

Optimal Event-Driven Multiagent Persistent Monitoring of a Finite Set of Data Sources

Nan Zhou^{ID}, *Student Member, IEEE*, Xi Yu^{ID}, *Student Member, IEEE*,
Sean B. Andersson^{ID}, *Senior Member, IEEE*, and Christos G. Cassandras^{ID}, *Fellow, IEEE*

Abstract—We consider the problem of controlling the movement of multiple cooperating agents so as to minimize an uncertainty metric associated with a finite number of data sources. In a one-dimensional (1-D) mission space, we adopt an optimal control framework and show that the solution can be reduced to a simpler parametric optimization problem: Determining a sequence of locations where each agent may dwell for a finite amount of time and then switch direction. This amounts to a hybrid system which we analyze using the infinitesimal perturbation analysis (IPA) to obtain a complete online solution through an event-driven gradient-based algorithm which is also robust with respect to the uncertainty model used. The resulting controller depends on observing the events required to excite the gradient-based algorithm, which cannot be guaranteed. We solve this problem by introducing a new metric for the objective function which creates a potential field guaranteeing that gradient values are nonzero. This approach is compared to an alternative graph-based target-visit scheduling and dwell times optimization algorithm. The simulation examples are included to demonstrate the proposed methods.

Index Terms—Agents and autonomous systems, cooperative control, hybrid systems, optimization.

I. INTRODUCTION

SYSTEMS consisting of cooperating mobile agents are often used to perform tasks such as coverage control [1], [2], surveillance, and environmental sampling. The persistent monitoring problem arises when agents are assigned to monitor a

dynamically changing environment which cannot be fully covered by a stationary agent allocation. Thus, persistent monitoring differs from traditional coverage tasks due to the perpetual need to cover a changing environment. This exploration process results in the eventual discovery of various “points of interest” which, once detected, become “data sources” or “targets” which need to be monitored. This setting arises in multiple application domains ranging from large-scale surveillance, environmental monitoring, and energy management [3], [4] in smart cities down to particle tracking in nanoscale systems tasked to the study of dynamic and interactive processes in biomolecular systems and in nanomedical research [5]–[7]. In contrast to patrol strategies for sweep coverage [8]–[11] or to discover intruders/new targets [12], [13] where every point in a mission space must be continually surveyed, the problem we address here involves a finite number of known data sources (typically larger than the number of agents and we will refer to them as “targets” for short) which the agents must cooperatively monitor through periodic visits.

The state of each target is observed and controlled by agents equipped with sensing capabilities (e.g., cameras) and which are normally dependent upon their physical distance from the target. The objective of cooperative persistent monitoring in this case is to minimize an overall measure of uncertainty about the target states. This may be accomplished by assigning the agents to specific targets dynamically or by a periodic scheduling approach of designing motion trajectories through which agents reduce the uncertainty state of a target by visiting it (and possibly remaining at the target for a finite amount of time) until a certain switching condition is met [14]. Viewed as an optimization problem, the goal is for the agents to jointly minimize some cost function that captures the desired features of the monitoring task [15]. The key problem is determining for each agent the sequence of target visits and the associated dwell time at each target. As long as the numbers of agents and targets are small, it is possible to identify sequences that yield a globally optimal solution; in general, however, this is a computationally intensive procedure which does not scale well [16].

Rather than viewing this problem as a target visiting task which eventually falls within the class of traveling salesman [17] or vehicle routing problems [18], in this paper we follow earlier work in [10] and introduce an optimal control framework whose objective is to control the movement of agents so as to collect information from targets and ultimately minimize an average metric of uncertainty over all targets. An important

Manuscript received September 11, 2017; revised December 8, 2017; accepted January 24, 2018. Date of publication April 23, 2018; date of current version December 3, 2018. This work was supported in part by NSF under Grants CNS-1239021, ECCS-1509084, and IIP-1430145, in part by AFOSR under Grant FA9550-15-1-0471, and in part by ONR under Grant N00014-09-1-1051 and in part by the NSF through Grant ECCS-1509084. Recommended by Associate Editor M. Kanat. (*Corresponding author: Nan Zhou.*)

N. Zhou is with the Division of Systems Engineering, Boston University, Boston, MA, USA (e-mail: nanzhou@bu.edu).

X. Yu is with the Department of Mechanical Engineering, Boston University, Boston, MA, USA (e-mail: xyu@bu.edu).

S. B. Andersson is with the Division of Systems Engineering, Boston University, Boston, MA, USA, and also with the Department of Mechanical Engineering, Boston University, Boston, MA, USA (e-mail: anderss@bu.edu).

C. G. Cassandras is with the Division of Systems Engineering, Boston University, Boston, MA, USA and also with the Department of Electrical and Computer Engineering, Boston University, Boston, MA, USA (e-mail: cgc@bu.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2018.2829469

difference between the previous work [10] and the current persistent monitoring setting is that there is now a finite number of targets that agents need to monitor as opposed to every point in the mission space. In a 1-D mission space, we show that the optimal control problem can be reduced to a parametric optimization problem. In particular, every optimal agent trajectory is characterized by a finite number of points where the agent switches direction and by a dwelling time at each such point. As a result, the behavior of the agents under optimal control is described by a hybrid system whose behavior is captured by agent control switches and states of the targets. This allows us to make use of infinitesimal perturbation analysis (IPA) [19], [20] to determine online the gradient of the objective function with respect to these parameters and to obtain a (possibly local) optimal trajectory. Our approach exploits an inherent property of IPA under mild conditions which allows virtually arbitrary stochastic effects in modeling target uncertainty. Moreover, IPA's event-driven nature renders it scalable in the number of events in the system and not its state space.

A potential drawback of event-driven control methods is that they obviously depend on the events which “excite” the underlying controller being observable. However, this is not guaranteed under every feasible control: It is possible that no such events are excited under a nominal control in which case the controller may be useless. The crucial events in persistent monitoring are “target visits” and it is possible that such events may never occur for a large number of feasible agent trajectories which IPA uses to estimate a gradient online, especially when targets are sparse and the corresponding gradient field is flat. At the heart of this problem is the fact that the objective function we define for a persistent monitoring problem has a nonzero cost metric associated with only a subset of the mission space centered around targets, while all other points have zero cost, since they are not “points of interest”. This lack of event excitation is a serious problem in many trajectory planning and optimization tasks [21]–[23]. Thus, to solve this problem, we propose a new cost metric introduced in [24] which creates a potential field based on the existing targets guaranteeing that gradient values are generally nonzero throughout the mission space and ensures that all events are ultimately excited.

To summarize, the contributions of this paper consist of:

- 1) presenting new results on characterizing the optimal trajectories of agents when a finite number of targets is known (in contrast to [10] where targets are unknown);
- 2) providing a globally optimal solution to the problem by using a graph-based scheduling method as a baseline for assessing the performance of the IPA gradient scheme we use to determine agent optimal trajectories;
- 3) addressing the potential lack of event excitation in an event-driven optimization approach as described above.

The rest of the paper is organized as follows. Section II formulates persistent monitoring as an optimal control problem and Section III presents a Hamiltonian analysis which characterizes the optimal solution in terms of two sets of parameter vectors specifying switching points and associated dwelling times for each agent. Section IV provides a complete solution obtained through event-driven IPA gradient estimation, and solves the

issue of potential lack of event excitation through a modified cost metric. In Section V, we present a graph-based scheduling approach as an alternative aimed at finding a global optimum and comparing it with the IPA-based solution. Section VI includes several simulation results and Section VII concludes the paper.

II. PERSISTENT MONITORING PROBLEM FORMULATION

We begin by reviewing the model and problem formulation introduced in [25] before providing a complete analysis of its solution. We consider N mobile agents moving in a 1-D mission space $[0, L] \subset \mathbb{R}$ and maintaining a fully connected network. Let the position of the agents at time t be $s_j(t) \in [0, L]$, $j = 1, \dots, N$, following the dynamics:

$$\dot{s}_j(t) = u_j(t) \quad (1)$$

i.e., we assume that an agent j can control its direction and speed. Without the loss of generality, after proper rescaling, we further assume that the speed is constrained by $|u_j(t)| \leq 1$, $j = 1, \dots, N$. As will become clear, the agent dynamics in (1) can be replaced by a more general model of the form $\dot{s}_j(t) = g_j(s_j(t)) + b_j u_j(t)$ without affecting the main results of our analysis. Finally, for convenience we label agents $1, \dots, N$ sequentially according to their initial positions $s_1(0) \leq s_2(0) \leq \dots \leq s_N(0)$ and we will show that this ordering is preserved throughout an optimal trajectory for all $j = 1, \dots, N - 1$ as follows:

$$s_j(t) - s_{j+1}(t) \leq 0. \quad (2)$$

The ability of an agent to sense its environment is modeled by a function $p_j(x, s_j)$ that measures the probability that an event at location $x \in [0, L]$ is detected by agent j . We assume that $p_j(x, s_j) = 1$ if $x = s_j$, and that $p_j(x, s_j)$ is monotonically nonincreasing in the distance $|x - s_j|$, thus capturing the reduced effectiveness of a sensor over its range which we consider to be finite and denoted by r_j . Therefore, we set $p_j(x, s_j) = 0$ when $|x - s_j| > r_j$. Although our analysis is not affected by the precise sensing model $p_j(x, s_j)$, we will consider a linear decay model as follows:

$$p_j(x, s_j) = \max \left\{ 1 - \frac{|s_j - x|}{r_j}, 0 \right\} \quad (3)$$

and limit ourselves to continuous sensing functions for simplicity, although this is not required for the subsequent analysis to hold. Unlike the persistent monitoring problem setting in [10], here we consider a known finite set of targets located at $x_i \in (0, L)$, $i = 1, \dots, M$ (we assume $M > N$ to avoid uninteresting cases where there are at least as many agents as targets, in which case every target can be assigned to at least one agent). We can then set $p_j(x_i, s_j(t)) \equiv p_{ij}(s_j(t))$ to represent the effectiveness with which agent j can sense target i when located at $s_j(t)$. Accordingly, the joint probability that $x_i \in (0, L)$ is sensed by at least one agent (assuming detection independence) is

$$P_i(s(t)) = 1 - \prod_{j=1}^N [1 - p_{ij}(s_j(t))] \quad (4)$$

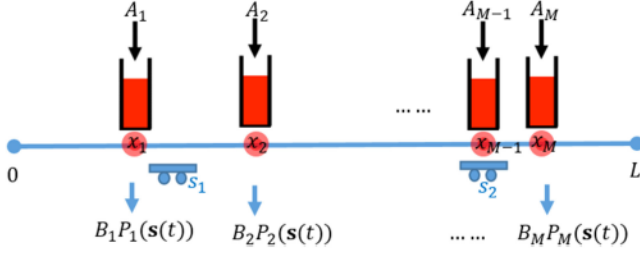


Fig. 1. Polling model interpretation of problem P1.

where we set $\mathbf{s}(t) = [s_1(t), \dots, s_N(t)]^T$.

Next, we define uncertainty functions $R_i(t)$ associated with targets $i = 1, \dots, M$, so that they have the following properties:

- 1) $R_i(t)$ increases with a prespecified rate A_i if $P_i(\mathbf{s}(t)) = 0$ (we will later allow this to be a random process $\{A_i(t)\}$),
- 2) $R_i(t)$ decreases with a fixed rate B_i if $P_i(\mathbf{s}(t)) = 1$, and
- 3) $R_i(t) \geq 0$ for all t . It is then natural to model uncertainty dynamics associated with each target as follows:

$$\dot{R}_i(t) = \begin{cases} 0 & \text{if } R_i(t) = 0 \text{ and } A_i \leq B_i P_i(\mathbf{s}(t)) \\ A_i - B_i P_i(\mathbf{s}(t)) & \text{otherwise} \end{cases} \quad (5)$$

where we assume that initial conditions $R_i(0)$, $i = 1, \dots, M$, are given and that $B_i > A_i > 0$ (thus, the uncertainty strictly decreases when there is perfect sensing $P_i(\mathbf{s}(t)) = 1$).

Our goal is to control the movement of the N agents through $u_j(t)$ in (1) so that the cumulative average uncertainty over all targets $i = 1, \dots, M$ is minimized over a fixed time horizon T . Thus, setting $\mathbf{u}(t) = [u_1(t), \dots, u_N(t)]^T$ we aim to solve the following optimal control problem P1:

$$\min_{\mathbf{u}(t)} J = \frac{1}{T} \int_0^T \sum_{i=1}^M R_i(t) dt \quad (6)$$

subject to the agent dynamics (1), uncertainty dynamics (5), control constraint $|u_j(t)| \leq 1$, $t \in [0, T]$, and state constraints (2). Fig. 1 shows a polling model version for problem P1 where each target is associated with a “virtual queue” where uncertainty accumulates with the inflow rate A_i . The service rate of this queue is time-varying and given by $B_i P_i(\mathbf{s}(t))$, controllable through the agent position at time t . This interpretation is convenient for characterizing the stability of such a system over a mission time T : For each queue, we may require that $\int_0^T A_i dt < \int_0^T B_i P_i(\mathbf{s}(t)) dt$. Alternatively, we may require that each queue becomes empty at least once over $[0, T]$. Note that, this analogy readily extends to two or 3-D settings.

III. OPTIMAL CONTROL SOLUTION

In this section, we derive properties of the optimal control solution of problem P1 and show that it can be reduced to a parametric optimization problem. This will allow us to utilize an IPA gradient estimation approach [19] to find a complete optimal solution through a gradient-based algorithm. We begin by defining the state vector $\mathbf{x}(t) = [R_1(t), \dots, R_M(t),$

$s_1(t), \dots, s_N(t)]$ and associated costate vector $\lambda = [\lambda_1(t), \dots, \lambda_M(t), \lambda_{s_1}(t), \dots, \lambda_{s_N}(t)]$. As in [10], due to the discontinuity in the dynamics of $R_i(t)$ in (5), the optimal state trajectory may contain a boundary arc when $R_i(t) = 0$ for some i ; otherwise, the state evolves in an interior arc. Thus, we first analyze such an interior arc. Using (1) and (5), the Hamiltonian is

$$H(\mathbf{x}, \lambda, \mathbf{u}) = \sum_{i=1}^M R_i(t) + \sum_{i=1}^M \lambda_i(t) \dot{R}_i(t) + \sum_{j=1}^N \lambda_{s_j}(t) u_j(t). \quad (7)$$

The costate dynamics are

$$\begin{aligned} \dot{\lambda}_i(t) &= -\frac{\partial H}{\partial R_i(t)} = -1, \quad \lambda_i(T) = 0; \\ \dot{\lambda}_{s_j}(t) &= -\frac{\partial H}{\partial s_j(t)} = \sum_{i=1}^M \lambda_i(t) B_i \frac{\partial P_i(\mathbf{s}(t))}{\partial s_j(t)}, \quad \lambda_{s_j}(T) = 0. \end{aligned} \quad (8)$$

Applying the Pontryagin minimum principle to (7) with $\mathbf{u}^*(t)$, $t \in [0, T]$, denoting an optimal control, a necessary condition for optimality is

$$H(\mathbf{x}^*, \lambda^*, \mathbf{u}^*) = \min_{\mathbf{u}(t)} H(\mathbf{x}, \lambda, \mathbf{u}) \quad (10)$$

from which it immediately follows that

$$u_j^*(t) = \begin{cases} 1 & \text{if } \lambda_{s_j}(t) < 0 \\ -1 & \text{if } \lambda_{s_j}(t) > 0 \end{cases}. \quad (11)$$

Note that, there exists a possibility that $\lambda_{s_j}(t) = 0$ over some finite singular intervals [26], in which case $u_j^*(t)$ may take values in $\{-1, 0, 1\}$. This requires further analysis, in particular we show in Lemma 2 that $u_j^*(t) = 0$ when $\lambda_{s_j}(t) = 0$.

Similar to the case of the persistent monitoring problem studied in [10], the complete solution requires solving the state and costate equations, which in turn involves the determination of all points where $R_i(t) = 0$, $i = 1, \dots, M$. This generally involves the solution of a two-point-boundary-value problem. However, we will next prove some structural properties of an optimal trajectory, based on which we show that it is fully characterized by two sets of parameters, thus reducing the optimal control problem to a much simpler parametric optimization problem.

We begin by assuming that targets are ordered according to their location, so that $x_1 < \dots < x_M$. Let $r = \max_{j=1, \dots, N} \{r_j\}$, $a = \max\{0, x_1 - r\}$, and $b = \min\{L, x_M + r\}$. Thus, if $s_j(t) < x_1 - r$ or $s_j(t) > x_M + r$, then it follows from (3) that $p_{ij}(s_j(t)) = 0$ for all targets $i = 1, \dots, M$. Clearly, this implies that the effective mission space is $[a, b]$, i.e.,

$$a \leq s_j(t) \leq b, \quad j = 1, \dots, N \quad (12)$$

imposing an additional state constraint for P1. We will show next that on an optimal trajectory every agent is constrained to move within the interval $[x_1, x_M]$. This implies that every agent must switch its direction no later than reaching the first or last target (possibly after dwelling at the switching point for a finite

time interval). To establish this and subsequent results, we first define the following.

Definition III.1: An agent switches direction at time t when the following conditions hold: There exists $t_0 \in [0, t)$ such that $u_j(t_0) \neq 0$; $u_j(\tau)u_j(t_0) \geq 0$ for all $\tau \in [t_0, t)$; and $u_j(t^+)u_j(t_0) < 0$.

In contrast, we define the agent control switching points as follows.

Definition III.2: A control switching point of agent j is $s_j(t) \in [a, b]$ such that $u_j(t^-) \neq u_j(t^+)$, $t \in (0, T)$.

Next, we will make a technical assumption that no two events altering the dynamics in (1) and (5), respectively, can occur at the exact same time when an agent switches direction. This will simplify the subsequent derivations without restricting the implementations presented in Section VI.

Assumption 1: Suppose that an agent switches direction at $\theta \in [a, b]$. For any $j = 1, \dots, N$, $i = 1, \dots, M$, $t \in (0, T)$, there exists $\epsilon > 0$, such that if $s_j(t) = \theta$, $s_j(t - \epsilon) > \theta$, or if $s_j(t) = \theta$, $s_j(t - \epsilon) < \theta$, then either $R_i(\tau) > 0$ for all $\tau \in [t - \epsilon, t]$ or $R_i(\tau) = 0$ for all $\tau \in [t - \epsilon, t]$.

Proposition 1: In an optimal trajectory, if $x_1 \leq s_j^*(0) \leq x_M$, then $x_1 \leq s_j^*(t) \leq x_M$, $t \in [0, T]$, $j = 1, \dots, N$.

Proof: We first prove that $s_j^*(t) \geq x_1$ for any agent j . Suppose that $s_j^*(t_0) = x_1$ and $u_j^*(t_0) = -1$. In view of (12), assume that agent j reaches a point $\theta \in [a, x_1)$ at time $t_1 > t_0$ where it switches direction, we will show that $\theta \notin [a, x_1)$ using a contradiction argument. There are two cases to consider. ■

Case 1: $\theta = a$. Assuming $s_j^*(t_1) = a$, we first show that $\lambda_{s_j}^*(t_1^-) = 0$ by a contradiction argument. If $\lambda_{s_j}^*(t_1^-) \neq 0$, recall that $u_j^*(t_1^-) = -1$, therefore $\lambda_{s_j}^*(t_1^-) > 0$ from (11). Since the constraint $a - s_j(t) \leq 0$ is active, $\lambda_{s_j}^*(t)$ may experience a discontinuity so that

$$\lambda_{s_j}^*(t_1^-) = \lambda_{s_j}^*(t_1^+) - \pi_j \quad (13)$$

where $\pi_j \geq 0$ is a scalar multiplier associated with the constraint $a - s_j(t) \leq 0$. It follows that $\lambda_{s_j}^*(t_1^+) = \lambda_{s_j}^*(t_1^-) + \pi_j > 0$. Since the Hamiltonian in (7) and the constraint $a - s_j(t) \leq 0$ are not explicit functions of time, we have [26] $H^*(x(t_1^-), \lambda(t_1^-), u(t_1^-)) = H^*(x(t_1^+), \lambda(t_1^+), u(t_1^+))$ which, under Assumption 1, reduces to

$$\lambda_{s_j}^*(t_1^-)u_j^*(t_1^-) = \lambda_{s_j}^*(t_1^+)u_j^*(t_1^+). \quad (14)$$

Recall that $\lambda_{s_j}^*(t_1^-)u_j^*(t_1^-) < 0$. However, $u_j^*(t_1^+) \geq 0$ (since the agent switches control), therefore, $\lambda_{s_j}^*(t_1^+)u_j^*(t_1^+) \geq 0$ which violates (14). This contradiction implies that $\lambda_{s_j}^*(t_1^-) = 0$. Recalling (4) and (9), we get $\dot{\lambda}_{s_j}^*(t_1^-) = \sum_{i=1, R_i \neq 0}^M \lambda_i^*(t_1^-) \frac{B_i}{r_j} \prod_{d \neq j} [1 - p_{id}(s_d^*(t_1^-))]$. Under Assumption 1, there exists $\delta > 0$ such that during interval $(t_1 - \delta, t_1)$, no $R_i(t) \geq 0$ becomes active, hence, no $\lambda_i^*(t)$ encounters a jump for $i = 1, \dots, M$ and it follows from (8) that $\lambda_i^*(t) > 0$. Moreover, $p_{id}(s_d^*(t)) \neq 1$ for at least some $d \neq j$ since we have assumed that $M > N$. Thus, we have $\dot{\lambda}_{s_j}^*(t) > 0$, for all $t \in (t_1 - \delta, t_1)$. However, since agent j is approaching a , there exists some $\delta' < \delta$, such that $u_j^*(t) = -1$ for all $t \in (t_1 - \delta', t_1)$, and $\lambda_{s_j}^*(t) \geq 0$. Thus for $t \in (t_1 - \delta', t_1)$, we have $\lambda_{s_j}^*(t) \geq 0$ and $\dot{\lambda}_{s_j}^*(t) > 0$.

This contradicts the established fact that $\lambda_{s_j}^*(t_1^-) = 0$. We conclude that $\theta \neq a$.

Case 2: $\theta \in (a, x_1)$. Assuming $s_j^*(t_1) = \theta$, we still have $u_j^*(t_1^-) = -1$, $u_j^*(t_1^+) \geq 0$. Since the Hamiltonian (7) is not an explicit function of time, we have $H^*(x(t_1^-), \lambda(t_1^-), u(t_1^-)) = H^*(x(t_1^+), \lambda(t_1^+), u(t_1^+))$ which leads to (14) under Assumption 1. First, we assume $\lambda_{s_j}^*(t_1^-) \neq 0$. Since $u_j^*(t_1^-) < 0$, we have $\lambda_{s_j}^*(t_1^-) > 0$ and the left hand side of (14) gives $\lambda_{s_j}^*(t_1^-)u_j^*(t_1^-) < 0$. On the other hand, in order to satisfy (14), we must have $u_j^*(t_1^+) > 0$ and $\lambda_{s_j}^*(t_1^+) < 0$. However, if $\lambda_{s_j}^*(t_1^-) > 0$ and $\lambda_{s_j}^*(t_1^+) < 0$, then either $\dot{\lambda}_{s_j}^*(t_1) < 0$ and $\lambda_{s_j}^*(t_1) = 0$, or $\lambda_{s_j}^*(t)$ experiences a discontinuity at t_1 . We show that neither condition is feasible. The first one violates our assumption that $\lambda_{s_j}^*(t_1) \neq 0$, while the second one is not feasible since at $t = t_1$ the constraint $a - s_j(t) \leq 0$ is not active. This implies that $\lambda_{s_j}^*(t_1^-) = 0$. Again, under Assumption 1, the same argument as in Case 1 can be used to show that $\lambda_{s_j}^*(t) \geq 0$ and $\dot{\lambda}_{s_j}^*(t) > 0$ for all $t \in (t_1 - \delta', t_1)$. This contradicts the established fact that $\lambda_{s_j}^*(t_1^-) = 0$ and we conclude that $\theta \notin (a, x_1)$.

Combining both the cases, we conclude that $\theta \notin [a, x_1)$, which implies that $s_j^*(t) \geq x_1$. The same line of argument can be used to show that $s_j^*(t) \leq x_M$. ■

Proposition 1, in conjunction with (11), leads to the conclusion that the optimal control consists of each agent moving with maximal speed in one direction until it reaches a point in the interval $[x_1, x_M]$ where it switches direction. Note that, this property applies to the problem in [10] as well, however, here we need to additionally prove that $s_j(t) \notin (a, x_1)$ as shown in Case 2 on the proof. However, the exclusion of the case $\lambda_{s_j}(t) = 0$ allows the possibility of singular arcs along the optimal trajectory, defined as intervals $[t_1, t_2]$ such that $\lambda_{s_j}(t) = 0$ for all $t \in [t_1, t_2]$ and $\lambda_{s_j}(t_1^-) \neq 0$, $\lambda_{s_j}(t_2^+) \neq 0$. The next result establishes the fact that we can exclude singular arcs from an agent's trajectory while this agent has no target in its sensing range.

Lemma 1: If $|s_j(t) - x_i| > r_j$ for $t \in [0, T]$, any target $i = 1, \dots, M$, and agent $j = 1, \dots, N$, then $u_j^*(t) \neq 0$.

Proof: We proceed with a contradiction argument. Suppose that $u_j^*(t) = 0$ for $t \in [t_1, t_2]$, such that $|s_j^*(t_1) - x_i| > r_j$ for all $i = 1, \dots, M$ and that $u_j^*(t) \neq 0$ (without the loss of generality, let $u_j^*(t) = 1$ for $t > t_2$, so that $|s_j^*(t_3) - x_i| = r_j$ for some $i = 1, \dots, M$ and $|s_j^*(t_3 + \Delta) - x_i| < r_j$ for $t_3 + \Delta > t_3 > t_2$. In other words, agent j eventually reaches a target i that it can sense at $t = t_3$. Assume that $u_j^*(t)$, $t \in [t_1, t_3 + \Delta]$ is replaced by $u_j'(t)$ as follows: $u_j'(t) = 1$ for $t \in [t_1, t_3 + \Delta + t_1 - t_2]$ and $u_j'(t) = 0$ for $t \in (t_3 + \Delta + t_1 - t_2, t_3 + \Delta]$. In other words, the agent moves to reach $s_j'(t_3 + \Delta + t_1 - t_2) = s_j^*(t_3 + \Delta)$ and then stops. The two controls are thereafter identical, as illustrated in Fig. 2. Then, referring to (6) we have $\int_{t_3 + \Delta + t_1 - t_2}^{t_3 + \Delta} R_i'(t) dt \leq \int_{t_3 + \Delta + t_1 - t_2}^{t_3 + \Delta} R_i^*(t) dt$ since under $u_j'(t)$ the agent may decrease $R_i(t)$ over $[t_3 + \Delta + t_1 - t_2, t_3]$, whereas under $u_j^*(t)$ this is impossible since $|s_j^*(t) - x_i| > r_j$ over this time interval. Since the cost in (6) is the same over $[0, t_3 + \Delta + t_1 - t_2)$ and $(t_3 + \Delta, T]$, it follows that $u_j^*(t) = 0$ when $|s_j(t) - x_i| > r_j$ cannot be optimal unless $u_j^*(t) = 0$ for

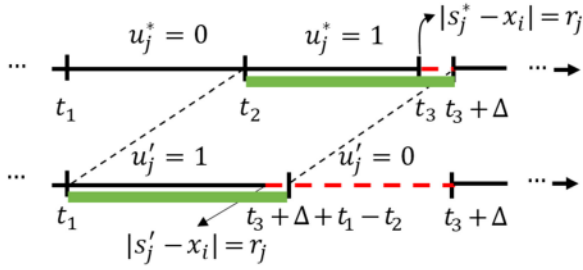


Fig. 2. Illustration of the control strategies compared in the proof of Lemma 1. The dashed red lines indicate the segments when the target is within the agent's sensing range and the thick green bars indicate the segments when the control takes the value 1.

all $t \in [0, T]$, i.e., the agent never moves and never senses any target, in which case the cost under $u_j'(t)$ is still no higher than that under $u_j^*(t)$. ■

Based on Lemma 1, we conclude that singular arcs in an agent's trajectory may occur only while it is sensing a target. Intuitively, this indicates that it may be optimal for an agent to stop moving and dwell in the vicinity of one or more targets that it can sense so as to decrease the associated uncertainty functions to an adequate level before it proceeds along the mission space. The next lemma establishes the fact that if the agent is visiting an isolated target and experiences a singular arc, then the corresponding optimal control is $u_j^*(t) = 0$. An isolated target with position x_i is defined to be one that satisfies $|x_i - x_j| > 2r$, for all $j \neq i$ where r was defined earlier as $r = \max_{j=1, \dots, N} \{r_j\}$. Accordingly, the subset $I \subseteq \{1, \dots, M\}$ of isolated targets is defined as

$$I = \{i : |x_i - x_j| > 2r \quad \forall j \neq i, r = \max_{j=1, \dots, N} \{r_j\}\}. \quad (15)$$

Lemma 2: Let $|s_j^*(t) - x_k| < r_j$ for some $j = 1, \dots, N$ and isolated target $k \in I$. If $\lambda_{s_j}^*(t) = 0$, $t \in [t_1, t_2]$, then $u_j^*(t) = 0$.

Proof: The proof is along the same line as in [10, Proposition III.3]. Assume that $\lambda_{s_j}^*(t) = 0$ over a singular arc $[t_1, t_2]$. Let $H^* \equiv H(x^*, \lambda^*, u^*)$. Since the Hamiltonian along an optimal trajectory is a constant, recalling (7) we have

$$\begin{aligned} \frac{dH^*}{dt} &= \sum_{i=1}^M \left[\dot{R}_i^*(t) + \dot{\lambda}_i^*(t) \dot{R}_i^*(t) + \lambda_i^*(t) \ddot{R}_i^*(t) \right] \\ &\quad + \sum_{j=1}^N \left[\dot{\lambda}_{s_j}^*(t) u_j^*(t) + \lambda_{s_j}^*(t) \dot{u}_j^*(t) \right] = 0 \end{aligned} \quad (16)$$

and since from (8) $\dot{\lambda}_{s_j}^*(t) = -1$, (16) reduces to

$$\frac{dH^*}{dt} = \sum_{i=1}^M \lambda_i^*(t) \ddot{R}_i^*(t) + \sum_{j=1}^N \left[\dot{\lambda}_{s_j}^*(t) u_j^*(t) + \lambda_{s_j}^*(t) \dot{u}_j^*(t) \right] = 0. \quad (17)$$

Define $S(t) = \{j | \lambda_{s_j}^*(t) = 0, \dot{\lambda}_{s_j}^*(t) = 0\}$ as the set of agents in singular arcs at t and $\bar{S}(t)$ as the set of all the remaining agents. If $j \in S(t)$, then $\dot{\lambda}_{s_j}^*(t) u_j^*(t) + \lambda_{s_j}^*(t) \dot{u}_j^*(t) = 0$. If $j \in \bar{S}(t)$, then $\lambda_{s_j}^*(t) \dot{u}_j^*(t) = 0$ since $u_j^*(t) = \pm 1$ and $\dot{u}_j^*(t) = 0$. Therefore,

we rewrite (17) as

$$\frac{dH^*}{dt} = \sum_{i=1}^M \lambda_i^*(t) \ddot{R}_i^*(t) + \sum_{j \in \bar{S}(t)} \dot{\lambda}_{s_j}^*(t) u_j^*(t) = 0. \quad (18)$$

Recalling (5), when $R_i(t) \neq 0$, we have $\dot{R}_i = A_i - B_i(1 - \prod_{n=1}^N [1 - p_{ij}(s_j(t))])$. Therefore,

$$\begin{aligned} \ddot{R}_i^*(t) &= \frac{d}{dt} \dot{R}_i^*(t) \\ &= - \sum_{j=1}^N u_j^*(t) B_i \frac{\partial p_{ij}(s_j^*(t))}{\partial s_j^*} \prod_{d \neq j} [1 - p_{id}(s_d^*(t))]. \end{aligned} \quad (19)$$

Moreover, from (9), we have

$$\dot{\lambda}_{s_j}^*(t) = \sum_{i=1, R_i \neq 0}^M \lambda_i^*(t) B_i \frac{\partial p_{ij}(s_j^*(t))}{\partial s_j^*} \prod_{d \neq j} [1 - p_{id}(s_d^*(t))]. \quad (20)$$

Combining (18)–(20), we get

$$\begin{aligned} \frac{dH^*}{dt} &= - \sum_{i=1}^M \sum_{j=1}^N u_j^*(t) \lambda_i^*(t) B_i \frac{\partial p_{ij}(s_j^*(t))}{\partial s_j^*} \prod_{d \neq j} [1 - p_{id}(s_d^*(t))] \\ &\quad + \sum_{j \in \bar{S}(t)} \sum_{i=1}^M u_j^*(t) \lambda_i^*(t) B_i \frac{\partial p_{ij}(s_j^*(t))}{\partial s_j^*} \prod_{d \neq j} [1 - p_{id}(s_d^*(t))] \\ &= - \sum_{j \in S(t)} \sum_{i=1}^M u_j^*(t) \lambda_i^*(t) B_i \frac{\partial p_{ij}(s_j^*(t))}{\partial s_j^*} \prod_{d \neq j} [1 - p_{id}(s_d^*(t))] \\ &= 0. \end{aligned} \quad (21)$$

Since we have assumed that $|s_j^*(t) - x_k| < r_j$ and k is an isolated target, it follows that $p_{kj}(s_j^*(t)) \neq 0$ and $p_{ij}(s_j(t)) = 0$ if $i \neq k$. Therefore, $\frac{\partial p_{kj}(s_j^*(t))}{\partial s_j^*} \neq 0$ and $\frac{\partial p_{ij}(s_j^*(t))}{\partial s_j^*} = 0$ for all $i \neq k$ and (21) reduces to

$$\sum_{j \in S(t)} u_j^*(t) \lambda_k^*(t) B_i \frac{\partial p_{kj}(s_j^*(t))}{\partial s_j^*} \prod_{d \neq j} [1 - p_{kd}(s_d^*(t))] = 0. \quad (22)$$

Observe that, from (8), $\lambda_i(t) > 0$ when $R_i(t) \neq 0$, $t < T$. In addition $B_i > 0$ and $\prod_{d \neq j} [1 - p_{kd}(s_d^*(t))] \neq 0$. Therefore, to satisfy (22) for all $t \in [t_1, t_2]$, we must have $u_j^*(t) = 0$, for all $j \in S(t)$. ■

We can further establish the fact that if an agent j experiences a singular arc while sensing an isolated target k , then the optimal point to stop is such that $s_j^*(t) = x_k$.

Proposition 2: Let $|s_j^*(t) - x_k| < r_j$ for some $j = 1, \dots, N$ and isolated target $k \in I$. If $\lambda_{s_j}^*(t) = 0$, $t \in [t_1, t_2]$, and $u_j^*(t_1^-) = u_j^*(t_2^+)$, then $s_j^*(t) = x_k$, $t \in [t_1, t_2]$.

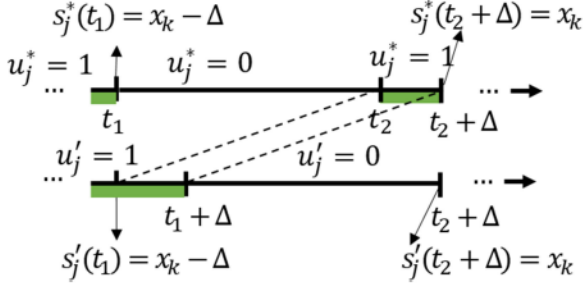


Fig. 3. Illustration of the control strategies compared in the proof of Proposition 2. The thick green bars indicate the segments when the control takes the value 1.

Proof: By Lemma 2, we know that $u_j^*(t) = 0, t \in [t_1, t_2]$. We use a contradiction argument similar to the one used in Lemma 1 to show that $s_j^*(t) = x_k, t \in [t_1, t_2]$. Suppose that $u_j^*(t_1^-) = 1$ (without the loss of generality) and that $s_j^*(t) = x_k - \Delta < x_k$. Note that, at the end of the singular arc $u_j^*(t_2^+) = 1$ since $u_j^*(t_1^-) = u_j^*(t_2^+)$. This implies that $s_j^*(t_2 + \Delta) = x_k$. Assume that $u_j^*(t), t \in [t_1, t_2 + \Delta]$ is replaced by $u_j'(t)$ as follows: $u_j'(t) = 1$ for $t \in [t_1, t_1 + \Delta]$ and $u_j'(t) = 0$ for $t \in (t_1 + \Delta, t_2 + \Delta]$. In other words, the agent moves to reach $s_j'(t_1 + \Delta) = s_j^*(t_2 + \Delta) = x_k$ and then stops. The two controls are thereafter identical, see Fig. 3. Then, referring to (6) we have $\int_{t_1}^{t_2 + \Delta} R_i'(t) dt < \int_{t_1}^{t_2 + \Delta} R_i^*(t) dt$ since $\dot{R}_i^*(t) < \dot{R}_i'(t)$ due to (5) and the fact that $p_{kj}(s_j(t))$ is monotonically decreasing in $|s_j(t) - x_k|$. Since the cost in (6) is the same over $[0, t_1]$ and $(t_2 + \Delta, T]$, it follows that $s_j^*(t) = x_k - \Delta$ cannot be optimal. The same argument holds for any $\Delta > 0$, leading to the conclusion that $s_j^*(t) = x_k, t \in [t_1, t_2]$. A similar argument also applies to the case $s_j^*(t) = x_k + \Delta > x_k$. ■

Finally, we consider the case where the state constraint (2) is included. We can then prove that this constraint is never active on an optimal trajectory, i.e., the agents reverse their directions before making contact with any other agent. Therefore, the constraint (2) is superfluous.

Proposition 3: Under the constraint $s_j(t) \leq s_{j+1}(t)$, on an optimal trajectory, $s_j(t) \neq s_{j+1}(t)$ for all $t \in (0, T), j = 1, \dots, N - 1$.

Proof: The proof is almost identical to that of Proposition III.4 in [10] and is, therefore, omitted. ■

The above analysis, including Propositions 1–3, fully characterize the structure of the optimal control as consisting of intervals in $[0, T]$ where $u_j^*(t) \in \{-1, 0, 1\}$ depending entirely on the sign of $\lambda_{s_j}(t)$. Based on this analysis, we can parameterize P1 so that the cost in (6) depends on a set of First, switching points (see Definition III.2) where an agent switches its control from $u_j(t) = \pm 1$ to ∓ 1 or possibly 0, and second, dwelling times if an agent switches from $u_j(t) = \pm 1$ to 0. In other words, the optimal trajectory of each agent j is totally characterized by two parameter vectors: Switching points $\theta_j = [\theta_{j1}, \theta_{j2} \dots \theta_{j\Gamma}]$ and dwelling times $\omega_j = [\omega_{j1}, \omega_{j2} \dots \omega_{j\Gamma'}]$ where Γ and Γ' are prior parameters depending on the given time horizon. This defines a hybrid system with state dynamics (1), (5). The dynamics remain unchanged in between events that cause them to change, i.e., the points $\theta_{j1}, \dots, \theta_{j\Gamma}$ above and instants when

$R_i(t)$ switches from > 0 to 0 or vice versa. Therefore, the overall cost function (6) can be parametrically expressed as $J(\theta, \omega)$ and rewritten as the sum of costs over corresponding interevent intervals over a given time horizon

$$J(\theta, \omega) = \frac{1}{T} \sum_{k=0}^K \int_{\tau_k(\theta, \omega)}^{\tau_{k+1}(\theta, \omega)} \sum_{i=1}^M R_i(t) dt \quad (23)$$

where τ_k in (23) is the k th event time. This will allow us to apply IPA to determine a gradient $\nabla J(\theta, \omega)$ with respect to these parameters and apply any standard gradient-based optimization algorithm to obtain a (locally) optimal solution.

IV. INFINITESIMAL PERTURBATION ANALYSIS

As concluded in the previous section, the optimal agent trajectories may be selected from the family $\{s(\theta, \omega, t, s_0)\}$ with parameter vectors θ and ω and a given initial condition s_0 . Along these trajectories, the agents are subject to dynamics (1) and the targets are subject to (5). An event (e.g., an agent stopping at some target x_i) occurring at time $\tau_k(\theta, \omega)$ triggers a switch in these state dynamics. IPA specifies how changes in θ and ω influence the state $s(\theta, \omega, t, s_0)$, as well as event times $\tau_k(\theta, \omega), k = 1, 2, \dots$, and, ultimately the cost function (23). We briefly review next the IPA framework for general stochastic hybrid systems as presented in [19].

Let $\{\tau_k(\theta)\}, k = 1, \dots, K$, denote the occurrence times of all events in the state trajectory of a hybrid system with dynamics $\dot{x} = f_k(x, \theta, t)$ over an interval $[\tau_k(\theta), \tau_{k+1}(\theta))$, where $\theta \in \Theta$ is some parameter vector and Θ is a given compact, convex set. For convenience, we set $\tau_0 = 0$ and $\tau_{K+1} = T$. We use the Jacobian matrix notation: $x'(t) \equiv \frac{\partial x(\theta, t)}{\partial \theta}$ and $\tau'_k \equiv \frac{\partial \tau_k(\theta)}{\partial \theta}$, for all the state and event time derivatives. It is shown in [19] that

$$\frac{d}{dt} x'(t) = \frac{\partial f_k(t)}{\partial x} x'(t) + \frac{\partial f_k(t)}{\partial \theta} \quad (24)$$

for $t \in [\tau_k, \tau_{k+1})$ with boundary condition

$$x'(\tau_k^+) = x'(\tau_k^-) + [f_{k-1}(\tau_k^-) - f_k(\tau_k^+)] \tau'_k \quad (25)$$

for $k = 0, \dots, K$. In order to complete the evaluation of $x'(\tau_k^+)$ in (25), we need to determine τ'_k . We classify events into two categories. An event is exogenous if it causes a discrete state transition at time τ_k independent of the controllable vector θ and, therefore, satisfies $\tau'_k = 0$. Otherwise, the event is endogenous and there exists a continuously differentiable function $g_k : \mathbb{R}^n \times \Theta \rightarrow \mathbb{R}$ such that $\tau_k = \min\{t > \tau_{k-1} : g_k(x(\theta, t), \theta) = 0\}$ and

$$\tau'_k = - \left[\frac{\partial g_k}{\partial x} f_k(\tau_k^-) \right]^{-1} \left(\frac{\partial g_k}{\partial \theta} + \frac{\partial g_k}{\partial x} x'(\tau_k^-) \right) \quad (26)$$

as long as $\frac{\partial g_k}{\partial x} f_k(\tau_k^-) \neq 0$ (details may be found in [19]).

Denote the time-varying cost along a given trajectory as $L(x, \theta, t)$, so the cost in the k th interevent interval is $J_k(x, \theta) = \int_{\tau_k}^{\tau_{k+1}} L(x, \theta, t) dt$ and the total cost is $J(x, \theta) = \sum_{k=0}^K J_k(x, \theta)$. Differentiating and applying the Leibniz rule with the observation that all terms of the form $L(x(\tau_k), \theta, \tau_k) \tau'_k$

are mutually canceled with $\tau_0 = 0, \tau_{K+1} = T$ fixed, we obtain

$$\begin{aligned} \frac{\partial J(x, \theta)}{\partial \theta} &= \sum_{k=0}^K \frac{\partial}{\partial \theta} \int_{\tau_k}^{\tau_{k+1}} L(x, \theta, t) dt \\ &= \sum_{k=0}^K \int_{\tau_k}^{\tau_{k+1}} \frac{\partial L(x, \theta, t)}{\partial x} x'(t) + \frac{\partial L(x, \theta, t)}{\partial \theta} dt. \end{aligned} \quad (27)$$

In our setting, we have $L(x, \theta, t) = \sum_{i=1}^M R_i(t)$ from (23), which is not an explicit function of the state $x(t) = [R_1(t), \dots, R_M(t), s_1(t), \dots, s_N(t)]$. Thus, the gradient $\nabla J(\theta, \omega) = [\frac{\partial J(\theta, \omega)}{\partial \theta}, \frac{\partial J(\theta, \omega)}{\partial \omega}]^T$ reduces to

$$\nabla J(\theta, \omega) = \frac{1}{T} \sum_{k=0}^K \sum_{i=1}^M \int_{\tau_k(\theta, \omega)}^{\tau_{k+1}(\theta, \omega)} \nabla R_i(t) dt \quad (28)$$

where $\nabla R_i(t) = [\frac{\partial R_i(t)}{\partial \theta}, \frac{\partial R_i(t)}{\partial \omega}]^T$.

Applying (24), (25), (26), we can evaluate $\nabla R_i(t)$. In contrast to [10], in our problem agents are allowed to dwell on every target and it is necessary to optimize these dwelling times. Therefore, we need to consider all the possible forms of control sequences: $\pm 1 \rightarrow 0, 0 \rightarrow \pm 1$, and $\pm 1 \rightarrow \mp 1$. Applying (24) on (5) and integrating over time from the last event, we obtain

$$\begin{aligned} \frac{\partial R_i(t)}{\partial \theta_j} &= \frac{\partial R_i(\tau_k^+)}{\partial \theta_j} \\ &- \begin{cases} 0 & \text{if } R_i(t) = 0 \text{ and } A_i < B_i P_i(s(t)) \\ B_i \frac{\partial p_{ij}(s_j)}{\partial s_j} \frac{\partial s_j(\tau_k^+)}{\partial \theta_j} G_{ij}(t) & \text{otherwise} \end{cases} \end{aligned} \quad (29)$$

$$\begin{aligned} \frac{\partial R_i(t)}{\partial \omega_j} &= \frac{\partial R_i(\tau_k^+)}{\partial \omega_j} \\ &- \begin{cases} 0 & \text{if } R_i(t) = 0 \text{ and } A_i < B_i P_i(s(t)) \\ B_i \frac{\partial p_{ij}(s_j)}{\partial s_j} \frac{\partial s_j(\tau_k^+)}{\partial \omega_j} G_{ij}(t) & \text{otherwise} \end{cases} \end{aligned} \quad (30)$$

where θ_j is the vector of switching points of agent j and ω_j the associated dwell times, $\frac{\partial p_{ij}(s_j)}{\partial s_j} = \pm \frac{1}{r_j}$ or 0 depending on the relative position of target i with respect to the position of agent j . Moreover, the term G_{ij} is defined as

$$G_{ij}(t) = \int_{\tau_k}^t \prod_{d \neq j} [1 - p_{id}(s_d(\tau))] d\tau \quad (31)$$

and may be interpreted as a ‘‘collaboration factor’’ capturing the contributions of all other agents $d \neq j$ in monitoring target i .

In between each two consecutive events, $\nabla R_i(t)$ evolves according to (29) and (30), but at the event times it may experience discontinuities as captured by the boundary condition (25) with τ_k' evaluated through (26).

First, let us consider the events that cause switches in $\dot{R}_i(t)$ in (5) at time τ_k . For these events, the dynamics of $s_j(t)$ are

continuous so that $\nabla s_j(\tau_k^-) = \nabla s_j(\tau_k^+)$. For target i

$$\nabla R_i(\tau_k^+) = \begin{cases} \nabla R_i(\tau_k^-) & \text{if } R_i(\tau_k) \neq 0 \\ 0 & \text{if } R_i(\tau_k) = 0 \end{cases}. \quad (32)$$

Notice that, $\nabla R_i(t)$ is reset to zero when $R_i(t)$ reaches zero at event time τ_k regardless of the value $\nabla R_i(\tau_k^-)$, otherwise $\nabla R_i(t)$ evolves continuously in t .

Second, let us consider events that cause switches in $\dot{s}_j(t)$ in (1) at time τ_k . For these events, the dynamics of $R_i(t)$ are continuous so that $\nabla R_i(\tau_k^-) = \nabla R_i(\tau_k^+)$. In order to evaluate (29) and (30), we need $\frac{\partial s_j(\tau_k^+)}{\partial \theta_j}$ and $\frac{\partial s_j(\tau_k^+)}{\partial \omega_j}$. Clearly, these cannot be affected by future events and we only have to consider the prior and current control switches from $l = 1, 2, \dots, \xi$. Let $\theta_{j\xi}$ and $\omega_{j\xi}$ be the current switching point and dwelling time. Again, applying (24), (25), (26) to (1), we have

Case 1: $u_j(\tau_k^-) = \pm 1, u_j(\tau_k^+) = 0$.

$$\frac{\partial s_j}{\partial \theta_{jl}}(\tau_k^+) = \begin{cases} 1 & \text{if } l = \xi \\ 0 & \text{if } l < \xi \end{cases}, \quad (33)$$

$$\frac{\partial s_j}{\partial \omega_{jl}}(\tau_k^+) = 0 \quad \text{for all } l \leq \xi. \quad (34)$$

Case 2: $u_j(\tau_k^-) = 0, u_j(\tau_k^+) = \pm 1$.

$$\begin{aligned} \frac{\partial s_j}{\partial \theta_{jl}}(\tau_k^+) &= \begin{cases} \frac{\partial s_j}{\partial \theta_{jl}}(\tau_k^-) - u_j(\tau_k^+) \text{sgn}(\theta_{j\xi} - \theta_{j(\xi-1)}) & \text{if } l = \xi \\ \frac{\partial s_j}{\partial \theta_{jl}}(\tau_k^-) - u_j(\tau_k^+) [\text{sgn}(\theta_{jl} - \theta_{j(l-1)}) - \text{sgn}(\theta_{j(l+1)} - \theta_{jl})] & \text{if } l < \xi \end{cases}, \end{aligned} \quad (35)$$

$$\frac{\partial s_j}{\partial \omega_{jl}}(\tau_k^+) = -u_j(\tau_k^+) \quad \text{for all } l \leq \xi. \quad (36)$$

Case 3: $u_j(\tau_k^-) = \pm 1, u_j(\tau_k^+) = \mp 1$.

$$\frac{\partial s_j}{\partial \theta_{jl}}(\tau_k^+) = \begin{cases} 2 & \text{if } l = \xi \\ -\frac{\partial s_j}{\partial \theta_{jl}}(\tau_k^-) & \text{if } l < \xi \end{cases}. \quad (37)$$

Details of these derivations can be found in the Appendix of [10]. An important difference arises in Case 2 above, where $\tau_k = |\theta_{j1} - a| + \omega_{j1} + \dots + |\theta_{j\xi} - \theta_{j(\xi-1)}| + \omega_{j\xi}$. We eliminate the constraints on the switching location that $\theta_{j\xi} \leq \theta_{j(\xi-1)}$ if ξ is even and $\theta_{j\xi} \geq \theta_{j(\xi-1)}$ if ξ is odd.

In addition, we show in a forthcoming paper that even though it appears that the IPA gradient in (29) and (30) depends on the state of other agents, it turns out that only the collaboration term (31) affects changes in the gradient. This suggests a decentralized algorithm as shown in [27] through which each agent can locally evaluate its gradient using only occasional interagent information exchange and still achieve the same solution as the centralized one obtained in this paper.

The event excitation problem: Note that, all the derivative updates above are limited to events occurring at times $\tau_k(\theta, \omega)$, $k = 1, 2, \dots$. Thus, this approach is scalable in the number of events characterizing the hybrid system, not its state space.

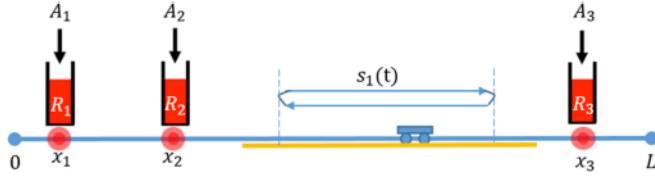


Fig. 4. Example of no event excitation leading to a failure of IPA finding an optimal agent trajectory. The yellow bar is the segment of the space initially covered by the agent.

While this is a distinct advantage, it also involves a potential drawback. In particular, it assumes that the events involved in IPA updates are observable along a state trajectory. However, if the current trajectory never reaches the vicinity of any target so as to be able to sense it and affect the overall uncertainty cost function, then any small perturbation to the trajectory will have no effect on the cost. As a result, the IPA will fail as illustrated in Fig. 4: where, the single agent trajectory $s_1(\theta, \omega, t)$ is limited to include no event. Thus, if a gradient-based procedure is initialized with such $s_1(\theta, \omega, t)$, no event involved in the evaluation of $\nabla R_i(t)$ is “excited” and the cost gradient remains zero.

In order to overcome this problem, we modify our cost metric by introducing a function $V(\cdot)$ with the property of “spreading” the value of some $R_i(t)$ over all points $w \in \Omega \equiv [0, L]$ as in (38). Recalling Proposition 1, we limit ourselves to the subset $\mathcal{B} = [x_1, x_M] \subset \Omega$. Then, for all points $w \in \mathcal{B}$, we define $V(w, t)$ as a continuous density function which results in a total value equivalent to the weighted sum of the target values $\sum_{i=1}^M R_i(t)$ (the existence of such a function is formally proved in [24]). We impose the condition that $V(w, t)$ be monotonically decreasing in the Euclidean distance $\|w - x_i\|$. More precisely, we define $d_i^+(w) = \max(\|w - x_i\|, r)$ where $r = \min_{j=1, \dots, N} \{r_j\}$ which ensures that $d_i^+(w) \geq r$. Thus, $d_i^+(w) = r > 0$ is fixed for all points within the target’s vicinity, $w \in [x_i - r, x_i + r]$. We define

$$V(w, t) = \sum_{i=1}^M \frac{\alpha_i R_i(t)}{d_i^+(w)}. \quad (38)$$

Note that, $V(w, t)$ corresponds to the “total weighted reward density” at $w \in \mathcal{B}$. The weight α_i may be included to capture the relative importance of targets, but we shall henceforth set $\alpha_i = 1$ for all $i = 1, \dots, M$ for simplicity. In order to differentiate points $w \in \mathcal{B}$ in terms of their location relative to the agents states $s_j(t)$, $j = 1, \dots, N$, we also define the travel cost function

$$Q(w, s(t)) = \sum_{j=1}^N \|s_j(t) - w\|. \quad (39)$$

Using these definitions we introduce a new objective function component, which is added to the objective function in (6)

$$J_2(t) = \int_{\mathcal{B}} Q(w, s(t)) V(w, t) dw. \quad (40)$$

The significance of $J_2(t)$ is that it accounts for the movement of agents through $Q(w, s(t))$ and captures the target state values through $V(w, t)$. Introducing this term in the objective function

in the following creates a nonzero gradient even if the agent trajectories are not passing through any targets. Defining the metric in (23) as $J_1(t)$ and combining it with $J_2(t)$, we get

$$\min_{\theta \in \Theta, \omega \geq 0} J(\theta, \omega, T) = \frac{1}{T} \int_0^T [J_1(\theta, \omega, t) + e^{-\beta t} J_2(\theta, \omega, t)] dt \quad (41)$$

where, as a reminder, $J_1(\theta, \omega, t) = \sum_{i=1}^M R_i(t)$ is the original uncertainty metric. This creates a continuous potential field for the agents which ensures a nonzero cost gradient even when the trajectories do not excite any events. This nonzero gradient will induce the trajectory adjustments that naturally bring them toward ones with observable events. The factor $e^{-\beta t}$ with $\beta > 0$ is included so that as the number of IPA iterations increases, the effect of $J_2(\theta, \omega, t)$ is diminished and the original objective is ultimately recovered. The IPA derivative of $J_2(\theta, \omega, t)$ is

$$\begin{aligned} \frac{\partial J_2(\theta, \omega, t)}{\partial \theta} &= \int_{\mathcal{B}} \left[\frac{\partial Q(w, \theta, \omega, s(t), t)}{\partial \theta} V(w, \theta, \omega, t) \right. \\ &\quad \left. + Q(w, \theta, \omega, s(t), t) \frac{\partial V(w, \theta, \omega, t)}{\partial \theta} \right] dw \end{aligned} \quad (42)$$

where the IPA derivatives of $Q(w, \theta, \omega, s(t), t)$ and $V(w, \theta, \omega, t)$ are obtained following the same procedure described previously. Before making this modification, the lack of event excitation in a state trajectory results in the total derivative (28) being zero. On the other hand, in (42) we observe that if no events occur, the second part in the integral, which involves $\frac{\partial V(\cdot)}{\partial \theta}$ is zero, since $\sum_{i=1}^M \frac{\partial R_i(t)}{\partial \theta} = 0$ all the time. However, the first part in the integral does not depend on events, but only the sensitivity of $Q(w, \theta, \omega, s(t), t)$ in (39) with respect to the parameters θ, ω . As a result, the agent trajectories are adjusted so as to eventually excite desired events and any gradient-based procedure we use in conjunction with IPA is no longer limited by the absence of event excitation.

IPA robustness to uncertainty modeling: Observe that the evaluation of $\nabla R_i(t)$, hence $\nabla J(\theta, \omega)$, is independent of A_i , $i = 1, \dots, M$, i.e., the parameters in our uncertainty model. In fact, the dependence of $\nabla R_i(t)$ on A_i , $i = 1, \dots, M$, manifests itself through the event times τ_k when $R_i(\tau_k)$ reaches zero, but they, unlike A_i which may be unknown, are directly observable during the gradient evaluation process. Thus, the IPA approach possesses an inherent robustness property: There is no need to explicitly model how uncertainty affects $R_i(t)$ in (5). Consequently, we may treat A_i as unknown without affecting the solution approach (the values of $\nabla R_i(t)$ are obviously affected). We may also allow this uncertainty to be modeled through random processes $\{A_i(t)\}$, $i = 1, \dots, M$. Under mild technical conditions on the statistical characteristics of $\{A_i(t)\}$ [19], the resulting $\nabla J(\theta, \omega)$ is an unbiased estimate of a stochastic gradient.

IPA gradient descent algorithm: We apply a standard gradient descent scheme to optimize parameter $[\theta, \omega]^T$ following

$$[\theta^{l+1}, \omega^{l+1}]^T = [\theta^l, \omega^l]^T - \begin{bmatrix} \alpha_\theta^l & 0 \\ 0 & \alpha_\omega^l \end{bmatrix} \nabla J(\theta, \omega) \quad (43)$$

Algorithm 1: IPA-Driven Gradient Descent Optimization.

-
- 1: Initialize parameters θ, ω
 - 2: Select an error tolerance $\epsilon > 0$
 - 3: **repeat**:
 - 4: Compute trajectory $s_j(t), t \in [0, T], \forall j = 1 \dots N$ using θ, ω .
 - 5: Compute the IPA gradient $\nabla J(\theta, \omega)$
 - 6: Update θ, ω using (43)
 - 7: **until** $\|\nabla J(\theta, \omega)\| < \epsilon$
 - 8: Set the optimized parameter $\theta^* = \theta, \omega^* = \omega$ and compute $J(\theta^*, \omega^*)$
-

where α_θ^l and α_ω^l are diminishing step-size sequences satisfying $\sum_{l=1}^{\infty} \alpha_\theta^l = \infty, \lim_{l \rightarrow \infty} \alpha_\theta^l = 0$ and $\sum_{l=1}^{\infty} \alpha_\omega^l = \infty, \lim_{l \rightarrow \infty} \alpha_\omega^l = 0$ (elementwise). Our gradient-based IPA optimization is summarized in Algorithm 1.

We briefly mention some technical issues concerning Algorithm 1. First, the dimensions Γ_j and Γ'_j of θ_j and ω_j (i.e., the number of switching points contained in an optimal trajectory) are a priori unknown and depend on T . However, a feasible upper bound for each can be easily derived as shown in [10] and we use the maximum upper bound over all agents $j = 1, \dots, N$ to initialize θ and ω . Second, K in (23) corresponds to the index of the last event observed within the given time horizon T , this is simply a counter which does not affect the algorithm implementation. Third, the convergence of Algorithm 1 is guaranteed under standard assumptions made on the step size sequences in (43) [28]. Finally, to ensure that the execution time of Algorithm 1 does not exceed a given desired upper bound (depending on the computation device used), we may select a maximum number of iterations n_0 to ensure that the algorithm terminates before exceeding this bound.

V. GRAPH-BASED SCHEDULING METHOD

While the IPA-driven gradient-based approach described in Section IV offers several compelling advantages, it is not guaranteed to find a global optimum. It is important, then, to understand the level of suboptimality that can occur. In this section, we develop a graph-based scheduling method which, at the cost of the expected but significant increase in computational complexity, guarantees a global optimal solution. While its complexity limits its applicability to problems of small size, it does allow us to compare the performance of the IPA-based scheme in Algorithm 1 to the global optimal in that setting. The complexity of graph-based approaches such as the one developed here are driven by the size of the graph and are invariant to the underlying dimensionality of the mission space. Such approaches may then have a greater role in mission spaces of dimension greater than one where it has been shown that it is challenging to identify a parametric representation of the optimal agent trajectories [11].

As illustrated in Fig. 5, our approach to the discrete setting is to divide the overall planning time horizon T for agent j into a sum of K_j consecutive time steps $\{t_j^1, t_j^2, \dots, t_j^{K_j}\}$, $j = 1, \dots, N$, with $t_j^1 = 0$. The dependence on j indicates that each



Fig. 5. Time sequence of a single agent on a given trajectory. The t_i are the time points where the agent begins to move to the next target in the sequence. Each move takes Δt_i units of time followed by a dwell period of Δd_i units of time during which information is collected from the target.

agent may have a different discretization. We denote the end of the K th step as $t_j^{K+1} = T$. Each step $k \in \{1, \dots, K_j\}$ begins with a travel stage where the agent moves to a particular target i . Under the assumption that during the transition between targets each agent moves at its maximum speed of $|u_j| = 1$, the travel time is

$$\Delta t_j^k = |s_j^k(t_j^k) - x_i|. \quad (44)$$

Upon arriving at a target, the agent dwells for a time Δd_j^k . Note that, due to the range-based nature of the sensing, the uncertainty actually begins to decrease before the arrival of the agent at the target and continues to decrease after the agent has departed until the target is out of the sensing range.

The problem of optimizing u_j to minimize the average uncertainty over all the targets has been translated into a mixed integer programming (MIP) problem to select the sequence of targets and the dwell time at each target. Letting a_{ji}^k be a binary variable denoting whether agent j is assigned to target i at time step k , this MIP is

$$\min_{a_{ji}^k, \Delta d_j^k} J = \frac{1}{T} \sum_{i=1}^M \int_0^T R_i(t) dt \quad (45)$$

$$\text{s.t. } a_{ji}^k \in \{1, 0\}, \quad \sum_{i=1}^M a_{ji}^k = 1, \quad \forall j, k \quad (46)$$

$$\sum_{k=1}^{K_j} \Delta t_j^k + \Delta d_j^k \leq T, \quad \forall j. \quad (47)$$

Note that, we assume that each agent is assigned to a maximum of only one target at any one time. The IPA-driven approach has no such restriction. We break the solution of this problem into three parts: the enumeration of all feasible trajectories, the calculation of the cost of the feasible trajectories, and then selection of the optimal trajectory based on those costs. We focus on the case of a single agent for simplicity of description before generalizing to the multiple agent case.

The first part, namely determining feasible trajectories, is straightforward. Given the fixed time horizon T , the target locations, the locations of the agent at the start of the time horizon, and the maximum speed of the agent, a feasible trajectory is one where the sequence of targets can all be visited within the time horizon. Similarly, the third part simply involves comparing the trajectories and selecting the one with the minimal cost.

In the second part, the cost of each feasible trajectory must be determined. Suppose we have a given feasible trajectory with K targets in its sequence. Note that, because a trajectory may

include multiple visits to the same target, K may be larger than m (and may be much larger for large time horizons and small m). Let $\{i_1, i_2, \dots, i_K\}$ denote the indices of the targets in the sequence. From (45), the cost of this trajectory is given by the optimization problem

$$\begin{aligned} \min_{\Delta d_j^k} J &= \frac{1}{T} \sum_{i=1}^M \int_0^T R_i(t) dt \\ \text{s.t.} \quad &\sum_{k=1}^K \Delta t^k + \Delta d^k \leq T. \end{aligned}$$

Our approach to solving this optimization problem is to setup a recursive calculation. As illustrated in Fig. 5, since the travel times Δt_i are completely determined by the sequence alone, optimizing over the dwell times is equivalent to optimizing the switching times t_i . Assume for the moment that the switching times through t_{K-1} have been determined (and thus the first $K-2$ dwell times, $\Delta d^1, \dots, \Delta d^{K-2}$ are known). The two final dwell times are completely determined by selecting time t_K at which to switch the agent from target i_{K-1} to target i_K . This then gives us a simple single variable optimization problem

$$\min_{\Delta T_K} J = \frac{1}{\Delta T} \int_{t_{K-1}}^T (R_{i_{K-1}}(t) + R_{i_K}(t)) dt$$

where $\Delta T = T - t_{K-1}$. This allows the final switching time to be expressed as a function of the previous time $t_K = t_K(t_{K-1})$. Repeating this leads to an expression of the optimal switching times as a nested sequence of optimization functions which can be solved numerically.

This same optimization procedure can be generalized to the case of multiple agents. The primary challenge is that the set of feasible trajectories, and the calculation of the cost of those trajectories, quickly becomes intractable since all the possible combinations of assignments of multiple agents must be considered. The computational complexity can be mitigated somewhat by taking advantage of the known properties of optimal solutions (as described in Section III).

Since the computational complexity is exponential in the length of the time horizon, this approach is limited to short horizons. In prior work on linear systems, it was shown that an appropriately defined periodic schedule is sufficient to ensure the entire system remains controllable [29], [30]. In the current context, this translates to being able to keep the uncertainty of each of the targets arbitrarily close to zero. Our most recent work [14] shows that, under a given periodic visiting sequence, it is generally optimal to stay with a target until its uncertainty reaches zero and then to switch to another. Motivated by this, we typically apply our discrete approach over a relatively short time horizon and extend the resulting optimal trajectory to longer horizons by simply repeating it in a periodic manner.

VI. SIMULATION EXAMPLES

To demonstrate the performance of the gradient-based algorithm using the IPA scheme described in Section IV, we present two sets of numerical examples. The first set uses determin-

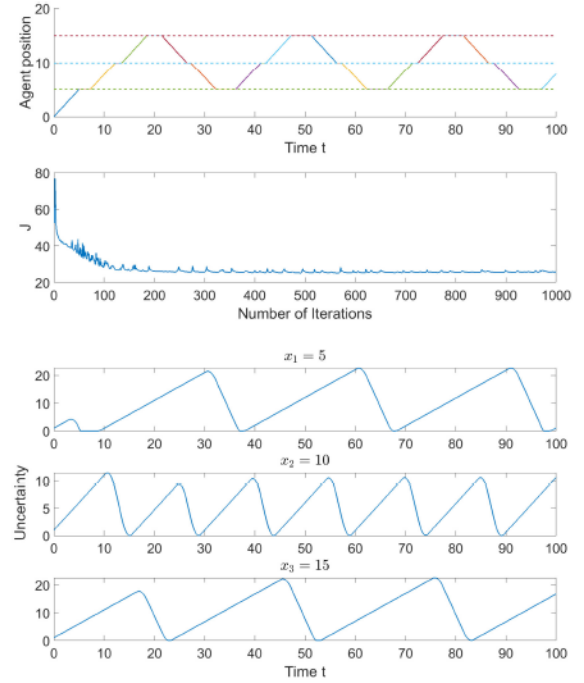


Fig. 6. Top: Trajectory of a single agent monitoring three targets using the IPA-driven gradient descent algorithm. Second: Calculated cost as a function of iteration in the gradient descent. Bottom figures: Target uncertainties along the trajectory. The final cost is 25.54.

istic target locations and dynamics. The results are compared against the optimal found by the discrete scheduling algorithm of Section V. The second set demonstrates the robustness of the IPA scheme with respect to a stochastic uncertainty model.

The first example consists of a single agent performing a persistent monitoring task on three targets over a time horizon of 100 s. The targets are located at positions $x_1 = 5$, $x_2 = 10$, $x_3 = 15$ and their uncertainty dynamics in (5) are defined by the parameters $A_i = 1$, $B_i = 5$, and $R_i(0) = 1$ for all i . The agent has a sensing range of 2 and is initialized with $s(0) = 0$, $u(0) = 1$. The results from the IPA gradient descent approach are shown in Fig. 6. The top image shows the optimal trajectory of the agent determined after 1000 iterations of Algorithm 1 while the bottom shows the evolution of the overall cost as a function of iteration number. The agent is moving through a periodic cycle of $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_2 \rightarrow x_1 \dots$, dwelling for a short time at each target before moving to the next. Notice that, the agent dwells for a shorter time at the center target since it visits that location twice per cycle. The second image in the figure shows that the gradient descent converges within the first 100 iterations. This example aims to test the event-driven IPA scheme with the discrete scheduling algorithm which yields a global optimal but suffers from computational intensity. Thus, we start with a short time horizon $T = 100$ s. Event-driven IPA in conjunction with Algorithm 1 optimizes the trajectory fast but the convergence exhibits the oscillatory behavior due to lack of an adequate number of observed events within a short time horizon. The final cost is 25.54. The bottom images in Fig. 6 show the evolution of the target uncertainties.

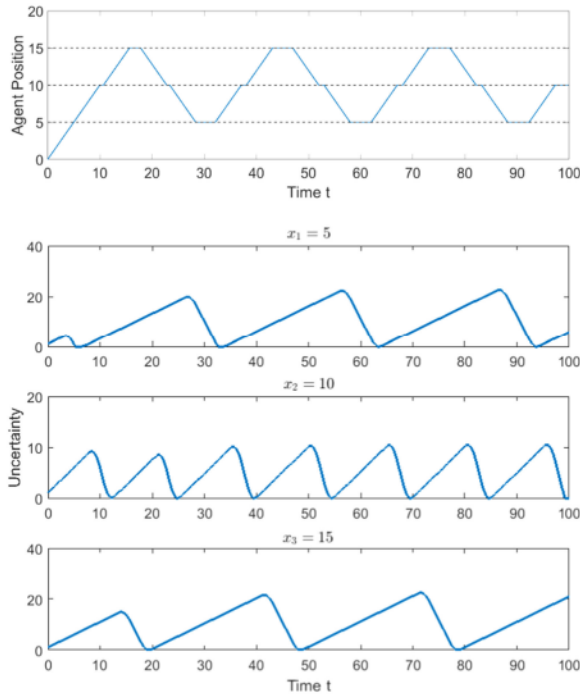


Fig. 7. Top: Trajectory of a single agent monitoring three targets using the optimal discrete assignment and dwelling time. Bottom: Target uncertainties along the trajectory. The final cost is 25.07.

The corresponding result based on the discrete setting of Section V is essentially the same with the agent moving through the three targets in a periodic fashion as shown in Fig. 7. The only deviation from the IPA scheme occurs at the end of the horizon where the discrete approach returns to the center target. The final cost was 25.07, thus verifying the approximate optimality of the solution found in Fig. 6.

The next example involves two agents and five targets over a time horizon of 500 s. The targets are located at $x_1 = 5$, $x_2 = 7$, $x_3 = 9$, $x_4 = 13$, $x_5 = 15$. The uncertainty dynamics were the same as in the single-agent, three-target case. As before, the agents have a sensing range of 2 and are initialized at $s_1(0) = s_2(0) = 0$, with $u_1(0) = u_2(0) = 1$. The results from the event-driven IPA gradient descent approach are shown in Fig. 8. The solution is again periodic with the agents dividing the targets into two groups. Notice that, the single agent on targets x_4 and x_5 is able to keep the uncertainties very close to zero since the targets are quite close relative to the sensing range of the agent. The other agent is able to hold its middle target (x_2) close to zero since it is visited more often. The corresponding result based on the discrete setting is shown in Fig. 9. Rather than solving over the full horizon, the problem was solved over a 60 s horizon and then, the periodic trajectory repeated to fill the 500 s horizon. The results are again very close to the event-driven IPA method.

Note that, the optimal trajectories in both one- and two-agent examples are bounded between $[5, 15]$ (positions of the first and last target), which is consistent with Proposition 1.

As mentioned earlier, the IPA robustness property allows us to handle stochastic uncertainty models at targets. We show a

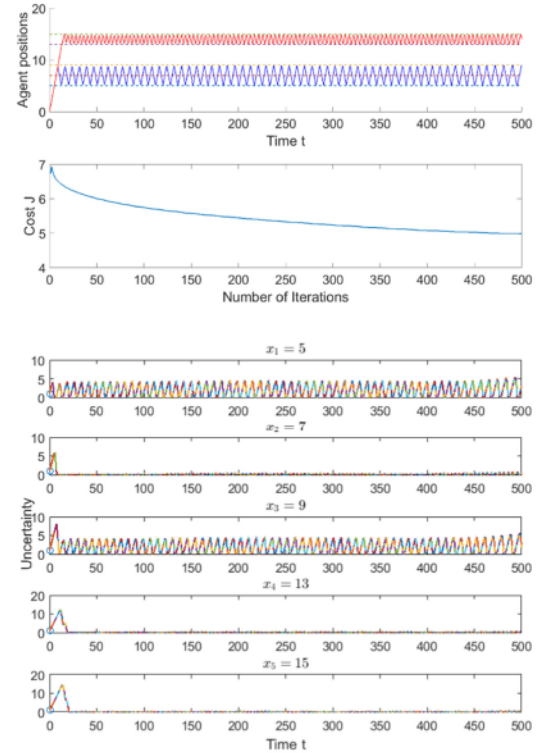


Fig. 8. Top: Trajectories of two agents monitoring five targets using the IPA gradient descent algorithm. Second: Calculated cost as a function of iteration. Bottom figures: Target uncertainty values along the above trajectories. The final cost is 4.99.

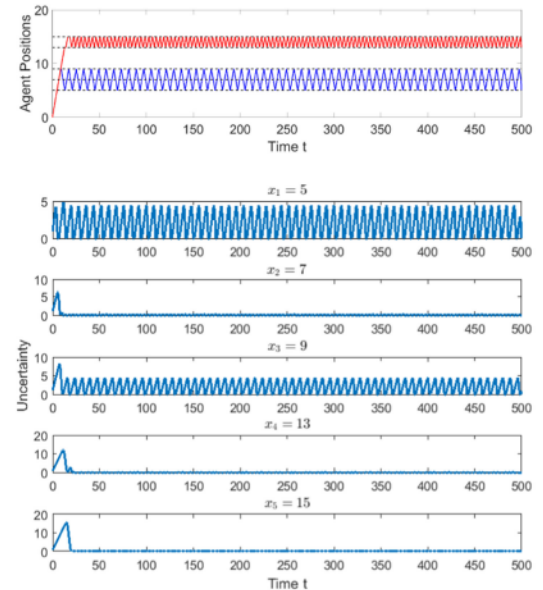


Fig. 9. Top: Trajectories of two agents monitoring five targets using the discrete assignment and dwelling time. Bottom: Target uncertainty values along the above trajectories. The final cost was 4.92.

one-agent example in Fig. 10(b) where the uncertainty inflow rate $A_i(t)$ is uniformly distributed between $(0.75, 1.25)$ for all targets. In Fig. 10(c), we introduce randomness by allowing target positions to vary uniformly over $(x_i - 0.25, x_i + 0.25)$. In both cases, the optimal cost in the stochastic models in Fig. 10(b)

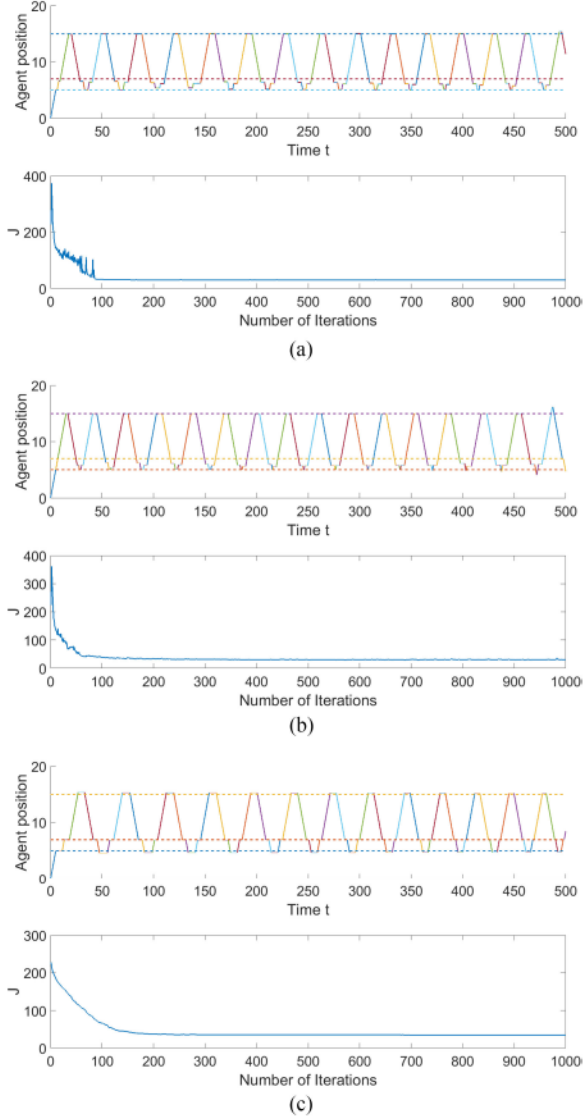


Fig. 10. Examples demonstrating IPA robustness with respect to stochastic uncertainty. (a)–(c) Top plot: Optimal trajectory $s^*(t)$. Bottom plot: Cost convergence. (a) Example of deterministic target model. Target positions 5; 7; 15, dynamics parameter $A_i = 1$, $B = 5$, $r = 2$. $J^*(\theta, \omega) = 29.40$. (b) Example with stochastic uncertainty inflow processes. $A_i \sim U(0.75, 1.25)$. $J^*(\theta, \omega) = 30.27$. (c) Example with stochastic target locations $\sim U(x_i - 0.25, x_i + 0.25)$. $J^*(\theta, \omega) = 34.89$.

and (c) are close to the optimal cost of the deterministic case in Fig. 10(a) where the parameter A_i and target positions x_i are the means of the associated random processes in the stochastic models. The convergence depends on the variance of these random processes. As variance increases, so does the cost, as expected.

The event excitation issue is addressed in Fig. 11(a), where the agent trajectory is initialized so that it is not close to any of the targets. Using the original problem formulation (without the inclusion of $J_2(\theta, \omega, t)$ in (41)), the initial trajectory and cost remain unchanged. After adding $J_2(\theta, \omega, t)$, the blue, green, and red curves in Fig. 11(c) show the trajectory adjustment after 5, 10, and 15 iterations, respectively. After 100 iterations, the cost converges to 30.24 as shown in Fig. 11(b) which is close to

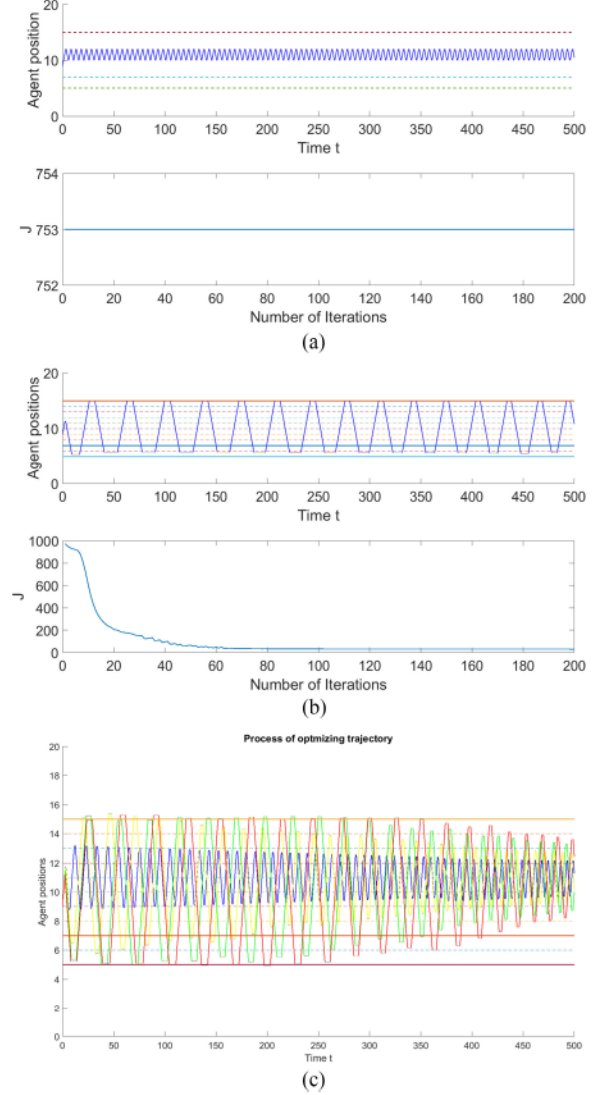


Fig. 11. The event excitation issue. After adding $J_2(\theta, \omega, t)$, the trajectory adjusts to include targets, the cost converges to 30.24 which is close to the optimal cost in Fig. 10(a) where the target dynamics are the same. (a) A trajectory where IPA fails due to lack of event excitation. Top plot: agent trajectory. Bottom plot: cost convergence. (b) IPA optimization after event excitation. Top plot: optimal agent trajectory. Bottom plot: cost convergence. $J^*(\theta, \omega) = 30.24$. (c) Trajectory adjustments with event excitation after 5 (blue), 10 (green), and 15 (red) iterations.

the optimal cost in Fig. 10(a) where the target dynamics are the same.

VII. CONCLUSION

We have formulated a persistent monitoring problem with the objective of controlling the movement of multiple cooperating agents so as to minimize an uncertainty metric associated with a finite number of targets. We have established properties of the optimal control solution which reduce the problem to a parametric optimization one. A complete online solution is given by IPA to evaluate the gradient of the objective function with respect to all parameters. We also address the case when IPA gradient estimation fails because of the lack of event excitation. We solve this problem by introducing a new metric for the

objective function which creates a potential field guaranteeing that gradient values are nonzero. This approach is compared to an alternative graph-based task scheduling algorithm for determining an optimal sequence of target visits. Ongoing research includes the study of optimal switching conditions for target visits and the periodic behavior in the steady-state following a graph-theoretic approach. Further, we are in the process of developing a decentralized version of the IPA-driven optimization in which each agent evaluates its own local gradient using only occasional interagent communication [27]. Finally, we are considering extensions to higher-dimensional mission spaces with certain constraints such as 2-D grids consisting of intersecting linear segments (e.g., urban street settings).

REFERENCES

- [1] M. Zhong and C. G. Cassandras, "Distributed coverage control and data collection with mobile sensor networks," *IEEE Trans. Autom. Control*, vol. 56, no. 10, pp. 2445–2455, Oct. 2011.
- [2] X. Sun and C. G. Cassandras, "Optimal dynamic formation control of multi-agent systems in environments with obstacles," *Automatica*, vol. 73, pp. 169–179, Nov. 2016.
- [3] N. Michael, E. Stump, and K. Mohta, "Persistent surveillance with a team of mavs," in *Proc. IEEE/RSJ Intl. Conf. Intell. Robots Syst.*, 2011, pp. 2708–2714.
- [4] S. L. Smith, M. Schwager, and D. Rus, "Persistent monitoring of changing environments using a robot with limited range sensing," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2011, pp. 5448–5455.
- [5] Z. Shen and S. B. Andersson, "Tracking nanometer-scale fluorescent particles in two dimensions with a confocal microscope," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 5, pp. 1269–1278, Sep. 2011.
- [6] S. M. Cromer Berman, P. Walczak, and J. W. Bulte, "Tracking stem cells using magnetic nanoparticles," *Wiley Interdiscip. Rev. Nanomed. Nanobiotechnol.*, vol. 3, no. 4, pp. 343–355, 2011.
- [7] T. T. Ashley, E. L. Gan, J. Pan, and S. B. Andersson, "Tracking single fluorescent particles in three dimensions via extremum seeking," *Biomed. Opt. Exp.*, vol. 7, no. 9, pp. 3355–3376, 2016.
- [8] I. Rekleitis, V. Lee-Shue, A. P. New, and H. Choset, "Limited communication, multi-robot team based coverage," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2004, pp. 3462–3468, vol. 4.
- [9] Y. Elmaliach, N. Agmon, and G. A. Kaminka, "Multi-robot area patrol under frequency constraints," *Ann. Math. Artif. Intell.*, vol. 57, no. 3–4, pp. 293–320, 2009.
- [10] C. G. Cassandras, X. Lin, and X. Ding, "An optimal control approach to the multi-agent persistent monitoring problem," *IEEE Trans. Autom. Control*, vol. 58, no. 4, pp. 947–961, Apr. 2013.
- [11] X. Lin and C. G. Cassandras, "An optimal control approach to the multi-agent persistent monitoring problem in two-dimensional spaces," *IEEE Trans. Automat. Control*, vol. 60, no. 6, pp. 1659–1664, Jun. 2015.
- [12] N. Basilico, N. Gatti, and F. Amigoni, "Developing a deterministic patrolling strategy for security agents," in *Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. Intell. Agent Technol.*, 2009, pp. 565–572, vol. 2.
- [13] V. A. Huynh, J. J. Enright, and E. Frazzoli, "Persistent patrol with limited-range on-board sensors," in *Proc. IEEE Conf. Decis. Control*, 2010, pp. 7661–7668.
- [14] X. Yu, S. B. Andersson, N. Zhou, and C. G. Cassandras, "Optimal dwell times for persistent monitoring of a finite set of targets," in *Proc. Amer. Control Conf.*, 2017, pp. 5544–5549.
- [15] B. Horling and V. Lesser, "A survey of multi-agent organizational paradigms," *Knowl. Eng. Rev.*, vol. 19, no. 4, pp. 281–316, 2004.
- [16] J. Yu, S. Karaman, and D. Rus, "Persistent monitoring of events with stochastic arrivals at multiple stations," *IEEE Trans. Robot.*, vol. 31, no. 3, pp. 521–535, Jun. 2015.
- [17] T. Bektas, "The multiple traveling salesman problem: An overview of formulations and solution procedures," *Omega*, vol. 34, no. 3, pp. 209–219, 2006.
- [18] E. Stump and N. Michael, "Multi-robot persistent surveillance planning as a vehicle routing problem," in *Proc. IEEE Conf. Automat. Sci. Eng.*, 2011, pp. 569–575.
- [19] C. G. Cassandras, Y. Wardi, C. G. Panayiotou, and C. Yao, "Perturbation analysis and optimization of stochastic hybrid systems," *Eur. J. Control*, vol. 16, no. 6, pp. 642–661, 2010.
- [20] Y. Wardi, R. Adams, and B. Melamed, "A unified approach to infinitesimal perturbation analysis in stochastic flow models: The single-stage case," *IEEE Trans. Automat. Control*, vol. 55, no. 1, pp. 89–103, Jan. 2010.
- [21] M. Schwager, D. Rus, and J.-J. Slotine, "Decentralized, adaptive coverage control for networked robots," *Int. J. Robot. Res.*, vol. 28, no. 3, pp. 357–375, 2009.
- [22] M. Cao, A. S. Morse, C. Yu, B. Anderson, and S. Dasgupta, "Maintaining a directed, triangular formation of mobile autonomous agents," *Commun. Inf. Syst.*, vol. 11, no. 1, pp. 1–16, 2011.
- [23] K.-K. Oh and H.-S. Ahn, "Formation control and network localization via orientation alignment," *IEEE Trans. Autom. Control*, vol. 59, no. 2, pp. 540–545, Feb. 2014.
- [24] Y. Khazaeni and C. G. Cassandras, "Event excitation for event-driven control and optimization of multi-agent systems," in *Proc. IEEE Intl. Workshop Discr. Event Syst.*, 2016, pp. 197–202.
- [25] N. Zhou, X. Yu, S. B. Andersson, and C. G. Cassandras, "Optimal event-driven multi-agent persistent monitoring of a finite set of targets," in *Proc. IEEE Conf. Decision Control*, 2016, pp. 1814–1819.
- [26] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation and Control*. Boca Raton, FL, USA: CRC Press, 1975.
- [27] N. Zhou, C. G. Cassandras, X. Yu, and S. B. Andersson, "Decentralized event-driven algorithms for multiagent persistent monitoring," in *Proc. of IEEE Conf. Decis. Control* arXiv: 1708.06432.
- [28] D. P. Bertsekas, *Nonlinear programming*. Belmont, MA, USA: Athena scientific, 1999.
- [29] X. Yu and S. B. Andersson, "Effect of switching delay on a networked control system," in *Proc. IEEE Conf. Decision Control*, 2013, pp. 5945–5950.
- [30] X. Yu and S. B. Andersson, "Preservation of system properties for networked linear, time-invariant control systems in the presence of switching delays," in *Proc. IEEE Annu. Conf. Decision Control*, 2014, pp. 5260–5265.



Nan Zhou (S'14) received the B.S. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2011, the M.S. degree in control engineering from Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2014. He is currently a Ph.D. candidate in systems engineering at Boston University, Boston, MA, USA.

His research interests include optimal control of hybrid systems, cooperative control, and multi-agent systems, with applications in robotics, mobile sensor networks, and smart cities.

Dr. Zhou is a Student Member of the IEEE Control Systems Society and Robotics Automation Society.



Xi Yu (S'13) received the B.S. and Dipl. Ing. degrees in mechanical engineering from Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2010 and 2011, respectively. She is currently a Ph.D. candidate in mechanical engineering at Boston University, Boston, MA, USA.

Her research interests include networked control systems, multiagent systems with applications in nanoscale particle tracking.

Dr. Yu is a Student Member of the IEEE Control Systems Society.



Sean B. Andersson (SM'14) received the B.S. degree in engineering and applied physics from Cornell University, Ithaca, NY, USA, in 1994, the M.S. degree in mechanical engineering from Stanford University, Stanford, CA, USA, in 1995, and the Ph.D. degree in electrical and computer engineering from the University of Maryland, College Park, MD, USA, in 2003.

He has worked with AlliedSignal Aerospace and Aerovironment, Inc. and is currently an Associate Professor of mechanical engineering and of systems engineering with Boston University, Boston, MA, USA. His research interests include systems and control theory with applications in scanning probe microscopy, dynamics in molecular systems, and robotics.

Dr. Andersson received an NSF CAREER award in 2009 and since 2014, he has been an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL and since 2013, for the *SIAM Journal on Control and Optimization*.



Christos G. Cassandras (F'96) received the B.S. degree in engineering and applied science from Yale University, New Haven, CT, USA, in 1977, the M.S.E.E. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 1978, and the M.S. and Ph.D. degrees in applied mathematics from Harvard University, Cambridge, MA, USA, in 1979 and 1982, respectively.

He was with ITP Boston, Inc., Cambridge, from 1982 to 1984, where he was involved in the design of automated manufacturing systems. From 1984 to 1996, he was a faculty member with the Department of Electrical and Computer Engineering, University of Massachusetts Amherst, Amherst, MA, USA. He is currently a Distinguished Professor of Engineering with Boston University, Brookline, MA, USA, the Head of the Division of Systems Engineering, and a Professor of Electrical and Computer Engineering. He serves on several editorial boards and has been a Guest Editor for various journals. He specializes in the areas of discrete event and hybrid systems, cooperative control, stochastic optimization, and computer simulation, with applications to computer and sensor networks, manufacturing systems, and transportation systems. He has authored more than 380 refereed papers in these areas, and five books.

Dr. Cassandras is a Member of Phi Beta Kappa and Tau Beta Pi. He is also a Fellow of the International Federation of Automatic Control (IFAC). He was a recipient of several awards, including the 2011 IEEE Control Systems Technology Award, the 2006 Distinguished Member Award of the IEEE Control Systems Society, the 1999 Harold Chestnut Prize (IFAC Best Control Engineering Textbook), a 2011 prize, and a 2014 prize for the IBM/IEEE Smarter Planet Challenge competition, the 2014 Engineering Distinguished Scholar Award at Boston University, several honorary professorships, a 1991 Lilly Fellowship, and a 2012 Kern Fellowship. He was the Editor-in-Chief of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 1998 to 2009. He was the President of the IEEE Control Systems Society in 2012.