This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2405292, IEEE Transactions on Automatic Control


# A Geometric Transversals Approach to Sensor Motion Planning for Tracking Maneuvering Targets

Hongchuan Wei, *Student Member, IEEE*, and Silvia Ferrari, *Senior Member, IEEE*

***Abstract*—This paper presents a geometric transversals approach for representing the probability of track detection as an analytic function of time and target motion parameters. By this approach, the optimization of the detection probability subject to sensor kinodynamic constraints can be formulated as an optimal control problem. Using the proposed detection probability function, the necessary conditions for optimality can be derived using calculus of variations, and solved numerically using a variational iteration method (VIM). The simulation results show that sensor state and control trajectories obtained by this approach bring about a significant increase in detection probability compared to existing strategies, and require a computation that is significantly reduced compared to direct methods.**

***Index Terms*—Mobile sensor networks, geometric transversals, track coverage, optimal control, target tracking, detection theory.**

## I. Introduction

THE problem of tracking moving targets by means of a mobile sensor network is relevant to a wide range of applications, including environmental and atmospheric monitoring, security and surveillance, tracking of endangered species, and condition-based diagnostics [1]–[3]. It has been previously shown that the quality-of-service (QoS) of sensor networks performing cooperative target tracking can be quantified by track coverage functions derived using geometric transversals and probability theory, assuming targets move at constant speed and heading in the region-of-interest (RoI) [4], [5].

Recently, the geometric transversals approach in [4] was extended to maneuvering targets described by Markov motion models and used to optimize the detection probability of static sensor networks [6]. This paper extends the results in [6] to the problem of tracking a maneuvering target by a network of omnidirectional sensors mounted on mobile vehicles, and referred to simply as *mobile sensors*. The advantage of mobile sensors over static sensors is that, over time, they can cover larger portions of the RoI, and they can plan their paths based on where targets are expected to travel to at future times. Although optimal control has been previously applied to mobile sensor networks, its applicability is often limited by the lack of suitable objective functions. This paper shows that, using the proposed track coverage function, optimal control can be used to obtain optimality conditions and solutions for maximizing the detection probability over time, based on the probability distributions describing the target Markov motion model.

There is considerable precedence in the tracking and estimation literature for modeling target dynamics by Markov motion models [7], [8]. Using the approach presented in this paper, mobile sensors can be controlled based on the Markov transition probability density functions (PDFs) that are routinely outputted by tracking and estimation algorithms [7], [9]. Because the track coverage function is not quadratic, the optimal control problem may be solved using direct or indirect numerical methods [10], [11]. Direct methods determine near optimal solutions by discretizing the continuous-time problem and transcribing it into a finite-dimensional nonlinear program (NLP). Thus, they may become intractable for more than a few sensors. Using the proposed track coverage function, this paper derives necessary conditions for optimality, also known as Euler-Lagrange (EL) equations, and then determines a numerical solution using a variational iteration method (VIM) that exploits the integro-differential structure of the EL equations to reduce computational complexity. The numerical simulations show that, by this approach, the detection probability is significantly increased compared to existing potential field, greedy, grid, and random deployment algorithms.

The authors are with the Laboratory for Intelligent Systems and Controls (LISC), at Duke University, Durham, NC 27708-0005. E-mail: hongchuan.wei@duke.edu, sferrari@duke.edu

## II. Sensor Network Motion Planning (SNMP) Problem Formulation

This paper considers the problem of planning the state and control trajectories of a network of $n$ mobile sensors that seek to cooperatively detect a moving target in a two-dimensional RoI, $\mathcal{A} = [0,\ L_x] \times [0,\ L_y]$, during a fixed time interval $(T_0, T_f]$. Each sensor is mounted on a vehicle that is assumed to obey linear and time-invariant (LTI) equations of motion. Let $\mathbf{s}_i \in \mathcal{A}$ and $\mathbf{u}_i \in \mathbb{R}^2$ denote the state and control of the $i$th vehicle, respectively, such that $\mathbf{s} = [\mathbf{s}_1^T \dots \mathbf{s}_n^T]^T$ and $\mathbf{u} = [\mathbf{u}_1^T \dots \mathbf{u}_n^T]^T$ denote the state and control of the sensor network, respectively. Then, the network dynamics can be represented by the state-space equation,

$$\dot{\mathbf{s}}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{B}\mathbf{u}(t), \quad \mathbf{s}(T_0) = \mathbf{s}_0 \tag{1}$$

where $\mathbf{A}$ and $\mathbf{B}$ are known matrices of constant parameters [12]. From the actuator limits, the control vector is subject to the inequality constraint,

$$-\mathbf{1} \leq \mathbf{u}(t) \leq \mathbf{1}, \tag{2}$$

where $\mathbf{1}$ denotes a $2n \times 1$ vector of 1s, and the physical scaling parameters are absorbed into $\mathbf{B}$.

Assuming every sensor in the network is a passive, omnidirectional sensor, the field-of-view (FoV) can be represented by a disk $\mathcal{C}_i(t) = \mathcal{C}[\mathbf{s}_i(t), r_i]$, with constant radius or *effective range* $r_i \in \mathbb{R}$, and centered at $\mathbf{s}_i$. Then, the probability that the $i$th sensor detects a target at $\mathbf{x}(t) \in \mathcal{A}$, at time $t$, can be described by the Boolean detection model [13]–[16],

$$P_b[\mathbf{s}_i(t), \mathbf{x}(t)] = \begin{cases} 0: & \|\mathbf{s}_i(t) - \mathbf{x}(t)\| > r_i \\ 1: & \|\mathbf{s}_i(t) - \mathbf{x}(t)\| \leq r_i \end{cases}, \ 1 \leq i \leq n \tag{3}$$

where $\|\cdot\|$ denotes the $L_2$-norm.

This paper considers the problem of planning the sensor motion based on the Markov transition probability density functions (PDFs) that are routinely outputted by tracking and estimation routines for assimilating distributed sensor measurements [7]. Markov motion models assume that the target obeys the kinematic equations,

$$\dot{\mathbf{x}}(t) \triangleq \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \end{bmatrix} = \begin{bmatrix} v(t)\cos\theta(t) \\ v(t)\sin\theta(t) \end{bmatrix}, \ t \in (T_0,\ T_f] \tag{4}$$

where $v(t)$ is the target velocity, and $\theta(t)$ is the target heading. It is also assumed that the target heading and velocity remain constant during $m$ subintervals $(t_j,\ t_{j+1}]$, $j = 1, \dots, m$, that are an exact cover of $(T_0, T_f]$. At any time $t_j$, $j = 1, \dots, m$, the target may change its heading and velocity and, thus, $t_1, \dots, t_m$ are referred to as *maneuvering times*. Now, letting $\mathbf{x}_j \triangleq \mathbf{x}(t_j)$, $\theta_j \triangleq \theta(t_j)$,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2405292, IEEE Transactions on Automatic Control


$v_j \triangleq v(t_j)$, and integrating (4) over time yields the target motion model,

$$\mathbf{x}_{j+1} = \mathbf{x}_j + [v_j \cos\theta_j \quad v_j \sin\theta_j]^T \Delta t_j, \quad j = 1, \ldots, m \tag{5}$$

where $\Delta t_j \triangleq t_{j+1} - t_j$.

Because the target motion is unknown *a priori*, the target position, speed, and heading, are all viewed as independent, continuous random variables. Let $\mathbf{X}_j$ denote the random target position at $t_j$, $\Theta_j$ denote the random target heading in $(t_j,\ t_{j+1}]$, and $V_j$ denote the random target speed in $(t_j,\ t_{j+1}]$. Then, $\mathbf{X}_j$ can take any value $\mathbf{x}_j \in \mathcal{A}$ with a probability defined by the PDF $f_{\mathbf{X}_j}(\mathbf{x}_j)$, $\Theta_j$ can take any value $\theta_j \in [\theta_{\min},\ \theta_{\max}]$ with a probability defined by the PDF $f_{\Theta_j}(\theta_j)$, and $V_j$ can take any value $v_j \in [v_{\min},\ v_{\max}]$ with a probability defined by the PDF $f_{V_j}(v_j)$. From (5), the set of Markov parameters at the $j$th time interval, $\mathcal{M}_j \triangleq \{\mathbf{x}_j, \theta_j, v_j\}$, depends only on the motion parameters at the previous time interval, or $\mathcal{M}_{j-1}$. Thus, it can be easily shown that the sequence $\{\mathcal{M}_1, \ldots, \mathcal{M}_m\}$ is a Markov chain [17], and $\mathcal{M}_j$ is a set of Markov motion parameters that can be described by the PDFs $f_{\mathbf{X}_j}(\mathbf{x}_j)$, $f_{\Theta_j}(\theta_j)$, and $f_{V_j}(v_j)$, $j = 1, \ldots, m$. For simplicity, in this paper, the maneuvering time(s), $t_j$, are assumed known *a priori* for all $j$.

An example of Markov motion realization (target track) obtained from the PDFs in Table I is shown in Fig. 1, and an example of sensor trajectory and FoV are plotted in Fig. 2. Since both the target and the sensor move over time, a detection can only occur when the target track intersects the region spanned by the sensor FoV in $\Omega \triangleq \mathcal{A} \times (T_0, T_f] \subset \mathbb{R}^3$. We are now ready to state the problem addressed in this paper:

**Problem II.1 (Sensor Network Motion Planning (SNMP)).** Given the PDFs of the Markov parameters $\mathcal{M}_j$, $j = 1, \ldots, m$, for a target traversing the RoI $\mathcal{A} \subset \mathbb{R}^2$, find the network state and control trajectories, $\mathbf{s}^*(t)$ and $\mathbf{u}^*(t)$, such that the probability of detection is maximized over $(T_0,\ T_f]$.

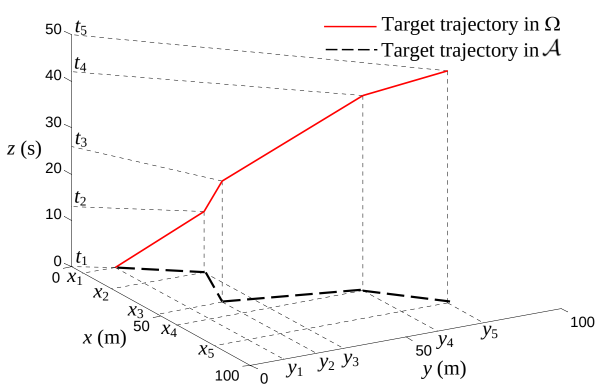


Fig. 1. Example of target trajectory realization sampled from Markov motion model in Table I.

## III. Probability of Track Detection

This section extends the results in [6] to mobile sensor networks and derives an objective function representing the probability of track detection as a function of the time-varying network state, $\mathbf{s}(t)$. Then, the detection probability function can be optimized subject to the network dynamic equation (1) using optimal control theory. From the detection model (3), the $i$th sensor has a nonzero probability to detect a target if and only if $\|\mathbf{x}(t)-\mathbf{s}_i(t)\| \le r_i$. It can be shown that, as the $i$th sensor moves along a trajectory $\mathbf{s}_i(t)$, the set of all tracks detected is contained by a time-varying three-dimensional coverage cone in $\Omega$ defined according to the following lemma:

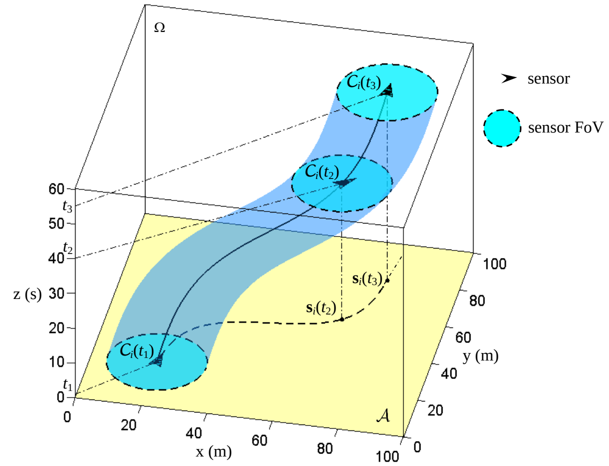


Fig. 2. Example of sensor trajectory and sensor FoV plotted at three moments in time $t_1 = 1(s)$, $t_2 = 40(s)$, and $t_3 = 55(s)$.

**Lemma III.1.** The $i$th coverage cone defined as,

$$K_i(t) = \Big\{ [x\ y\ z]^T \in \mathbb{R}^3 \ \Big|\ t_j < z \le t_{j+1}, \tag{6}$$
$$\Big\| [x\ y]^T - \frac{(z-t_j)}{(t-t_j)}[\mathbf{s}_i(t) - \mathbf{x}_j] - \mathbf{x}_j \Big\| \le \frac{(z-t_j)}{(t-t_j)} r_i \Big\}$$

contains the set of all target tracks that intersect the $i$th sensor FoV, $\mathcal{C}_i(t)$, at any time $t \in (t_j,\ t_{j+1}]$.

*Proof:* Let the directed line segment $\mathbf{m}_j(t) \subset \Omega$ represent a target track as it evolves from time $t_j$ to $t$, such that,

$$\mathbf{m}_j(t) = \left\{ \mathbf{y} \in \mathbb{R}^3 \ \middle|\ \mathbf{y} = \mathbf{z}_j + \alpha\left(\begin{bmatrix} \mathbf{x}^T(t) \\ t \end{bmatrix} - \mathbf{z}_j\right),\ \alpha \in (0, 1] \right\} \tag{7}$$

where $\mathbf{z}_j = [\mathbf{x}_j^T \quad t_j]^T$ is the segment origin in an inertial frame $\mathcal{F}_\Omega$ embedded in $\Omega$. Then, any point $\mathbf{a} \in \mathbf{m}_j(t)$, represented as a constant three-dimensional vector $\mathbf{a} = [a_x\ a_y\ a_z]^T$, obeys the equality, $[a_x\ a_y]^T = (a_z - t_j)[\mathbf{x}(t) - \mathbf{x}_j]/(t - t_j) + \mathbf{x}_j$. From (3), a target at $\mathbf{a}$ is detected if and only if

$$\Big\| [a_x\ a_y]^T - \frac{(a_z - t_j)}{(t-t_j)}[\mathbf{s}_i(t) - \mathbf{x}_j] - \mathbf{x}_j \Big\| \le \frac{(a_z - t_j)}{(t-t_j)} r_i, \tag{8}$$

Thus, from (6), any point on $\mathbf{m}_j(t)$ contained by $\mathcal{C}_i(t)$ must be contained by the coverage cone, and thus $\mathbf{m}_j(t) \in K_i(t)$. ■

As illustrated in Fig. 3, the above lemma extends the definition of the *fixed* spatio-temporal coverage cone presented in [6] to a *time-varying* coverage cone $K_i(t)$ that is a function of the sensor trajectory $\mathbf{s}_i(t)$. Because $K_i(t)$ is a circular cone that is possibly oblique, a Lebesgue measure of the tracks contained by $K_i(t)$ can be obtained by considering the pair of two-dimensional (2D) cones, referred to as *heading cone* and *velocity cone* [6], and reviewed in the next section.

### A. Heading and Velocity Cones Representation

The heading cone, denoted by $K_\theta(t)$, contains all target headings that lead to a detection by the $i$th sensor at any time $t \in (t_j,\ t_{j+1}]$ and, thus, it is obtained from the projection of $K_i(t)$ onto the heading plane

$$\Psi_\theta \triangleq \{[x\ y\ z]^T \in \Omega \mid z = t_j\} \tag{9}$$

Preprint submitted to IEEE Transactions on Automatic Control. Received: February 4, 2015 19:38:03 PST

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2405292, IEEE Transactions on Automatic Control


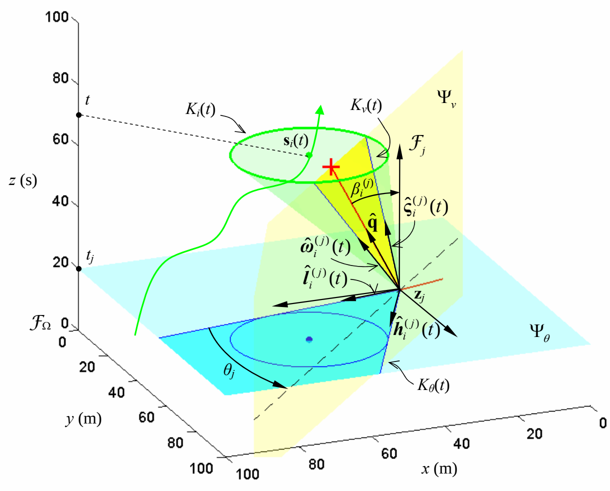


Fig. 3. Time-varying coverage cone (green) with its and heading-cone (cyan) and velocity-cone (yellow) representations (adapted from [6]).

Because $K_\theta$ is a 2D cone, it can be expressed as a linear combination of two unit vectors in $\Psi_\theta$,

$$\hat{\boldsymbol{h}}_{ij}(t) = \begin{bmatrix} \cos\alpha_{ij}(t) & -\sin\alpha_{ij}(t) \\ \sin\alpha_{ij}(t) & \cos\alpha_{ij}(t) \\ 0 & 0 \end{bmatrix} \hat{\mathbf{d}}_{ij}(t) \triangleq \begin{bmatrix} \cos\phi_{ij}(t) \\ \sin\phi_{ij}(t) \\ 0 \end{bmatrix}$$

$$\hat{\mathbf{l}}_{ij}(t) = \begin{bmatrix} \cos\alpha_{ij}(t) & \sin\alpha_{ij}(t) \\ -\sin\alpha_{ij}(t) & \cos\alpha_{ij}(t) \\ 0 & 0 \end{bmatrix} \hat{\mathbf{d}}_{ij}(t) \triangleq \begin{bmatrix} \cos\psi_{ij}(t) \\ \sin\psi_{ij}(t) \\ 0 \end{bmatrix}$$

where,

$$\hat{\mathbf{d}}_{ij}(t) = [\mathbf{s}_i(t) - \mathbf{x}_j]/\|\mathbf{s}_i(t) - \mathbf{x}_j\|$$
$$\alpha_{ij}(t) = \sin^{-1}\left(r_i/\|\mathbf{s}_i(t) - \mathbf{x}_j\|\right)$$

such that the heading cone defined with respect to a local coordinate frame $\mathcal{F}_j$ is:

$$K_\theta[\mathbf{s}_i(t), \mathbf{z}_j] \triangleq \{c_1\hat{\boldsymbol{h}}_{ij}(t) + c_2\hat{\mathbf{l}}_{ij}(t) \mid c_1, c_2 \geq 0\}, \qquad (10)$$

Examples of heading cone and heading plane are illustrated in Fig. 3, along with the unit vector representation.

The velocity cone, denoted by $K_v(t)$, contains all target speeds that lead to a detection by the $i$th sensor at any time $t \in (t_j,\ t_{j+1}]$ and, thus, it is obtained from the intersection of $K_i(t)$ with the velocity plane

$$\Psi_v \triangleq \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \Omega \;\middle|\; \begin{bmatrix} \sin\theta_j \\ \cos\theta_j \end{bmatrix}^T \begin{bmatrix} x \\ -y \end{bmatrix} = \begin{bmatrix} \sin\theta_j \\ \cos\theta_j \end{bmatrix} \mathbf{x}_j,\ z \geq t_j \right\} \qquad (11)$$

Similarly to the heading cone, $K_v$ can be represented by two unit vectors,

$$\hat{\boldsymbol{\varsigma}}_{ij}(t) = \left[\sin\eta_{ij}(t)\cos\theta_j \quad \sin\eta_{ij}(t)\sin\theta_j \quad \cos\eta_{ij}(t)\right]^T$$
$$\hat{\boldsymbol{\omega}}_{ij}(t) = \left[\sin\mu_{ij}(t)\cos\theta_j \quad \sin\mu_{ij}(t)\sin\theta_j \quad \cos\mu_{ij}(t)\right]^T$$

where,

$$\eta_{ij}(t), \mu_{ij}(t) = \tan^{-1}\left[\frac{1}{t - t_j}\Big([\cos\theta_j \;\; \sin\theta_j](\mathbf{s}_i(t) - \mathbf{x}_j) \mp \sqrt{r_i^2 - ([\sin\theta_j \;\; -\cos\theta_j](\mathbf{s}_i(t) - \mathbf{x}_j))^2}\Big)\right]$$

such that the velocity cone in $\mathcal{F}_j$ is:

$$K_v[\mathbf{s}_i(t), \mathbf{z}_j] \triangleq \{c_1\hat{\boldsymbol{\varsigma}}_{ij}(t) + c_2\hat{\boldsymbol{\omega}}_{ij}(t) \mid c_1, c_2 \geq 0\}, \qquad (12)$$

Then, the pair of cones $\{K_\theta(t), K_v(t)\}$, defined in (10) and (12), can be used to represent all tracks in $K_i(t)$, as summarized by the following lemma (adapted from [6]):

**Lemma III.2.** A target track $\mathbf{m}_j(t)$ is contained by the coverage cone $K_i(t)$ if and only if its projection in $\Psi_\theta$ is contained by the heading cone $K_\theta(t)$, and its projection in $\Psi_v$ is contained by the corresponding velocity cone $K_v(t)$.

The proof of Lemma III.2 is a simple extension of the proof in [6].

### B. SNMP Objective Function

The extremals of the heading and velocity cones presented in the previous section determine upper and lower bounds for the target heading angle and speed, respectively, that lead to a detection by the $i$th sensor, as functions of the time-varying sensor position $\mathbf{s}_i$. Let the intervals $\mathcal{H}_{ij}(t) \triangleq [\psi_{ij}(t),\ \phi_{ij}(t)]$ and $\mathcal{V}_{ij}(t) \triangleq [\tan\eta_{ij}(t),\ \tan\mu_{ij}(t)]$ respectively denote the headings and speeds contained by the heading and velocity cones. Then, the probability that the $i$th sensor detects the target at any time $t \in (t_j, t_{j+1}]$ is the probability that the Markov parameters are contained by the coverage cone $K_i(t)$,

$$P_d(i,j,t) = \int_{\mathcal{A}\times\mathcal{H}_{ij}(t)\times\mathcal{V}_{ij}(t)} f_{\mathbf{X}_j,\Theta_j,V_j}(\mathbf{x}_j, \theta_j, v_j)\, d\mathbf{x}_j d\theta_j dv_j \qquad (13)$$

where $f_{\mathbf{X}_j,\Theta_j,V_j}(\cdot)$ is the joint PDF of the Markov parameters $\mathbf{x}_j$, $\theta_j$, and $v_j$. Since these parameters are independent random variables, the probability of detection can be simplified to,

$$P_d(i,j,t) = \int_{\mathcal{A}\times\mathcal{H}_{ij}(t)\times\mathcal{V}_{ij}(t)} f_{\mathbf{X}_j}(\mathbf{x}_j) f_{\Theta_j}(\theta_j) f_{V_j}(v_j)\, d\mathbf{x}_j d\theta_j dv_j \qquad (14)$$
$$= \int_{\mathbf{x}_j\in\mathcal{A}} f_{\mathbf{X}_j}(\mathbf{x}_j) \int_{\psi_{ij}(t)}^{\phi_{ij}(t)} f_{\Theta_j}(\theta_j) \int_{\tan\eta_{ij}(t)}^{\tan\mu_{ij}(t)} f_{V_j}(v_j)\, dv_j d\theta_j d\mathbf{x}_j,\ \forall t \in (t_j, t_{j+1}]$$

It can be seen that using the 2D coverage cones reduces the region of integration from $\Omega$ to the product space $\mathcal{A}\times\mathcal{H}_{ij}(t)\times\mathcal{V}_{ij}(t)$, and thus reduces the computation required to evaluate the detection probability.

Then, the objective function for the SNMP problem can be obtained by integrating over time the probability of independent sensor detections by the $n$ sensors for all $m$ time intervals, as follows:

$$J = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\int_{t_j}^{t_{j+1}} P_d(i,j,t)\,dt$$
$$= -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\int_{t_j}^{t_{j+1}} \int_{\mathcal{A}} f_{\mathbf{X}_j}(\mathbf{x}_j) \int_{\psi_{ij}(t)}^{\phi_{ij}(t)} f_{\Theta_j}(\theta_j)$$
$$\times \int_{\tan\eta_{ij}(t)}^{\tan\mu_{ij}(t)} f_{V_j}(v_j)\, dv_j d\theta_j d\mathbf{x}_j dt \qquad (15)$$

The above objective function is to be optimized subject to the network dynamics (1) and the inequality constraints on the network state and control given by $\mathbf{c}[\mathbf{s}(t)] \leq \mathbf{0}_{(n-1)n\times 1}$ and (2), respectively. The inequality constraint on the state is defined as the vector function $\mathbf{c} = [c_{12}\cdots c_{il}\cdots c_{n(n-1)}]^T$, where,

$$c_{il} \triangleq (r_i + r_l)^2 - \|\mathbf{s}_i(t) - \mathbf{s}_l(t)\|^2, \quad i, l = 1,\ldots,n, \quad i \neq l$$

and is used to guarantee independent sensor detections (see [5] and references therein for a comprehensive treatment of detection theory). Therefore, the SNMP problem can be formulated as the following

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2405292, IEEE Transactions on Automatic Control


optimal control problem:

$$\begin{aligned} \min \quad & J, \\ \text{sbj. to} \quad & \dot{\mathbf{s}}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{B}\mathbf{u}(t), \\ & \mathbf{c}[\mathbf{s}(t)] \leq \mathbf{0}, \\ & -\mathbf{1} \leq \mathbf{u}(t) \leq \mathbf{1} \end{aligned} \tag{16}$$

Because $J$ is not quadratic, the above SNMP optimal control problem must be solved numerically for the optimal state and control trajectories $\mathbf{s}^*(t)$ and $\mathbf{u}^*(t)$. Section IV derives the SNMP EL equations, and explains how their numerical solution can be obtained via VIM. The VIM numerical simulation results and complexity analysis are presented in Section V.

## IV. Optimal Control Solution

In order to maximize the detection probability and minimize the control usage, the SNMP objective function is chosen to be of the Lagrange type, with Lagrangian

$$\mathscr{L}[\mathbf{s}(t), \mathbf{u}(t), t] = -\sum_{i=1}^{n}\sum_{j=1}^{m}\int_{\mathcal{A}} f_{\mathbf{X}_j}(\mathbf{x}_j)\int_{\psi_{ij}(t)}^{\phi_{ij}(t)} f_{\Theta_j}(\theta_j) \times \int_{\tan\eta_{ij}(t)}^{\tan\mu_{ij}(t)} f_{V_j}(v_j)dv_j d\theta_j d\mathbf{x}_j + \alpha\mathbf{u}^T\mathbf{u} \tag{17}$$

To find the necessary conditions for optimality, the Hamiltonian,

$$\begin{aligned} \mathscr{H} &\triangleq \mathscr{L}[\cdot] + \boldsymbol{\lambda}^T(t)\left[\mathbf{A}\mathbf{s}(t) + \mathbf{B}\mathbf{u}(t)\right] + \boldsymbol{\gamma}^T(t)\mathbf{c}[\mathbf{s}(t)] \\ &= \mathscr{H}[\mathbf{s}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t), \boldsymbol{\gamma}(t)] \end{aligned} \tag{18}$$

is introduced, adjoining the constraints on the state and control to (17) by means of the Lagrange multipliers $\boldsymbol{\lambda}$ and $\boldsymbol{\gamma}$.

Then, the SNMP Euler-Lagrange equations are,

$$\dot{\boldsymbol{\lambda}}(t) = -\left(\partial\mathscr{L}[\cdot]/\partial\mathbf{s}\right)^T - \mathbf{A}^T\boldsymbol{\lambda}(t) - \left(\partial\mathbf{c}[\cdot]/\partial\mathbf{s}\right)^T\boldsymbol{\gamma}(t) \tag{19}$$

$$\boldsymbol{\lambda}(T_f) = \mathbf{0} \tag{20}$$

$$\left(\partial\mathscr{L}[\cdot]/\partial\mathbf{u}\right)^T + \mathbf{B}^T\boldsymbol{\lambda}(t) + \left(\partial\mathbf{c}[\cdot]/\partial\mathbf{u}\right)^T\boldsymbol{\gamma}(t) = \mathbf{0} \tag{21}$$

where $\partial\mathscr{L}/\partial\mathbf{s} = \left[(\partial\mathscr{L}/\partial\mathbf{s}_1)^T \quad \cdots \quad (\partial\mathscr{L}/\partial\mathbf{s}_n)^T\right]^T$. Letting $\xi_{ij}, \zeta_{ij} = (\psi_{ij} \mp \phi_{ij})/2$, the partial derivatives of the Lagrangian with respect to the state can be approximated as,

$$\frac{\partial\mathscr{L}}{\partial\mathbf{s}_i} \approx \begin{bmatrix} \rho\int_{\mathcal{A}}\left\{f_{\mathbf{X}_j}(\mathbf{x}_j)\xi_{ij}(\mathbf{s}_i,\mathbf{x}_j)\sin\left[\zeta_{ij}(\mathbf{s}_i,\mathbf{x}_j)\right]\right\}d\mathbf{x}_j \\ \rho\int_{\mathcal{A}}\left\{f_{\mathbf{X}_j}(\mathbf{x}_j)\xi_{ij}(\mathbf{s}_i,\mathbf{x}_j)\cos\left[\zeta_{ij}(\mathbf{s}_i,\mathbf{x}_j)\right]\right\}d\mathbf{x}_j \end{bmatrix} \triangleq \mathbf{g}_i[\mathbf{s}_i(t)] \tag{22}$$

where $\rho = -8\ln(\pi/2)/(|V_j| \; |\Theta_j|(t - t_j))$ and $|\cdot|$ denotes the variable's range, and the partial derivative of the Lagrangian with respect to the control is $\partial\mathscr{L}/\partial\mathbf{u} = \alpha\mathbf{u}^T(t)$. Since $\partial\mathbf{c}/\partial\mathbf{u} = \mathbf{0}$, (21) simplifies to $\alpha\mathbf{u}(t) + \mathbf{B}^T\boldsymbol{\lambda} = \mathbf{0}$ and, thus,

$$\mathbf{u}(t) = -\frac{1}{\alpha}\mathbf{B}^T\boldsymbol{\lambda}(t) \tag{23}$$

Now, from the transition matrix solution of the state-space form (1), $\mathbf{s}(t) = e^{\mathbf{A}(t-T_0)}\mathbf{s}_0 + \int_{T_0}^{t}\mathbf{B}\mathbf{u}(\tau)d\tau$, and, thus, from (23) it follows that

$$\mathbf{s}(t) = e^{\mathbf{A}(t-T_0)}\mathbf{s}_0 - \frac{1}{\alpha}\int_{T_0}^{t}\mathbf{B}\mathbf{B}^T\boldsymbol{\lambda}(\tau)d\tau \tag{24}$$

Because $\boldsymbol{\gamma} = \mathbf{0}$ when $\mathbf{c}[\mathbf{s}(t)] \neq \mathbf{0}$, it also follows from (24) that the first optimality condition (19) can be simplified to,

$$\dot{\boldsymbol{\lambda}}(t) = -\mathbf{g}[\mathbf{s}(t)]\left(\int_{0}^{t}\mathbf{B}\mathbf{B}^T\boldsymbol{\lambda}(\tau)d\tau\right) - \mathbf{A}^T\boldsymbol{\lambda}(t) \tag{25}$$

TABLE I
Markov motion model probability density functions (PDFs)

| Interval $(t_j,\ t_{j+1}]$ (s) | Heading PDF $f_{\Theta_j}(\theta_j)$ | Velocity PDF $f_{V_j}(v_j)$ |
|---|---|---|
| (0, 10] (s) $(j = 1)$ | $\mathcal{U}(-\pi/3, -\pi/6)$ | $\mathcal{U}(13, 16)$ |
| (10, 20] (s) $(j = 2)$ | $\mathcal{U}(-\pi/16, \pi/16)$ | $\mathcal{U}(18, 22)$ |
| (20, 30] (s) $(j = 3)$ | $\mathcal{U}(\pi/2, 2\pi/3)$ | $\mathcal{U}(11, 14)$ |
| (30, 40] (s) $(j = 4)$ | $\mathcal{U}(-\pi/2, -\pi/3)$ | $\mathcal{U}(21, 26)$ |
| (40, 50] (s) $(j = 5)$ | $\mathcal{U}(-\pi/8, \pi/8)$ | $\mathcal{U}(10, 14)$ |

where the vector function $\mathbf{g}[\cdot] \triangleq \left[\mathbf{g}_1^T[\cdot] \quad \cdots \quad \mathbf{g}_n^T[\cdot]\right]$ is defined according to (22). Thus (25) represents a set of integro-differential equations with boundary conditions (20).

Many algorithms have been developed for solving integro-differential equations, including the Adomian decomposition method [18], the homotopy perturbation method [19], and the VIM [20]. In this paper, VIM is chosen to solve (25) because its intermediate approximations are known to converge rapidly to an accurate solution. VIM starts with a linear trial function and obtains higher order terms iteratively as follows,

$$\begin{aligned} \boldsymbol{\lambda}^{(\ell+1)}(t) = \boldsymbol{\lambda}^{(\ell)}(t) - \\ \int_{T_0}^{t}\left\{\mathbf{A}^T\boldsymbol{\lambda}^{(\ell)}(\sigma) - \mathbf{g}[\mathbf{s}(t)]\left[\int_{T_0}^{\sigma}\mathbf{B}\mathbf{B}^T\boldsymbol{\lambda}^{(\ell)}(\tau)d\tau\right]\right\}d\sigma \end{aligned} \tag{26}$$

where the superscript $\ell$ denotes the $\ell$th-order approximation.

By exploiting the integro-differential structure of the EL equations, VIM can significantly reduce computational complexity when compared to direct methods of solution. In direct methods, the dynamic equation and objective function are discretized and transcribed into an NLP that, typically, is solved using sequential quadratic programming (SQP) [21]. The computational complexity of SQP direct methods is $O(n^3K^3M)$, where $n$ is the number of sensors, $K$ is the number of collocation points, and $M$ is the number of iterations required for convergence [21]. The indirect VIM, on the other hand, requires a computation time of $O(nK^2)$ to evaluate (26) using Euler integration. Therefore, the computation complexity for VIM is $O(nK^2M)$, where in practice $M$ is quite small. Therefore, the VIM solution is efficient for mobile sensor networks with a few dozen sensors. For larger $n$, efficient solutions can be obtained by combing the results in this paper with the distributed optimal control approach presented in [22].

## V. Simulation Results

Consider the Markov motion model in Table I for a target traversing the RoI over a time interval $(T_0,\ T_f] = (0, 50](s)$, where $m = 5$. At $t_1 = T_0 = 0$ (s), the PDF of the target position, $f_{\mathbf{X}_1}(\mathbf{x}_1)$, is a 2D multivariate Gaussian distribution, denoted by $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, with mean $\boldsymbol{\mu} = [20 \quad 180]^T$(m) and covariance matrix $\boldsymbol{\Sigma} = \text{diag}([10 \quad 10])(\text{m}^2)$, where, diag$(\cdot)$ denotes an operator that places a row vector on the diagonal of a zero matrix. The heading and velocity PDFs are uniform distributions, denoted by $\mathcal{U}(a, b)$, with support $[a, b]$, as shown in Table I. Then, the PDFs of $\mathbf{x}_2, \ldots, \mathbf{x}_5$, can be computed recursively, as shown in [6].

Simulation results are presented for two example cases, one network with $n = 9$ and $r_i = 6$ (m) (Fig. 4), and one network with $n = 20$ and $r_i = 5$ (m) (Fig. 5). Figures 4-5 show the sensor trajectories and FoVs, and the PDF of the target position, at four sample instants in time. It can be seen that by the geometric transversals approach the sensors plan their motion such that the detection probability in $(T_0,\ T_f]$ is maximized. The optimal control histories of a randomly chosen sensor (red arrow in Fig. 5) are plotted

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2405292, IEEE Transactions on Automatic Control


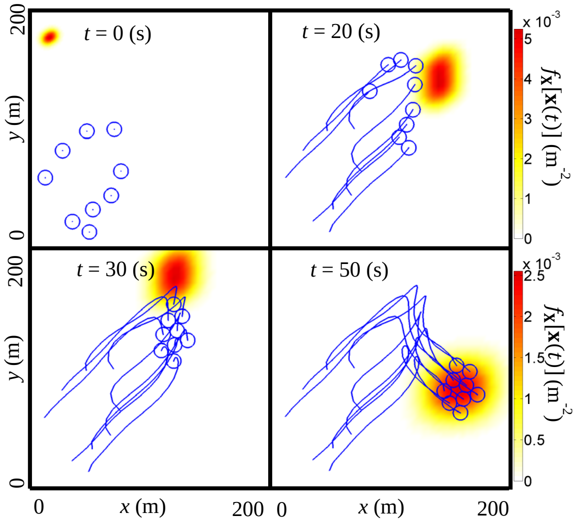


Fig. 4. Optimal sensor trajectories for $n = 9$ and $r_i = 6$ (m), given the target motion model in Table I.

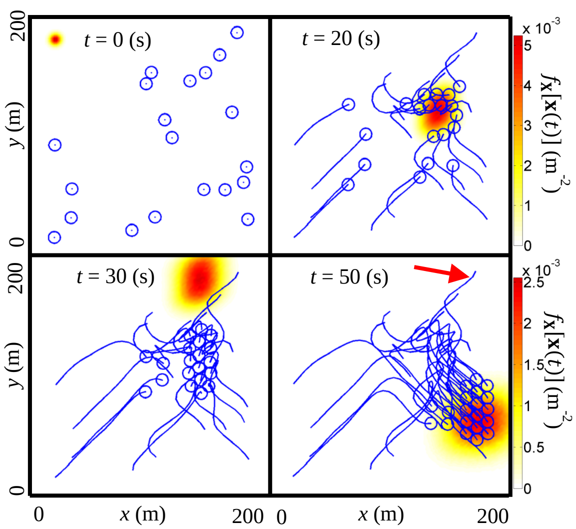


Fig. 5. Optimal sensor trajectories for $n = 20$ and $r_i = 5$ (m), given the target motion model in Table I.

in Fig. 6 to illustrate that control inputs obtained by this approach are smooth and obey the desired bounds in (2).

The effectiveness of the geometric transversals approach is illustrated by comparing the probability of detection obtained by the network in Fig. 5 to that obtained by potential field, greedy, uniform grid, and random algorithms. In potential field [23], the PDF of the target position is used to build an attractive potential, and a repulsive force $f_r = -c_r/\|\mathbf{s}_i(t) - \mathbf{s}_j(t)\|^2$ is used to prevent collisions between sensors, where $c_r = 1$ [23]. The greedy algorithm proposed in [24] places the sensors at $n$ fixed locations, such that the network coverage is maximized while retaining line-of-sight relationships between sensors. The grid and random algorithms proposed in [25] place the sensors at $n$ fixed locations in $\mathcal{A}$ according to a uniformly spaced grid or by sampling a uniform distribution.

The results in Fig. 7 are representative of extensive simulations performed using different sensor networks, target models, and initial conditions. Because the network performance is highly sensitive to initial conditions, the average probability of detection, denoted by $P_e$, is computed by considering over 100 initial conditions, sampled

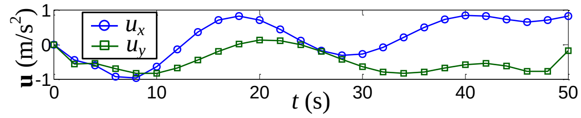


Fig. 6. Optimal control histories of one sensor chosen at random from the network in Fig. 5 (as shown by red arrow).

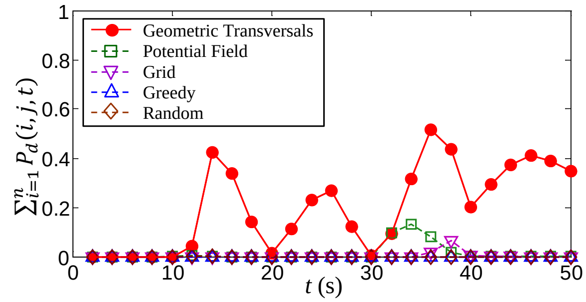


Fig. 7. Performance comparison for sensor network in Fig. 5, with $n = 20$, $r_i = 5$ (m), and the target motion model in Table I.

uniformly at random from the RoI, holding network and target parameters constant. The mean performance ($P_e$) and three standard deviations (SDs) obtained by the five algorithms are plotted in Fig. 8 and show that the geometric transversals approach significantly outperforms other algorithms over the entire time interval $(T_0, T_f]$.

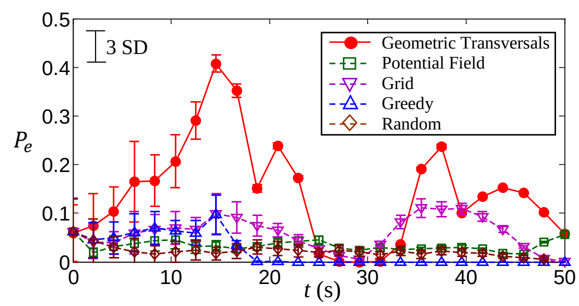


Fig. 8. Probability of detection averaged over 100 initial conditions for $n = 20$, $r_i = 5$ (m), and the target motion model in Table I.

## VI. Summary and Conclusions

This paper presents a geometric transversals approach for planning the motion of a mobile sensor network such that its detection probability is maximized over time. By this approach, the approach derives a track coverage objective function in closed form, based on the transition PDFs of the target Markov motion model. By this novel approach, the probability of detection can be optimized subject to the sensor kinodynamic equations, and inequality constraints on the sensor state and control. The necessary conditions for optimality are derived and reduced to a set of integro-differential equations that are solved numerically using a variational iteration method. The results show that by this approach the computational complexity is significantly reduced compared to a direct method, and the detection probability is significantly increased compared to existing potential field, greedy, grid, or random algorithms.

## References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2015.2405292, IEEE Transactions on Automatic Control




[2] D. Chen and P. K. Varshney, "Qos support in wireless sensor networks: A survey," in *International Conference on Wireless Networks*, vol. 233, 2004.

[3] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 39, no. 4, pp. 1333–1364, 2003.

[4] K. Baumgartner and S. Ferrari, "A geometric transversal approach to analyzing track coverage in sensor networks," *Computers, IEEE Transactions on*, vol. 57, no. 8, pp. 1113 –1128, Aug. 2008.

[5] T. A. Wettergren, "Performance of search via track-before-detect for distributed sensor networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 314–325, January 2008.

[6] H. Wei and S. Ferrari, "A geometric transversals approach to analyzing the probability of track detection for maneuvering targets," *Computers, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2013.

[7] Y. Bar-Shalom and X. R. Li, *Multitarget-multisensor Tracking: Principles and Techniques, 1995*. Yaakov Bar-Shalom, 1995.

[8] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[9] H. Wei, W. Lu, P. Zhu, G. Huang, J. Leonard, and S. Ferrari, "Optimized visibility motion planning for target tracking and localization," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 76–82.

[10] J. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[11] Y. Bar-Shalom and X.-R. Li, *Estimation and tracking: principles, techniques, and software*. Artech House Norwood, 1993, vol. 393.

[12] H. Wei, W. Ross, S. Varisco, P. Krief, and S. Ferrari, "Modeling of human driver behavior via receding horizon and artificial neural network controllers," in *Decision and Control (CDC), IEEE Annual Conference on*, Florence, Italy, Dec 2013, pp. 6778–6785.

[13] B. Liu and D. Towsley, "A study of the coverage of large-scale sensor networks," in *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, 2004, pp. 475–483.

[14] H. Wei, W. Lu, and S. Ferrari, "An information value function for nonparametric gaussian processes," in *Proc. Neural Information Processing Systems Conference*. Lake Tahoe, NV: NIPS, 2012.

[15] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292 – 2330, 2008.

[16] H. Wei, W. Lu, P. Zhu, S. Ferrari, R. H. Klein, S. Omidshafiei, and J. P. How, "Camera control for learning nonlinear target dynamics via bayesian nonparametric dirichlet-process gaussian-process (DP-GP) models," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 95–102.

[17] S. Meyn and R. L. Tweedie, *Markov chains and stochastic stability; 2nd ed.*, ser. Cambridge Mathematical Library. Leiden: Cambridge Univ. Press, 2009.

[18] G. Adomian, "A review of the decomposition method and some recent results for nonlinear equations," *Mathematical and Computer Modelling*, vol. 13, no. 7, pp. 17–43, 1990.

[19] J.-H. He, "Homotopy perturbation technique," *Computer methods in applied mechanics and engineering*, vol. 178, no. 3, pp. 257–262, 1999.

[20] B. Neta, "Numerical solution of a nonlinear integro-differential equation," *Journal of Mathematical Analysis and Applications*, vol. 89, no. 2, pp. 598–611, 1982.

[21] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations," *Numerical Analysis*, vol. 630, 1978.

[22] G. Foderaro, S. Ferrari, and T. Wettergren, "Distributed optimal control for multi-agent trajectory optimization," *Automatica*, vol. 50, pp. 149–154, 2014.

[23] S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots*, vol. 13, no. 3, pp. 207–222, 2002.

[24] A. Howard, M. J. Matari, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, pp. 113–126, 2002.

[25] H. Gonzales-Banos and J. C. Latombe, "Robot motion planning with uncertainty in control and sensing," $17^{th}$ *Annual Symposium on Computational Geometry (SCG'01)*, 2001.