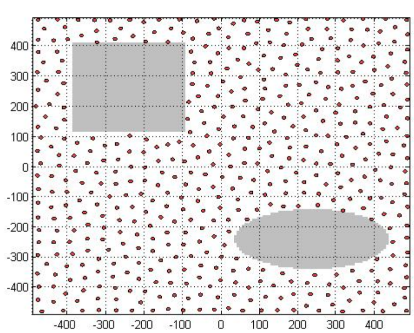


Fig. 3. Static SMC Configuration obtained for a uniform probability distribution on an irregular domain and with $N = 500$.

V. Coverage Control for Dubins vehicle dynamics

A Dubins vehicle is a vehicle moving with bounded curvature on a plane and is considered an adequate model from the perspective of path planning of unmanned vehicles (see [14] and [15]). The dynamics of an agent with Dubins vehicle dynamics is described by

$$\begin{aligned} \dot{x}_j &= v_j e^{i\theta_j} \\ \dot{\theta}_j &= \omega_j, \end{aligned} \tag{23}$$

where $e^{i\theta_j} = (cos(\theta_j), sin(\theta_j))'$ denotes the unit vector in the direction of the vehicle heading θ_j and v_j is the speed of the vehicle. Here, the controls to be chosen at each time instant are the speeds v_j and the turn-rates ω_j. We assume that the speeds and turn-rates are subject to the constraints:

$$\begin{aligned} 0 \le v_j \le v_{max} \\ \omega_{min} \le w_j \le w_{max}. \end{aligned} \tag{24}$$

The controls are obtained by using a receding horizon approach over a short time horizon Δt and deriving the feedback law in the limit as Δt goes to zero. For details see the Appendix (Section VII-A). The resulting control laws are given as

$$v_j = \begin{cases} 0 \text{ if } B_j(\{x_j(t)\}) \cdot e^{i\theta_j} >= 0 \\ v_{max} \text{ if } B_j(\{x_j(t)\}) \cdot e^{i\theta_j} < 0 \end{cases} \tag{25}$$

$$\omega_j = \begin{cases} \omega_{min}, \text{ if } B_j(\{x_j(t)\}) \cdot ie^{i\theta_j} >= 0 \\ \omega_{max} \text{ if } B_j(\{x_j(t)\}) \cdot ie^{i\theta_j} < 0. \end{cases} \tag{26}$$

The vector $ie^{i\theta_j}$ denotes the unit vector $(-sin(\theta_j), cos(\theta_j))'$. An alternate interpretation of the control in (25) and (26) is that the choice of the speed guarantees that the first time-derivative of the coverage metric is non-positive, while the choice of the turn-rate makes the second time-derivative of the coverage metric as negative as possible. This can be observed as follows. Computing the time-derivative of the coverage measure Φ, we get

$$\dot{\Phi}(\{x_j\}) = \sum_{j=1}^{N} B_j(\{x_j(t)\}) \cdot e^{i\theta_j} v_j \tag{27}$$

Therefore choosing v_j as in (25), it is guaranteed that the first time-derivative is always non-positive. Computing the second time-derivative of the coverage measure Φ, we get

$$\begin{aligned} \ddot{\Phi}(\{x_j\}) = &\sum_{j=1}^{N} B_j(\{x_j(t)\}) \cdot v_j ie^{i\theta_j} \omega_j + \sum_{j=1}^{N} B_j(\{x_j(t)\}) \cdot e^{i\theta_j} \dot{v}_j \\ &+ \sum_{j=1}^{N} \dot{B}_j(\{x_j(t)\}) \cdot e^{i\theta_j} v_j \end{aligned} \tag{28}$$

Therefore choosing ω_j as in (26) makes the second time-derivative as negative as possible. Figure 4 shows a sample simulation of the coverage control for the Dubins vehicle model with $v_{max} = 0.5$ and $-\omega_{min} = \omega_{max} = 10$. In this example, the probability distribution μ is uniform within the shaded region shown in Figure 4 and zero outside the shaded region.

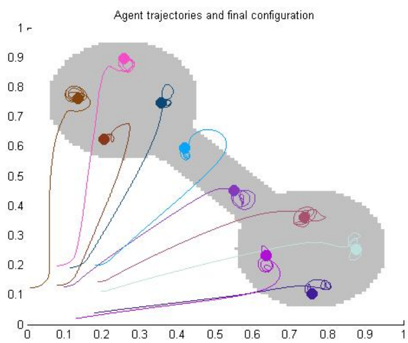


Fig. 4. Trajectories obtained with Static SMC control for Dubins vehicle model. The probability distribution μ is uniform within the shaded region and zero outside the shaded region.

VI. Coverage of Boundaries

The coverage algorithm described in this paper can also be effectively used for the uniform coverage of the boundary of a domain or more generally, for the coverage of geometric patterns. The problem of uniformly distributing agents on the boundary of an envirnoment has been studied using a different approach in [16]. The only element that needs to be changed in the Static SMC algorithm is in the definition of the probability distribution μ. For coverage of curves, the probability distribution μ will be a delta-like distribution whose support is on the curve. Let a curve be given as $\gamma : [a,b] \to \mathbb{R}^n$. Then the corresponding probability distribution is given as:

$$\mu = \frac{1}{b-a}\int_a^b \delta_{\gamma(s)} ds, \tag{29}$$

where $\delta_{\gamma(s)}$ is the delta distribution with support at the location $\gamma(s)$. Note that the Fourier coefficients of the probability distribution μ can be easily computed as:

$$\mu_k = \frac{1}{b-a}\int_a^b f_k(\gamma(s)) ds. \tag{30}$$

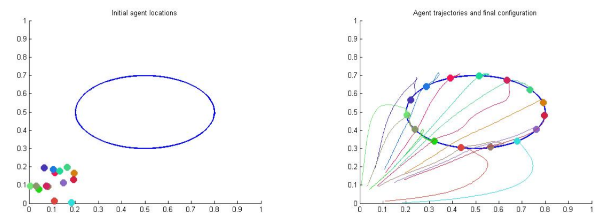
For an elliptical curve, we can use the parametrization $\gamma_{ell} : [0,2\pi] \to \mathbb{R}^2$ defined as

$$\gamma_{ell}(\theta) = \begin{bmatrix} \bar{x}_1 + acos(\theta) \\ \bar{x}_2 + bsin(\theta) \end{bmatrix}. \tag{31}$$

Figure 5 shows the uniform coverage of an elliptical curve with $(\bar{x}_1, \bar{x}_2) = (0.5, 0.5)$, $a = 0.3$ and $b = 0.2$. As a final example, we use a variation of a curve used in the paper [16]. Figure 6 shows the uniform coverage of a curve $\gamma_p : [0,1] \to \mathbb{R}^2$ defined as

$$\gamma_p(\theta) = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} + \frac{1}{8}(2 + cos(10\pi\theta) + 0.5sin(4\pi\theta)) \begin{bmatrix} cos(2\pi\theta) \\ sin(2\pi\theta) \end{bmatrix}. \tag{32}$$

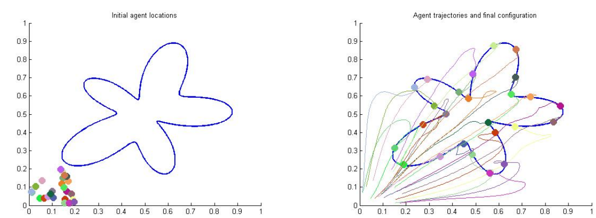
For both the simulations shown in Figures 5 and 6, the control law in (19) was used with $u_{max} = 1.0$ and $\varepsilon = 10^{-3}$. For better coverage of curves, it would be desirable for the probability distribution μ to be defined using unit speed curves so that μ is uniform along the curve.



(a) Initial agent locations and curve to be covered

(b) Agent trajectories and final configuration

Fig. 5. Agent trajectories and Static SMC Configurations obtained for coverage of an elliptical curve and with $N = 15$.



(a) Initial agent locations and curve to be covered

(b) Agent trajectories and final configuration

Fig. 6. Agent trajectories and Static SMC Configurations obtained for coverage of the curve (32) and with $N = 25$.

VII. Conclusions

In this paper, we have presented a centralized algorithm referred to as *Static SMC* for locational optimization of multi-agent systems. We make use of a multiscale coverage metric to quantify the quality of coverage by an agent network and use gradient descent algorithms for coverage control. In related work ([17]), we have extended the work in this paper to develop distributed and asynchronous implementations of the coverage algorithm.

Here it is worth mentioning and comparing Centroidal Voronoi Tessellations (CVT) and Static SMC configurations. Recall that for CVTs, the agent locations $\{x_j\}$ are obtained by minimizing the metric,

$$V(\{x_j\},\{W_j\}) = \sum_{j=1}^{N} \int_{W_i} f(||x_j - x||)\mu(x)dx, \tag{33}$$

which also depends on a chosen partition $\{W_i\}$ of U, and $||\cdot||$ is a Euclidean norm. Here, the non-decreasing function $f(||x_j - x||)$ provides a quantitative assessment of how poor the sensing performance of the sensor at x_j is at point x. It turns out that for any choice of the function f and for a fixed agent configuration, the Voronoi partition computed with respect to the Euclidean metric is the optimal partition [8]. For $f = ||x_i - x||^2$, which is a very common choice for f in locational optimization problems, CVTs are optimal solutions. A CVT is a configuration is one in which the centroids of the Voronoi cells (w.r.t μ) coincide with the generators of the Voronoi partition. A continuous time Lloyd descent algorithm to compute the CVTs has been proposed in [8], and is very similar to the form of the coverage control introduced in section IV for determining the Static SMC configurations. From numerical simulations, it was observed that CVTs and Static SMC configurations are qualitatively very similar. However, there is a very important difference. The Static SMC algorithm does not require the computation of any partitions at each time-step whereas the Lloyd algorithm and its variants require computing the Voronoi partition at every time-step. The computational complexity of the Static SMC algorithm is $O(N)$ as it requires only the computation of the sample averages of the Fourier basis functions over the agent locations. The computational complexity of the Lloyd algorithm is $O(NlogN)$ as the lower bound complexity for computing the Voronoi partition is $O(NlogN)$. Moreover, sophisticated algorithms and data structures have to be used for the efficient computation of the Voronoi partition. Further theoretical and numerical investigations are required to establish the connection between CVT and Static SMC configurations.

APPENDIX

A. Receding horizon control for Dubins vehicle

At a given time t, let us solve the optimal control problem over the time horizon $[t, t+\Delta t]$. The cost-function we are going to use is the coverage measure Φ at the end of the horizon. i.e., we aim to drive the agents to positions with the least value of the coverage measure subject to the dynamics of the agents and the constraints on the control. The cost-function that we want to minimize is given as

$$C(t,\Delta t) = \Phi(t+\Delta t) = \frac{1}{2}\sum_{K} \Lambda_k s_k^2(t+\Delta t). \tag{34}$$

It is convenient to write the optimal control solution in terms of the costate(Lagrange multipliers) and the Hamiltonian. The dynamics of the costates are given by the costate equations and the Hamiltonian is a function of the states, costates and the controls. The optimal control solutions v_j^* and ω_j^* are the values of the admissible controls v_j and ω_j that minimize the Hamiltonian. For more details on how to form the Hamiltonian and costate equations, see [18]. In our notation the costates are $\gamma_j(\tau) \in \mathbb{R}^n$ and $\sigma_j(\tau) \in \mathbb{R}$. The Hamiltonian can be written as

$$H = \sum_{j=1}^{N} \gamma_j \cdot v_j e^{i\theta_j} + \sum_{j=1}^{N} \sigma_j \omega_j \tag{35}$$

The costate equations are:

$$\begin{aligned} \dot{\gamma}_j &= -\frac{\partial H}{\partial x_j} = 0 \\ \dot{\sigma}_j &= -\frac{\partial H}{\partial \theta_j} = -\gamma_j \cdot v_j i e^{i\theta_j}. \end{aligned} \tag{36}$$

The terminal constraints are

$$\begin{aligned} &\gamma_j(t+\Delta t) = B_j(\{x_j(t+\Delta t\}); \\ &\sigma_j(t+\Delta t) = 0. \end{aligned} \tag{37}$$

The co-state values at time t upto first-order accuracy in Δt are given by using the terminal conditions and integrating the costate equations for a short horizon Δt and are given as

$$\begin{aligned} \gamma_j(t) &= \gamma_j(t+\Delta t) - \dot{\gamma}_j(t+\Delta t)\Delta t = B_j(\{x_j(t+\Delta t\}) \\ \sigma_j(t) &= \sigma_j(t+\Delta t) - \dot{\sigma}_j(t+\Delta t)\Delta t = B_j(\{x_j(t+\Delta t\}) \cdot v_j i e^{i\theta_j}\Delta t \end{aligned}$$

Therefore, in the limit as $\Delta t \to 0$, the optimal controls v_j and ω_j that minimize the Hamiltonian are given by equations (25) and (26).

ACKNOWLEDGMENT

The authors would like to thank Tuhin Sahai from UTRC for suggesting the problem. The authors would also like to thank Kunal Srivastava and Shaunak Bopardikar from UTRC for useful discussions on this work.

REFERENCES

[1] Q. Du, V. Faber, and M. Gunzburger, "Centroidal voronoi tessellations: Applications and algorithms," *SIAM Review*, vol. 41, no. 4, pp. 637–676, 1999.

[2] Q. Du, M. Gunzburger, and L. Ju, "Advances in studies and applications of centroidal voronoi tessellations," *Numerical mathematics: Theory, Methods and Applications*, 2010.

[3] G. Mathew and I. Mezić, "Spectral multiscale coverage: A uniform coverage algorithm for mobile sensor networks," in *IEEE Conf. on Decision and Control*, Shanghai, China, Dec. 2009.

[4] ——, "Metrics for ergodicity and design of ergodic dynamics for multi-agent systems," *Physica D: Nonlinear Phenomena*, vol. 240, no. 4-5, pp. 432–442, 2010.

[5] S. P. Lloyd, "Least squares quantization in pcm," *IEEE Tranactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[6] A. Okabe and A. Suzuki, "Locational optimization problems solved through voronoi diagrams," *European Journal of Operational Research*, vol. 98, no. 3, pp. 445–456, May 1997.

[7] C. Cassandras and W. Li, "Sensor networks and cooperative control," *European Journal of Control*, vol. 11, no. 4-5, pp. 436–463, 2005.

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

[9] J. W. Durham, R. Carli, P. Frasca, and F. Bullo, "Discrete partitioning and coverage control for gossiping robots," *CoRR*, vol. abs/1011.1939, 2010.

[10] G. Mathew, I. Mezić, and L. Petzold, "A multiscale measure for mixing," *Physica D: Nonlinear Phenomena*, vol. 211, pp. 23–46, November 2005.

[11] G. Mathew, I. Mezić, S. Grivopoulos, U. Vaidya, and L. Petzold, "Optimal control of mixing in stokes fluid flows," *Journal of Fluid Mechanics*, vol. 580, pp. 261–281, 2007.

[12] H. Niederreiter, *Random number generation and quasi-Monte Carlo methods*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[13] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.

[14] L. E. Dubins, "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, pp. 497–516, 1957.

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

[16] S. Susca, S. Martinez, and F. Bullo, "Monitoring environmental boundaries with a robotic sensor network," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 288–296, 2008.

[17] K. Srivastava, G. Mathew, A. Surana, and S. Bopardikar, "Distributed algorithms for static multiscale coverage," *to be submitted*, 2012.

[18] D. E. Kirk, *Optimal Control Theory: An Introduction*. Dover Publications, April 2004.