

A Variational Approach to Trajectory Planning for Persistent Monitoring of Spatiotemporal Fields

Xiaodong Lan and Mac Schwager

Abstract—This paper considers the problem of planning a trajectory for a sensing robot to best estimate a time-changing scalar field in its environment. We model the field as a linear combination of basis functions with time-varying weights. The robot uses a Kalman-like filter to maintain an estimate of the field, and to compute the error covariance of the estimate. The goal is to find a trajectory for the sensing robot that minimizes a cost metric on the error covariance and the control effort expended by the robot. Pontryagin’s Minimum Principle is used to find a set of differential equations that must be satisfied by the optimal trajectory. A numerical solver is used to find a trajectory satisfying these equations to give persistent monitoring trajectories.

I. INTRODUCTION

In this paper we adapt tools from optimal control to plan a trajectory for a sensing robot to monitor a continually changing field over its environment. Consider, for example, an aerial vehicle with a radiation sensor that must continually fly over the site of a nuclear accident to maintain an estimate of the time changing levels of radiation in the region. The objective of the vehicle is to execute a trajectory that optimizes the estimation accuracy over an arbitrarily long time horizon.

We model the environmental field using a set of spatial basis functions, where the field value at every point in the environment is given by a linear combination of these basis functions. The weights on the basis functions change according to a linear stochastic model with Gaussian white noise. This is a suitable model for a broad range of environmental scalar fields [1], [2], [3], including temperature fields, radio signal strength or electric field strength, radioactivity in a region around a nuclear accident, or fields of the concentration of some chemical contaminant such as oil around a leaking well in the ocean, or carbon monoxide in some region of the atmosphere, just to name a few. The model can also be applied to estimate the density of a population of interest, such as the concentration of people in different areas of a city, the density of vegetation over a forest, or the density of some species of animal over a region.

We assume that our robot has a sensor to measure the value of the field at its own location, with some sensor noise. This would be the case, for example, for a temperature sensor, a chemical concentration sensor, a radiation sensor,

and many other point-based sensors commonly used to sense scalar fields. Because of spatial correlation in the scalar field, these point measurements can be used to estimate the value at every other point in the field. We show that the optimal estimator has a Kalman-like form, with the critical distinction that the error covariance of the filter is a function of the trajectory of the robot. Intuitively, this is because the robot only obtains information about the part of the environment that is closest to it. Hence, we seek to plan a trajectory for our sensing robot to minimize a metric representing estimation quality, while also not expending excessive control effort. Since the environment is dynamic the agent must perpetually move to maintain an estimate of the field value at different places.

This is an example of a *persistent monitoring* task, as described in the literature [4], [5], [6]. Persistent monitoring requires the agents to monitor a dynamically changing environment that cannot be fully covered by a stationary team of agents. This differs from traditional coverage tasks because it requires all areas in the environment to be visited repeatedly since the environment is changing. Persistent monitoring has seen much interest in the recent literature. In [6] the authors approach this problem by controlling the speed of the agent to move along a given closed path. This problem is adapted to persistent ocean monitoring in [7]. A similar problem is solved using optimal control techniques in [4], where the authors consider multiple agents monitoring a 1-D dynamic environment. Our recent work [5] uses a randomized trajectory planner to get a periodic trajectory for the sensing robot in a Gaussian random field.

Our key contribution is a method for planning an optimal trajectory for monitoring the scalar field using Pontryagin’s Minimum Principle (PMP). PMP is typically used to find optimal *control* trajectories. Our application to find optimal *sensing* trajectories is a novel usage of this classical tool. In fact, one can consider our approach as optimally controlling the *uncertainty* of the estimation. Using PMP, we obtain a set of nonlinear differential equations that the optimal trajectory must satisfy. We also obtain initial conditions for some of the differential equations and final conditions for others, hence to find trajectories that satisfy these equations we must solve a two point boundary value problem. We do this by employing a numerical boundary value problem solver. Trajectories found in this way are known to be locally optimal in the sense that the cost of the trajectory cannot be reduced by infinitesimally perturbing the trajectory at any point in any direction. In future research we will investigate globally optimal trajectories using further arguments.

This work was supported in part by ONR grant N00014-12-1-1000. We are grateful for this support.

X. Lan is with the Department of Mechanical Engineering, Boston University, Boston, MA 02215, USA, xlan@bu.edu.

M. Schwager is with the Department and Mechanical Engineering and the Division of Systems Engineering, Boston University, Boston, MA, 02215, USA, schwager@bu.edu.

The rest of this paper is organized as follows. The problem is formulated in Section II. Pontryagin's Minimum Principle is applied to obtain necessary conditions for optimality for persistent monitoring in Section III. Section IV presents the results of numerical simulations, and we discuss conclusions in Section V.

II. PROBLEM FORMULATION

Here we introduce the mathematical model of the spatiotemporal field and the sensing robot, formulate two cost functions for evaluating persistent monitoring performance of the robot, and we formally state the persistent monitoring problem given this model and cost.

A. Spatiotemporal Field and Robot Model

Consider a dynamic scalar field $\phi(q, t) : \mathcal{Q} \times \mathbb{R}_{\geq 0} \mapsto \mathbb{R}$, where $\mathcal{Q} \subset \mathbb{R}^2$ is the domain of interest in the environment, and $q \in \mathcal{Q}$ is an arbitrary point in the domain. Suppose that the field can be decomposed as the dot product of a row vector of *static* spatial basis functions $C(q) = [c_1(q) \cdots c_n(q)]$ and a column vector of *time-changing* weights $x(t) = [x_1(t) \cdots x_n(t)]^T$, so that

$$\phi(q, t) = C(q)x(t). \quad (1)$$

Although it may appear to be limiting, this decomposition is quite general. For example, given any analytic spatiotemporal field, $\phi(q, t)$, we can take a Taylor series expansion, or a Fourier series expansion, (or many other kinds of expansions) to yield $C(q)x(t)$. In this case $C(q)$ will be the vector of 2D polynomial bases for the Taylor expansion, or a vector of 2D Fourier bases for the Fourier expansion, and $x(t)$ is the vector of Taylor or Fourier coefficients at each time. This is a common way to represent environmental fields [1], [2], [3]. In this paper we choose Gaussian radial basis functions as the bases, so that the i th element of $C(q)$ is given by $c_i(q) = K e^{-\|q - q_i\|^2 / 2\sigma_i^2}$, where q_i is the center position of the basis function. An example of a spatiotemporal scalar field modeled in this way is shown in Figure 1.

To model the time changing weights $x(t)$, we let the weights be driven by a stochastic linear system of the form

$$\dot{x}(t) = Ax(t) + w(t)$$

where A is a known $n \times n$ matrix and $w(t) \in \mathbb{R}^n$ is a Gaussian white noise process with known distribution $w(t) \sim N(0, Q)$, and Q is the covariance matrix which is positive definite and symmetric. Let t_0 denote the initial time and $x(t_0)$ the initial state vector. It is assumed that $x(t_0)$ is Gaussian distributed, independent of w , with known mean and known covariance matrix Σ_0 .

The sensing robot moves along a path in the environment and takes measurements. The dynamics of the robot is given by

$$\dot{p}(t) = u, \quad p(t_0) = p_0$$

where p is the position of the robot and u is the control input, which is unconstrained, and p_0 is the initial position of the robot and t_0 the initial time. We consider these simple

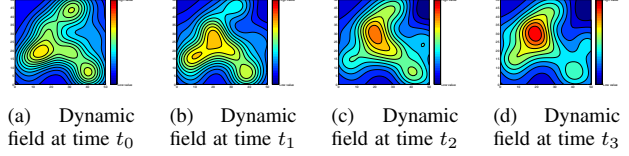


Fig. 1. Dynamic environment is changing with time $t_0 < t_1 < t_2 < t_3$. The color map indicates the field values and points on the same contour line have the same field value.

dynamics so as not to complicate the scenario, although more complex dynamics $\dot{p} = f(p, u)$ can be accommodated with some modifications. The robot has a sensor with which it can measure the field value at the robot's location, with some additive sensor noise

$$y(t) = \phi(p(t), t) + v(t) = C(p(t))x(t) + v(t)$$

where $v(t)$ a Gaussian white noise process uncorrelated with $w(t)$, and with known Gaussian distribution $v(t) \sim N(0, R)$. The sensor noise covariance matrix R is known. Our goal is to drive the sensing robot to estimate $\phi(q, t)$ while minimizing some measure of the estimation uncertainty and control effort.

B. Estimation Performance Metric

In order to measure the estimation performance from moving along a trajectory, an estimation performance metric must be defined. For a scalar field, we choose the variance of the estimation over the whole domain of interest as the metric. Denote the estimate of the field by $\hat{\phi}(q, t)$ and the estimation error by $e_\phi(q, t) = \hat{\phi}(q, t) - \phi(q, t)$. So our goal is to find a trajectory to minimize $E[\int_{\mathcal{Q}} e_\phi(q, t)^2 dq]$, either at some final time $t = t_f$, or integrated over an interval $t \in [t_0, t_f]$. Next we give a theorem to express $E[\int_{\mathcal{Q}} e_\phi(q, t)^2 dq]$ in a more convenient form. Denote the estimate of $x(t)$ by $\hat{x}(t)$ and the estimation error of $x(t)$ by $e_x(t) = \hat{x}(t) - x(t)$.

Theorem 1: Given an environmental region \mathcal{Q} , $E[\int_{\mathcal{Q}} e_\phi(q, t)^2 dq] = \text{tr}(M\Sigma(t))$, where $M = \int_{\mathcal{Q}} C(q)^T C(q) dq$ is a symmetric $n \times n$ constant matrix, and $\Sigma(t) = E[e_x(t)e_x(t)^T]$ is the $n \times n$ error covariance matrix of the weights.

Proof: By (1), we have $e_\phi(q, t) = C(q)e_x(t)$. By this fact and because $e_\phi(q, t)$ is scalar, we have the following relation.

$$\begin{aligned} E\left[\int_{\mathcal{Q}} e_\phi(q, t)^2 dq\right] &= E\left[\int_{\mathcal{Q}} (C(q)e_x(t))^T (C(q)e_x(t)) dq\right] \\ &= E\left[\int_{\mathcal{Q}} e_x(t)^T C(q)^T C(q) e_x(t) dq\right] \\ &= E\left[e_x(t)^T \left(\int_{\mathcal{Q}} C(q)^T C(q) dq\right) e_x(t)\right] \\ &= E[e_x(t)^T M e_x(t)] \end{aligned}$$

where $M = \int_{\mathcal{Q}} C(q)^T C(q) dq$, which is symmetric. Since $e_x(t)$ is a column vector and $M e_x(t)$ is also a column vector, we can write $e_x(t)^T M e_x(t) = \text{tr}(M e_x(t) e_x(t)^T)$. Hence, we

have

$$\begin{aligned} E\left[\int_{\mathcal{Q}} e_{\phi}(q, t)^2 dq\right] &= E[e_x(t)^T M e_x(t)] \\ &= E[\text{tr}(M e_x(t) e_x(t)^T)] = \text{tr}(M \Sigma(t)) \end{aligned}$$

because the trace and expectation operators are linear and they commute. ■

By this theorem, if we know $\Sigma(t)$, we know the estimation error of the field over the whole domain of interest. Since $\Sigma(t)$ denotes the estimation error of $x(t)$ and we know the linear dynamics of $x(t)$, we construct a Kalman-like filter to estimate $x(t)$. In this paper we will derive the filter from first principles (together with the optimal robot trajectory) using an optimal control approach. The traditional Kalman filter (outside of the context of persistent monitoring) was derived with a similar technique by Athans and Tse in [8]. Following their derivation, we know that the optimal filter for $\hat{x}(t)$ has a linear form,

$$\begin{aligned} \dot{\hat{x}}(t) &= F \hat{x}(t) + G y(t) \\ y(t) &= C(p(t)) x(t) + v(t) \end{aligned} \quad (2)$$

where F is a $n \times n$ matrix and G is a $n \times 1$ vector since the measurement $y(t)$ is a scalar. Athans and Tse [8] prove that if $\hat{x}(t)$ is an unbiased estimate of $x(t)$, then the initial state of $\hat{x}(t)$ must be equal to the mean of the initial state of $x(t)$ and the filter matrices F and G must be related according to

$$F = A - GC(p(t)) \quad (3)$$

And the error covariance matrix of $x(t)$ satisfies a linear matrix differential equation

$$\dot{\Sigma} = (A - GC(p(t)))\Sigma + \Sigma(A - GC(p(t)))^T + GRG^T + Q$$

where the initial covariance matrix is equal to Σ_0 , which is known. We can view the elements of the error covariance matrix as the state variables of a dynamical system and the filter gain G and robot trajectory $p(t)$ as the control variables of the dynamical system.

C. Trajectory Quality

We choose to minimize a function of $\text{tr}(M\Sigma(t))$ when a sensing robot moves along a trajectory. However, control energy cost is another factor considered in many problems since a robot often carries its batteries, so minimizing the control effort is necessary and useful to maximize the robot lifetime. Hence, the goal is to design a trajectory for the robot which minimizes the estimation error of the field as well as the control energy cost. Define the cost of a trajectory starting at t_0 and ending at t_f as follows:

$$J(u, p_0, \Sigma_0) = \int_{t_0}^{t_f} \left(\frac{1}{2} \|u(t)\|^2 + \text{tr}(M\Sigma(t)) \right) dt \quad (4)$$

Another choice for the cost could be considering the final trace and the energy cost:

$$J(u, p_0, \Sigma_0) = \text{tr}(M\Sigma(t_f)) + \int_{t_0}^{t_f} \frac{1}{2} \|u(t)\|^2 dt \quad (5)$$

It is possible to put some weights on the process trace, final trace and the energy cost. For example, if we care more about the estimation performance, we can put heavy weights on the trace instead of the energy cost.

We now give a formal statement of our persistent environmental monitoring problem.

Problem 1 (persistent monitoring problem): Given an initial time t_0 and a final time t_f , find an optimal control input (u^*, G^*) such that

$$(u^*, G^*) \in \underset{u \in U, G \in \mathbb{R}^{n \times 1}}{\text{argmin}} J(u, p_0, \Sigma_0)$$

subject to

$$\begin{aligned} \dot{\Sigma} &= (A - GC(p(t)))\Sigma + \Sigma(A - GC(p(t)))^T \\ &\quad + GRG^T + Q \end{aligned} \quad (6)$$

$$\dot{p}(t) = u, \quad p(t_0) = p_0, \quad \Sigma(t_0) = \Sigma_0 \quad (7)$$

III. NECESSARY OPTIMALITY CONDITIONS

As in [8], [9], the necessary condition is given by the Pontryagin's minimum principle. Define Hamiltonian for the cost functional in (4)

$$H = \text{tr}(\dot{\Sigma}\Lambda^T) + \lambda^T \dot{p} + \frac{1}{2} \|u\|^2 + \text{tr}(M\Sigma) \quad (8)$$

where $\Lambda \in \mathbb{R}^{n \times n}$ and $\lambda \in \mathbb{R}^{n \times 1}$ are the costate for Σ and p , respectively. By Pontryagin's minimum principle [10], we have

$$\dot{\Lambda} = -\frac{\partial H}{\partial \Sigma}, \quad \Lambda(t_f) = 0 \quad (9)$$

$$\dot{\lambda} = -\frac{\partial H}{\partial p}, \quad \lambda(t_f) = 0 \quad (10)$$

$$\frac{\partial H}{\partial u} = 0 \quad (11)$$

$$\frac{\partial H}{\partial G} = 0 \quad (12)$$

Theorem 2: $G^* = \Sigma C(p(t))^T R^{-1}$ is the optimal filter gain for the linear filter (2) used to do persistent monitoring for both cost functionals (4) and (5).

Proof: Let's consider the Hamiltonian corresponding to the cost functional (4). The proof is the same if we consider the Hamiltonian corresponding to cost functional (5).

Necessity: Let's calculate the partial derivative of the Hamiltonian with respect to G . By (6), (7) and (8),

$$\begin{aligned} H &= \text{tr}(\dot{\Sigma}\Lambda^T) + \lambda^T \dot{p} + \frac{1}{2} \|u\|^2 + \text{tr}(M\Sigma) \\ &= \text{tr}(A\Sigma\Lambda^T) - \text{tr}(GC(p(t))\Sigma\Lambda^T) + \text{tr}(\Sigma A^T \Lambda^T) \\ &\quad - \text{tr}(\Sigma C(p(t))^T G^T \Lambda^T) + \text{tr}(Q\Lambda^T) \\ &\quad + \text{tr}(GRG^T \Lambda^T) + \lambda^T u + \frac{1}{2} \|u\|^2 + \text{tr}(M\Sigma) \end{aligned} \quad (13)$$

Using the properties that $\partial \text{tr}(AX')/\partial X = A$, $\partial \text{tr}(AX'B)/\partial X = BA$, $\partial \text{tr}(AXBX')/\partial X = A'XB' + AXB$, $\text{tr}(A^T B) = \text{tr}(AB^T)$, $\text{tr}(AB) = \text{tr}(BA)$, and $\Sigma^T = \Sigma$, $R^T = R$, by (12), we have

$$-\Lambda \Sigma C(p(t))^T - \Lambda^T \Sigma C(p(t))^T + \Lambda^T GR + \Lambda GR = 0 \quad (14)$$

By (9), we have a differential equation for Λ .

$$\dot{\Lambda} = -(A - GC(p(t)))^T \Lambda - \Lambda(A - GC(p(t))) - M \quad (15)$$

By $\Lambda(t_f) = 0$ and (15), these two facts imply that Λ is symmetric positive definite. So Λ^{-1} exists and $\Lambda^T = \Lambda$. We can eliminate Λ in (14).

$$GR = \Sigma C(p(t))^T \Rightarrow G^* = \Sigma C(p(t))^T R^{-1}$$

Sufficiency: $G^* = \Sigma C(p(t))^T R^{-1}$ is also the sufficient condition that the linear filter is optimal to the persistent monitoring problem. The proof is straightforward but lengthy by considering the Hamilton-Jacobi-Bellman (HJB) equation and verifying $G^* = \Sigma C(p(t))^T R^{-1}$ satisfies the HJB equation. We refer the reader to [8], [11] for more details. ■

Remark 1: The above proof follows directly from the derivation of the continuous Kalman filter in [8] from the optimal control point of view. One can see that if we replace G by $\Sigma C(p(t))^T R^{-1}$ in (6), we get a continuous time Riccati equation for Kalman-Bucy filter: $\dot{\Sigma} = A\Sigma + \Sigma A^T - \Sigma C(p(t))^T R^{-1} C(p(t)) \Sigma + Q$. In the proof, we only consider the cost functional (4), but this theorem is still true for cost functional (5). This fact says that Kalman-Bucy filter is the optimal filter for the persistent monitoring problem in this paper, since the cost functional satisfies the condition given in [8]. That is, as long as the cost functional satisfies the condition: $\partial J / \partial \Sigma$ is symmetric and positive definite, Kalman-Bucy filter will be the optimal filter to minimize the cost functional.

Next we give the necessary conditions the optimal trajectory has to satisfy for the persistent monitoring problem.

Theorem 3: Necessary conditions for problem 1 with cost functional (4) are given by

$$\left. \begin{aligned} \dot{\Sigma} &= A\Sigma + \Sigma A^T - \Sigma C(p(t))^T R^{-1} C(p(t)) \Sigma + Q \\ \dot{p}(t) &= u, \quad p(t_0) = p_0, \quad \Sigma(t_0) = \Sigma_0 \\ \dot{\Lambda} &= -(A - \Sigma C(p(t))^T R^{-1} C(p(t)))^T \Lambda - \\ &\quad \Lambda(A - \Sigma C(p(t))^T R^{-1} C(p(t))) - M, \quad \Lambda(t_f) = 0 \\ \dot{\lambda} &= \frac{\partial C(p(t))}{\partial p} 2(\Sigma \Lambda \Sigma C(p(t))^T R^{-1}), \quad \lambda(t_f) = 0 \\ u^* &= -\lambda. \end{aligned} \right\} \quad (16)$$

Proof: The proof is straightforward by following Pontryagin's minimum principle. By Theorem 2 and (6) and (7), we have the differential equation for Σ and p , respectively. By Theorem 2 and (15), we have the differential equation for Λ .

By (10) and (13), using the property that $\partial \text{tr}(AXB) / \partial X' = BA$, $\partial \text{tr}(AXB') / \partial X = A'B$, we have a differential equation for λ .

$$\begin{aligned} \dot{\lambda} &= - \left(- \frac{\partial C(p(t))}{\partial p} \frac{\partial \text{tr}(GC(p(t)) \Sigma \Lambda^T)}{\partial C(p(t))^T} \right. \\ &\quad \left. - \frac{\partial C(p(t))}{\partial p} \frac{\partial \text{tr}(\Sigma C(p(t))^T G^T \Lambda^T)}{\partial C(p(t))^T} \right) \\ &= - \left(- \frac{\partial C(p(t))}{\partial p} \Sigma \Lambda^T G - \frac{\partial C(p(t))}{\partial p} \Sigma^T \Lambda G \right) \\ &= \frac{\partial C(p(t))}{\partial p} 2(\Sigma \Lambda \Sigma C(p(t))^T R^{-1}) \end{aligned}$$

because $\Sigma^T = \Sigma$, $\Lambda^T = \Lambda$ and $G = \Sigma C(p(t))^T R^{-1}$, $\frac{\partial C(p(t))}{\partial p}$ is $2 \times n$ matrix.

By (11) and (13), we can get the optimal control input:

$$u^* + \lambda = 0 \Rightarrow u^* = -\lambda.$$

This completes the proof. ■

If we consider cost functional (5), we will get similar necessary conditions except that $\dot{\Lambda} = -(A - \Sigma C(p(t))^T R^{-1} C(p(t)))^T \Lambda - \Lambda(A - \Sigma C(p(t))^T R^{-1} C(p(t)))$, $\Lambda(t_f) = M$.

IV. NUMERICAL SIMULATIONS

In this section, we give the results of numerical simulations finding persistent monitoring trajectories from the differential equations in Theorem 3. Solving the necessary conditions (16) analytically is difficult. There may be no closed form solutions for these conditions. It is common in optimal control problems to use numerical methods to solve such equations. These methods often transform the optimal control problem into a nonlinear programming problem. There are several methods used to do this transformation [12]: the shooting method [13], the collocation method [14], [15], [16], [17] and pseudospectral methods [18], [19]. In this paper, we will use the boundary value problem (BVP) solver *bvp4c* and *bvp5c* in Matlab to solve the equations numerically. Since BVPs may not have a solution, or may have finite number of solution, or may have infinitely many solutions, a guess for the solution is required when using this solver [20], [21]. This solver implements a collocation method for the solution of BVPs, which is called Simpson's method. It approximates the real solution by a cubic polynomial on each subinterval of a mesh in the time horizon. And the error is bounded by a fourth order of the maximum stepsize of the subintervals. We will consider three different cases in a 2D environment when the cost functional is given by (4).

A. One Basis Function

If there is only one basis function to represent the environment, then the system dimension $n = 1$ and the covariance matrix is a scalar. According to our model, the sensor will get more information when it stays close to the center position of the basis. Hence, we would expect the sensing robot will stay at some point which is very close to the basis function, even on top of the basis function exactly.

We use Matlab solver *bvp4c* to solve the necessary conditions (16). The simulation parameters are chosen as follows. The environmental region is a 35×35 square region with x coordinate ranging from 0 to 35 and y coordinate ranging from 0 to 35. The starting time is $t_0 = 0$, the final time is $t_f = 10s$, the system matrix is $A = -0.99$, the process covariance matrix is $Q = 5$ and the measurement covariance matrix is $R = 10$. The basis function is given by $C(p(t)) = K e^{-\|p - q_1\|^2 / 2\sigma_c^2}$, where $\sigma_c = 6$, the basis center position is chosen to be $q_1 = [20, 25]^T$ and the scaling factor $K = 1$. So $M = \int_{\mathcal{Q}} C(q)^T C(q) dq = 112.0327$. The boundary conditions are $p(t_0) = [0, 0]^T$, $\Sigma(t_0) = 3$, $\lambda(t_f) = [0, 0]^T$, $\Lambda(t_f) = 0$. The guess of the mesh points is 1000

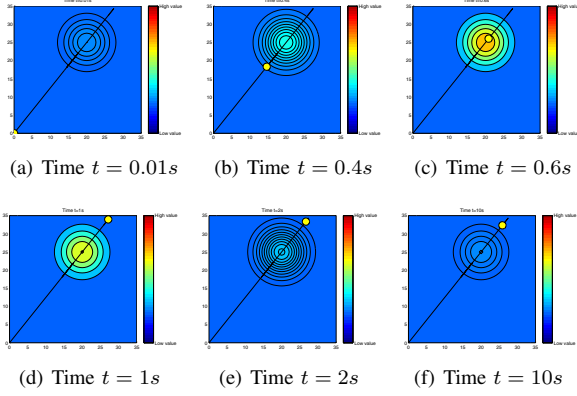


Fig. 2. Trajectory (black line) of a sensing robot (yellow circle) monitoring a dynamic environment. The colormap represents the estimation uncertainties of the field over the whole environment. The points in the same contour line have the same estimation uncertainties. The sensing robot will move towards the basis function quickly and stays at one point close to it.

linearly and equally spaced points between t_0 and t_f . The guess for the solution is a constant guess with $p(t) = [8, 10]^T$, $\lambda(t) = [0.1, 0.1]^T$, $\Sigma(t) = 200$, $\Lambda(t) = 0.05$. The trajectory for the sensing robot monitoring the environment is shown in Figure 2.

In Figure 2, the environment is discretized into a 50×50 grid. The estimation uncertainty of the field at each point on the grid is equal to $\text{tr}(m\Sigma)$, where $m = C(p(t))^T C(p(t))$ is a scalar, note that m is different from M since M is for the whole domain of interest while m is for one point in the domain. The first 5 pictures show that the sensing robot starts at the origin, moves towards the basis function quickly and stays at one point which is close to the basis function, because the uncertainties close to the basis function centers are higher.

B. Two Basis Functions

When there are two basis functions representing the environment, the system dimension is $n = 2$. The sensor is always trying to get more information about the basis so as to minimize the estimation uncertainty. Hence, we would expect the sensing robot will move back and forth between these two basis functions.

We again use Matlab solver *bvp4c* to solve the necessary conditions (16). The simulation parameters are chosen as follows. The environmental region is a 40×35 rectangular region with the x coordinate ranging from 10 to 50 and y coordinate ranging from 0 to 35. The starting time is $t_0 = 0$, the final time is $t_f = 10s$, the system matrix is $A = \text{diag}([-0.99, -0.99])$, where $\text{diag}([a_1, a_2, \dots, a_n])$ denotes a diagonal matrix whose diagonal entries are equal to $[a_1, a_2, \dots, a_n]$, the process covariance matrix is $Q = \text{diag}([5, 5])$ and the measurement covariance matrix is $R = 10$. The basis functions are given by $C(p(t)) = [Ke^{-\|p-q_1\|^2/2\sigma_c^2}, Ke^{-\|p-q_2\|^2/2\sigma_c^2}]$, where $\sigma_c = 6$, the basis positions are $q_1 = [20, 25]^T$, $q_2 = [40, 25]^T$ and the scaling factor is $K = 1$. So $M = \int_{\mathcal{Q}} C(q)^T C(q) dq$ is a symmetric 2×2 matrix. The boundary condition is $p(t_0) =$

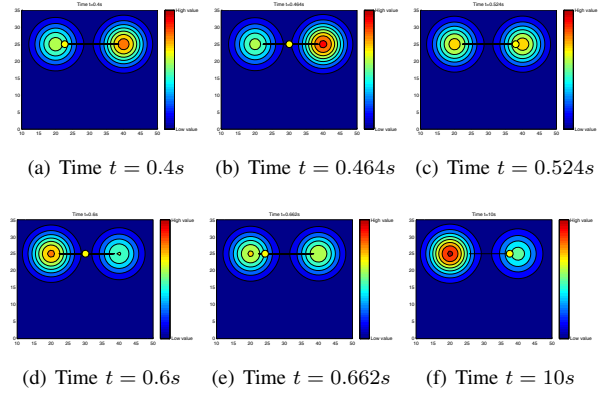


Fig. 3. Trajectory (black line) of a sensing robot (yellow circle) monitoring a dynamic environment. The sensing robot will move back and forth between the two basis functions.

$[25, 25]^T$, $\Sigma(t_0) = \text{diag}([3, 3])$, $\lambda(t_f) = [0, 0]^T$, $\Lambda(t_f) = \text{diag}([0, 0])$. The guess of the mesh points is 5000 linearly and equally spaced points between t_0 and t_f . The guess for the solution is a constant guess with $p(t) = [30, 25]^T$, $\lambda(t) = [0, 0]^T$, $\Sigma(t) = \text{diag}([200, 200])$, $\Lambda(t) = [0.1, 0.1; 0.1, 0.1]$.

The trajectory for the sensing robot monitoring the environment is shown in Figure 3. In this figure, the environment is also discretized into a 50×50 grid. The estimation uncertainty of the field at each point on the grid is equal to $\text{tr}(m\Sigma)$, where $m = C(p(t))^T C(p(t))$ is a 2×2 symmetric matrix. The robot will move back and forth repeatedly between the two basis functions since the field of the points close to these two basis functions changes quickly and the uncertainties there are higher. The first 5 pictures show how the field estimation uncertainty changes when the robot moves between the two basis functions in a short time. This oscillating behavior continues until the last picture in which the robot stops close to the center of one basis function.

C. Three Basis Functions

The system dimension is $n = 3$ when there are three basis functions in the environment. The sensor will try to move between these three basis functions to update its information about the field of the nearby points.

We use Matlab solver *bvp5c* to solve the necessary conditions (16). The simulation parameters are chosen as follows. The environmental region is a 40×35 rectangular region with x coordinate ranging from 10 to 50 and y coordinate ranging from 0 to 35. The starting time is $t_0 = 0$, the final time is $t_f = 200s$, the system matrix is $A = \text{diag}([-0.99, -0.99, -0.99])$, the process covariance matrix is $Q = \text{diag}([5, 5, 5])$ and the measurement covariance matrix is $R = 10$. The basis functions are given by $C(p(t)) = [Ke^{-\|p-q_1\|^2/2\sigma_c^2}, Ke^{-\|p-q_2\|^2/2\sigma_c^2}, Ke^{-\|p-q_3\|^2/2\sigma_c^2}]$, where $\sigma_c = 6$, the basis center positions are $q_1 = [20, 10]^T$, $q_2 = [40, 10]^T$, $q_3 = [30, 25]^T$ and the scaling factor is $K = 30$. We put some weights on $\text{tr}(M\Sigma)$ and let $M = \frac{1}{900} \int_{\mathcal{Q}} C(q)^T C(q) dq$, which is a symmetric 3×3 matrix. The boundary conditions are

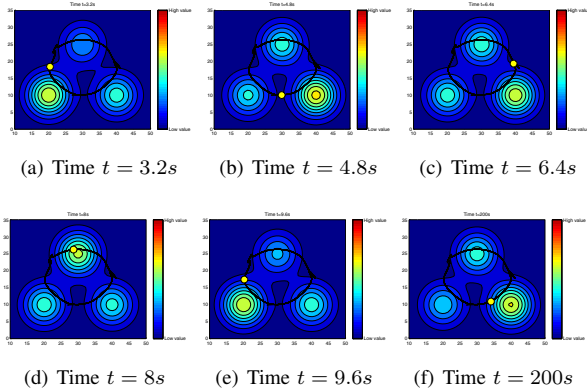


Fig. 4. Trajectory (black curve) of a sensing robot (yellow circle) monitoring a dynamic environment. The sensing robot will move around the three basis functions in a nearly periodic fashion.

$p(t_0) = [40, 18]^T$, $\Sigma(t_0) = \text{diag}([3, 3, 3])$, $\lambda(t_f) = [0, 0]^T$, $\Lambda(t_f) = \text{diag}([0, 0, 0])$. The guess of the mesh points is 5000 linearly and equally spaced points between t_0 and t_f . The guess for the solution is a function guess with $p(t) = [30 + 10\cos(t), 18 + 8\sin(t)]^T$, $\lambda(t) = [0.1, 0.1]^T$. The guess for Σ and Λ are chosen as follows:

$$\Sigma(t) = \begin{bmatrix} 20 & 0.5 & 0.5 \\ 0.5 & 20 & 0.5 \\ 0.5 & 0.5 & 20 \end{bmatrix} \quad \Lambda(t) = \begin{bmatrix} 0.5 & 0.1 & 0.1 \\ 0.1 & 0.5 & 0.1 \\ 0.1 & 0.1 & 0.5 \end{bmatrix}$$

The trajectory for the sensing robot monitoring the environment is shown in Figure 4. In this figure, the environment is also discretized into a 50×50 grid. The estimation uncertainty of the field at each point on the grid is equal to $\text{tr}(m\Sigma)$, where $m = \frac{1}{900}C(p(t))^TC(p(t))$ is a 3×3 symmetric matrix. The robot will move around the three basis functions. The first 5 pictures show how the field estimation uncertainty changes when the robot moves around the three basis functions in a short time. The last picture shows the stop position of the robot in the end.

V. CONCLUSIONS AND FUTURE WORK

In this paper we proposed an approach using the calculus of variations to plan trajectories for sensing robots to monitor a dynamic environment. We modeled the environment by basis functions with time-varying weights. Then we use a linear filter to estimate the field in the environment, which is proven to be the Kalman-Bucy filter. Then we use Pontryagin's minimum principle to get necessary conditions for the optimal trajectories of the mobile sensor. Numerical solvers are employed to solve these necessary conditions numerically and give locally optimal trajectories for the sensing robot. We demonstrate the effectiveness of this approach by considering three different cases for the environment. In the future we plan to investigate planning trajectories for persistent monitoring problems over an infinite horizon by using the same approach. Considering sufficient conditions for the optimal trajectories might be another interesting direction. Hamilton-Jacobi-Bellman equation can be employed to derive the sufficient conditions.

ACKNOWLEDGEMENT

We thank C. Cassandras, X. Lin, and S. L. Smith for many enlightening discussions on optimal control and persistent monitoring.

REFERENCES

- [1] J. Cortés, "Distributed Kriged Kalman filter for spatial estimation," *IEEE Transactions on Automatic Control*, vol. 54, no. 12, pp. 2816–2827, 2009.
- [2] N. Cressie, "The origins of kriging," *Mathematical Geology*, vol. 22, no. 3, pp. 239–252, 1990.
- [3] N. Cressie, *Statistics for Spatial Data*. New York: Wiley, 1993.
- [4] C. Cassandras, X. Lin, and X. Ding, "An optimal control approach to the multi-agent persistent monitoring problem," *Automatic Control, IEEE Transactions on*, vol. 58, no. 4, pp. 947–961, 2013.
- [5] X. Lan and M. Schwager, "Planning periodic persistent monitoring trajectories for sensing robots in gaussian random fields," in *2013 IEEE International Conference on Robotics and Automation*, 2013.
- [6] S. L. Smith, M. Schwager, and D. Rus, "Persistent robotic tasks: Monitoring and sweeping in changing environments," *IEEE Transactions on Robotics*, vol. 28, pp. 410–426, April 2012.
- [7] 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, pp. 714–741, September-October 2011.
- [8] M. Athans and E. Tse, "A direct derivation of the optimal linear filter using the maximum principle," *Automatic Control, IEEE Transactions on*, vol. 12, no. 6, pp. 690–698, 1967.
- [9] M. Athans, "The matrix minimum principle," *Information and control*, vol. 11, no. 5, pp. 592–606, 1967.
- [10] L. S. Pontryagin, V. G. Boltyanskii, and R. V. Gamkrelidze, *The Mathematical Theory of Optimal Processes*, vol. 4. Wiley and Sons, NY, 1962. Trans. L. Semenovich.
- [11] E. T.-S. Tse, "Application of pontryagin's minium principle to filtering problems," Master's thesis, Massachusetts Institute of Technology, 1967.
- [12] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of guidance, control, and dynamics*, vol. 21, no. 2, pp. 193–207, 1998.
- [13] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*. No. 13, SIAM, 1995.
- [14] U. Ascher, J. Christiansen, and R. D. Russell, "Collocation software for boundary-value odes," *ACM Transactions on Mathematical Software (TOMS)*, vol. 7, no. 2, pp. 209–222, 1981.
- [15] U. M. Ascher and R. J. Spiteri, "Collocation software for boundary value differential-algebraic equations," *SIAM Journal on Scientific Computing*, vol. 15, no. 4, pp. 938–952, 1994.
- [16] D. A. Benson, G. T. Huntington, T. P. Thorvaldsen, and A. V. Rao, "Direct trajectory optimization and costate estimation via an orthogonal collocation method," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1435–1440, 2006.
- [17] J. V. Villadsen and W. E. Stewart, "Solution of boundary-value problems by orthogonal collocation," *Chemical Engineering Science*, vol. 22, no. 11, pp. 1483–1501, 1967.
- [18] F. Fahroo and I. M. Ross, "Costate estimation by a legendre pseudospectral method," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, pp. 270–277, 2001.
- [19] F. Fahroo and I. M. Ross, "Direct trajectory optimization by a chebyshev pseudospectral method," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 1, pp. 160–166, 2002.
- [20] J. Kierzenka and L. F. Shampine, "A bvp solver based on residual control and the matlab pse," *ACM Transactions on Mathematical Software (TOMS)*, vol. 27, no. 3, pp. 299–316, 2001.
- [21] L. F. Shampine, J. Kierzenka, and M. W. Reichelt, "Solving boundary value problems for ordinary differential equations in matlab with bvp4c," *Tutorial notes*, 2000.
- [22] W. Zhang, M. P. Vitus, J. Hu, A. Abate, and C. J. Tomlin, "On the optimal solutions of the infinite-horizon linear sensor scheduling problem," in *2010 49th IEEE Conference on Decision and Control (CDC)*, (Atlanta, GA, USA), pp. 396 – 401, 15-17 Dec. 2010.