2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

Path planning with Incremental Roadmap Update for Visibility-based Target Tracking

Guillermo J. Laguna and Sourabh Bhattacharya

***Abstract*— In this paper, we address the visibility-based target tracking problem in which a mobile observer moving along a $p-$route, which we define as a fixed path for target tracking, tries to keep a mobile target in its field-of-view. By drawing a connection to the *watchman's route problem*, we find a set of conditions that must be satisfied by the $p-$route. Then we propose a metric for tracking to estimate a sufficient speed for the observer given the geometry of the environment. We show that the problem of finding the $p-$route on which the observer requires minimum speed is computationally intractable. We present a technique to find a $p-$route on which the observer needs at most twice the minimum speed to track the intruder and a reactive motion strategy for the observer.**

I. INTRODUCTION

Mobile robots have been extensively deployed in surveillance applications [1], [2]. This paper addresses a special class of problems in mobile surveillance called *target tracking*, which refers to the motion planning problem for a mobile observer that tries to keep a mobile target within its sensing range in an environment containing obstacles [3]. This is a well studied problem in the robotics [4], controls and computer vision communities [5], [6]. A detailed review regarding several formulations of the target-tracking problem is provided in [7], [8]. In general, a trajectory for the observer is obtained by optimizing a metric that models the tracking performance, for example, tracking time, distance from intruder, relative pose between the observer camera and the target, to name a few [9], [10].

In [11] the notion of mobile coverage to address the problem of placing mobile agents inside a polygon that can travel back and forth along a segment to cover that polygon is introduced. [12] leverages upon the concept of mobile coverage to propose tracking strategies for a team of observers that are restricted to move on a line-segment inside the polygon. Specifically, we show that $\lfloor n/4 \rfloor$ diagonal guards are sufficient to track a mobile intruder inside a polygon (where n is the number of vertices of the polygon). In contrary, our current work deals with the problem of path planning by designing the fixed trajectory of a mobile observer on which it can track the intruder while minimizing an appropriate metric.

A necessary condition for a prespecified path for the observer is that it should ensure coverage of the entire environment, so it should be a watchman's route, which is a closed trajectory from which an observer can "see" every region in the interior of an environment with obstacles [13]. In the *shortest watchman's route problem* the minimum length watchman's route is found. It can be solved in polynomial time when the region to be guarded is a simple polygon but it is NP-hard for polygons with holes [14]. In contrast to the shortest watchman's route problem, the parameter that governs the capability of a guard for target tracking is its maximum speed. Hence, we address the problem of minimizing **the guard's speed** required to track the intruder.

Guillermo J. Laguna and Sourabh Bhattacharya are affiliated with the Department of Computer Science and Department of Mechanical Engineering at Iowa State University, Ames, Iowa 50011, USA. {gjlaguna, sbhattac}@iastate.edu

The main contributions of this work are as follows: (i) We investigate a variant of the *watchman-route problem* in which a mobile observer restricted to move on a prespecified path tries to track a mobile intruder. To the best of our knowledge, this is the first work that draws a connection between the watchman route problem and the target-tracking problem. (ii) We propose a metric for tracking based on the geometric parameters of the environment that allows us to determine an upper bound on the speed of the observer required to persistently track a mobile intruder. (iii) We show that finding a path that minimizes the upper bound on the speed of the observer is computationally intractable. (iv) Consequently, we propose an iterative strategy to build a path based on the proposed tracking metric and the corresponding motion strategy for the guard.

The paper is organized as follows. In Section II, we present the problem formulation. In Section III, we define a metric for the speed required by the observer for persistent tracking. In Section IV, we simplify the problem of constructing a path for which the speed required by an observer to guarantee persistent tracking is minimized. In Section V, we present an approximation approach to construct a path for the observer. In Section VI, the motion strategy for the observer is presented, and we present the conclusions and future work in Section VII.

II. PROBLEM FORMULATION

Consider an environment that can be represented as a simply-connected polygon P. An unpredictable intruder I moves inside the environment with bounded speed. Let $x_I = x_I(t) \in P$, and $0 \leq v_I(t) \leq \overline{v}_I < \infty$ denote the instantaneous location and speed of the intruder at time t, respectively, where $\overline{v}_I$ denotes the maximum speed of I. There is a guard g in P assumed to have an omni-directional field-of-view with infinite range. The instantaneous location and speed of g at time t are denoted by $x_g = x_g(t) \in P$ and $0 \leq v_g(t) \leq \overline{v}_g < \infty$ respectively, where $\overline{v}_g$ is the maximum speed of g. g has the objective of maintaining a line-of-sight (LOS) with the intruder i.e., *persistent tracking*. Additionally, we assume that g is constrained to move on a prespecified path inside P, which we call a p-route (In light of [15], which refers to a similar problem as the paparazzi problem). Given $x_I(0)$ and $\overline{v}_I$, we investigate the problem of finding the p-route γ^*, and $x_g(0)$ on γ^* such that g has a motion strategy that guarantees persistent tracking while $\overline{v}_g$ is minimized. We want to find a fixed path $\gamma \subset P$ for g such that for any reflex corner $v_i \in V^{rf}(P) \subset V(P)$, where $V(P)$ and $V^{rf}(P)$ are the set of all vertices of P and the subset of reflex vertices of P respectively, there is a subset of points in γ such that when x_g is located at any of those points, the LOS between x_g and x_I does not cross any of the two edges of $E(P)$ (edge set of P) incident to v_i (an edge is incident to a vertex if such a vertex is an endpoint of the edge). When g is located at those points we say that "I cannot use v_i to break the LOS", and consequently, it "cannot escape". To determine the set of points of γ that g needs to reach to prevent I from breaking the LOS using the corner $v_i \in V^{rf}(P)$, we consider the *star region* of v_i, denoted by $R(v_i) \subset P$. It is the set of points in P that lie inside the region obtained by extending the edges of P incident to v_i that are visible from v_i. See Figure 1, the extension of the edges incident to v_2 are l_1 and l_3. However, since v_3 lies inside the region enclosed by l_1, l_3 and δP (δ stands for "boundary of"), such a region is not a star region, there is a region occluded by v_3, so $R(v_i)$ is enclosed by l_1, l_3, δP and l_2.

The edges incident to v_i along with their extensions inside P correspond to a *cut* from the watchman's route theory [16]. Based on the clockwise traversal of δP we can determine the orientation of each edge of P and each cut inherits the orientation of its corresponding edge. A cut separates the polygon into two sub-polygons. A point lies to the right (left) of a cut if the point lies locally to the right (left) in the sub-polygon separated by the cut. The underlying path of a watchman's route must have a point to the right of (or on) each cut. Otherwise, the edge that corresponds to such cut will not be visible from any point in the path. This is equivalent to say that every watchman route must visit every star region. Hence, γ needs to be the underlying path of a watchman's route or else there would be regions that are not visible from any point in γ.

III. A Metric for Tracking

Based on the distance between each pair of reflex corners and the path that g needs to travel to reach the corresponding star regions, we propose a metric to measure a sufficient speed that guarantees persistent tracking. Constructing the path of g such that $\overline{v}_g$ is minimized implies that the distance that g needs to travel for reaching the locations where it can prevent I from escaping must be minimized. Thus, γ consists of a set of connected line segments. Consequently, γ is represented using a graph G, where each line segment of γ corresponds to an edge in $E(G)$ (edge set of G), and the endpoints of those segments along with the points where γ intersects with itself correspond to the vertices in $V(G)$ (vertex set of G).

Consider the following scenario: P has two reflex vertices v_i and v_j. Assume that $\gamma \subset P$ is defined as an open path between $R(v_i)$ and $R(v_j)$ that does not visit the interior of the star regions. Let $p_i \in \gamma \cap \delta R(v_i)$ and $p_j \in \gamma \cap \delta R(v_j)$ be the endpoints of γ. Hence, $v_{p_i}, v_{p_j} \in V(G)$, where a vertex v_p is defined to be the vertex on G corresponding to $p \in \gamma$. p_i is the only location in γ where g can prevent I from using v_i to escape. The same situation occurs between p_j and v_j. Let x_g be any point along γ, and let $s_g^{v_i} = s_g^{v_i}(t)$ be the longest line segment lying entirely in P, such that $v_i \in s_g^{v_i}$ and x_g is an endpoint of $s_g^{v_i}$. We define $p^i(x_g) \in \delta P \backslash \{v_i\}$ as the opposite endpoint. Now, we define $s_{v_i} = s_{v_i}(t) \subset s_g^{v_i}$ as the directed segment from v_i to $p^i(x_g)$. As long as I lies to the left of (or at) s_{v_i}, the LOS between I and g is not broken by v_i, and visibility is lost as soon as I lies to the right of s_{v_i}. Preventing I from breaking the LOS is equivalent to prevent I from reaching the right side of any s_{v_i}. In Figure 1, a simple polygon is shown, g is located at $x_g \in \gamma$ and γ is represented as a chain of red segments. s_{v_1}, s_{v_2} and s_{v_3} are shown as directed green segments, each one corresponding to v_1, v_2 and v_3 respectively. The dashed segments represent the boundary of the star regions.

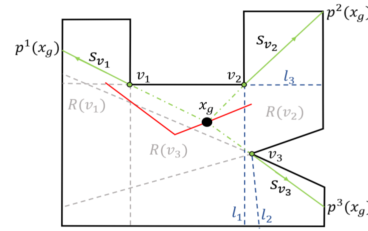

Fig. 1: Segments s_{v_1},s_{v_2} and s_{v_3}.

Let x_I be a location to the left (or at) of s_{v_i}. In general, given any watchman route with an underlying path γ represented as a graph G, given $\overline{v}_I$, $x_g \in \gamma$ and $v_i \in V^{rf}(P)$, we can determine a sufficient speed $\overline{v}_g^i(G)$ that prevents losing track of I when it approaches s_{v_i} as follows.

$$\overline{v}_g^i(G) = \overline{v}_I \frac{d(x_g, p_i)}{d(x_I, s_{v_i})} = \overline{v}_I \min_k \frac{d^k(x_g, p_i)}{d^k(x_I, s_{v_i})}, \qquad (1)$$

where $d^k(x_g, p_i)$ is the length of the sub-path $\gamma_k \subset \gamma$ between x_g and p_i (k stands for the k_{th} sub-path between x_g and p_i). Each γ_k is composed of a set of connected line segments $S^k_{x_g,p_i}$ between x_g and p_i. Thus, $d^k(x_I, s_{v_i}) = \max_{s^l \in S^k_{x_g,p_i}} \min\{d^l(x_I, y) : y \in s_{v_i}\}$, where $d^l(x_I, y)$ is the length of the shortest path inside P between x_I and s_{v_i} when $x_g \in s^l$. By definition, $d(x_I, s_{v_i})$ takes the length of the shortest path between x_I and s_{v_i} from placing the guard at the endpoints of each s^l. Thus, $\overline{v}_g^i(G) \geq \overline{v}_g^i(G)^*$ is always a sufficient speed for the g (and sometimes necessary) to guarantee that I cannot break the LOS by reaching s_{v_i}, where $\overline{v}_g^i(G)^*$ is the corresponding minimum (necessary) speed.

Consider the case where $x_g(0)$ and $x_I(0)$ are known, and $\frac{d(x_g,p_i)}{d(x_I,s_{v_i})} = \frac{d^k(x_g,p_i)}{d^k(x_I,s_{v_i})}$, with $d^k(x_I, s_{v_i}) = d^l(x_I, y)$, and $y \in s_{v_i}(x_g(0))$. Thus, g follows the path γ_k to reach p_i and $x_g(0)$ is an endpoint of $s^l = \min_{d^l(x_I,y)} S^k_{x_g,p_i}$. It seems like $\overline{v}_g^i(G)^* = \overline{v}_I \frac{d^k(x_g,p_i)}{d^k(x_I,s_{v_i})}$ but that is not necessarily true. Assume that g starts moving towards p_i (starting from $x_g(0)$)

at a speed $\overline{v}^i_g(G)$. Depending on the trajectory of g, s_{v_i} can rotate in a manner such that the distance between I and s_{v_i} increases even if I moves towards s_{v_i} at each moment.In such a scenario, although $\overline{v}^i_g(G)$ is sufficient to prevent the intruder from using v_i to escape, the necessary speed ($\overline{v}^i_g(G)^*$) to guarantee persistent tracking may be smaller than $\overline{v}^i_g(G)$.

When there is more than one reflex vertex in the environment, γ may contain more than one point that must be visited by g to prevent the LOS to be broken. Thus, g needs to move along γ as long as I tries to reach the segment s_{v_i} of each $v_i \in V^{rf}(P)$. Assume that $V^{rf}(P) = \{v_i, v_j\}$. $\overline{v}_g(G) \geq \overline{v}_I \frac{d(x_g,p_i)}{d(x_I,s_{v_i})}$ and $\overline{v}_g(G) \geq \overline{v}_I \frac{d(x_g,p_j)}{d(x_I,s_{v_j})}$ are sufficient conditions to guarantee persistent tracking. Hence, tracking may be lost if $x_g \in \gamma^k_{i,j} \subset \gamma$, and it needs to move towards p_i and p_j at the same time, which happens when the following condition is satisfied.

$$\overline{v}_g(G) < \overline{v}_I \frac{d^k(p_i,p_j)}{d^k(s_{v_i},s_{v_j})} \tag{2}$$

Consequently, $\overline{v}_g(G) \geq \overline{v}_I \frac{d^k(p_i,p_j)}{d^k(s_{v_i},s_{v_j})}$, with $d^k(s_{v_i},s_{v_j}) = \min\{d(a_i,a_j) : a_i \in s_{v_i} \text{ and } a_j \in s_{v_j}\}$, guarantees that g is able to persistently track I. Thus, $\overline{v}_g(G) = \overline{v}_I \frac{d^k(p_i,p_j)}{d^k(s_{v_i},s_{v_j})}$ is the lowest upper bound on $v_g(t)$. Since $\overline{v}_I$ is a constant, we want to minimize $\frac{d^k(p_i,p_j)}{d^k(s_{v_i},s_{v_j})}$. Notice that when $d^k(s_{v_i},s_{v_j}) = d(v_i,v_j)$, $\overline{v}^k_g(G) = \overline{v}^k_g(G)^*$.

Let $\overline{v}_g(G) = \overline{v}_I \frac{d^k(p_i,p_j)}{d^k(s_{v_i},x_I)+d^k(x_I,s_{v_j})}$, and assume that $x_g(0)$ is the location where $d^k(s_{v_i},s_{v_j})$ is defined. Then $d^k(p_i,x_g) = \frac{\overline{v}_g(G)d^k(s_{v_i},x_I)}{\overline{v}_I}$ with $d^k(s_{v_i},x_I) > 0$. The speed that guarantees persistent tracking $\overline{v}_g(G)$ is formulated in such a manner that as soon as I reaches $s_{v_i}(0)$, g reaches p_i. However, we know that g does not necessarily need to reach p_i when I approaches $s_{v_i}(t)$ since $d(x_I, s_{v_i}(x_g(0))) = 0$ at time $t > 0$ implies that the current segment $s_{v_i}(t)$ may be different from $s_{v_i}(x_g(0))$. Hence, even if $s_{v_i}(x_g(0))$ is reached, visibility may not be lost. Let I be located at $s_{v_i}(x_g(0))$, so $d^k(s_{v_j}(x_g(0)), x_I) = d(s_{v_i}(x_g(0)), s_{v_j}(x_g(0)))$ (recall that $d^k(s_{v_i}(x_g(0))), x_I) + d^k(x_I, s_{v_j}(x_g(0))) \geq d(s_{v_i}, s_{v_j})$) while g is located at $x_g(0)$. Thus, the LOS has not been broken yet, but x_g moving towards p_i while I tries to break the LOS does not to guarantee persistent tracking. Moreover, if we allow I to be located at the right of $s_{v_i}(x_g(0))$, visibility would be lost, so $x_I \in s_{v_i}(x_g(0))$ corresponds to the instant before persistent tracking is lost. Assume that visibility is not lost when $x_I \in s_{v_i}(0)$, and that $\overline{v}_g(G)$ is sufficient to prevent g from losing track of I. Hence, the distance between x_I and the region from which visibility may be lost from x_g should be $d^{lost} = \overline{v}_I \frac{d^k(p_i,x_g)}{\overline{v}_g(G)} \leq d(sv_i(x_g(0)), sv_j(x_g(0)))$ at least since $\overline{v}_g(G) = v_I \frac{d^k(p_i,p_j)}{d^k(s_(v_i)(x_g(0)),s_(v_j)(x_g(0)))} \geq v_I \frac{d^k(p_i,x_g)}{d^k(p_i,x_I)}$. Since $d^k(s_{v_j}, x_I) = d(s_{v_i}(0), s_{v_j}(0))$, it follows that $\overline{v}'_g(G) \geq \overline{v}_I \frac{d^k(p_i,p_j)}{2d(s_{v_i},s_{v_j})}$ is a sufficient speed that guarantees persistent tracking. However, for our original problem, the assumption that visibility is not lost when $x_I \in s_{v_i}$ is not true, so $\overline{v}'_g(G) < \overline{v}^*_g(G) \leq \overline{v}_g(G)$. Although, we do not know the precise value of $\overline{v}^*_g(G)$, we guarantee that $\frac{\overline{v}_g(G)}{\overline{v}^*_g(G)} < \frac{\overline{v}_g(G)}{\overline{v}'_g(G)} = 2$. Hence, our proposed metric gives us a speed that is at most twice the optimal.

IV. Considerations for the Design of γ

$V(G)$ consists of vertices that correspond to the endpoints of the segments of γ, and to intersection points between segments of γ. In this section, we prove that G can always be a tree. Moreover, we prove there is a point $p_i \in R(v_i) \cap \gamma$ such that the minimum speed required by g to prevent I from breaking the LOS by reaching any point in $R(v_i) \cap \gamma$ is achieved at p_i.

By definition, γ has at least one point at the right of (or at) each cut. Moreover, a cut c_{i_1} dominates another cut c_{i_2}, with $i_1 \neq i_2$ if all points in P to the right of c_{i_1} are also to the right of c_{i_2}, and it is called an *essential cut* if it is not dominated by any other cut [14], so there are no points inside P that I can use to break the LOS on the right of an essential cut.

Let S_c be the set of essential cuts of P and $S^{right}_R = \{R^{right}_j \subset P : c_j \in S_c\}$ be the set of regions R^{right}_j located at the right of each $c_j \in S_c$. We define a set of subpaths $S_\gamma = \{\gamma^{right}_{p^1,p^2} \subset \gamma \cap \bigcup_{R^{right}_j \in S^{right}_R} R^{right}_j : p^1, p^2 \in \bigcup_{c_j \in S_c} c_j$ are the endpoints of $\gamma^{right}_{p^1,p^2}\}$. By the definition of an essential cut, for each $c_j \in S_c$ there is a star region $R(v_i)$ such that $c_j \subset \delta R(v_i)$, so $R(v_i) \subseteq R^{right}_j$. Moreover, there is no other star region besides $R(v_i)$ that can only be reached at R^{right}_j. Thus, g does not need to travel along any $\gamma^{right}_{p^1,p^2}$, it just needs to reach each p^1 and p^2 since the corresponding $R(v_i)$ regions can be covered from the endpoints of each $\gamma^{right}_{p^1,p^2}$. Hence, each $\gamma^{right}_{p^1,p^2}$ is replaced by the line segment γ'_{p^1,p^2} defined by p^1 and p^2. Clearly, the length of $\gamma'_{p^1_i,p^2_i}$ is smaller than the length of $\gamma^{right}_{p^1_i,p^2_i}$. Thus, the minimum sufficient speed to guarantee persistent tracking for the modified path is smaller or equal to $\overline{v}_g(G)$, and we can always obtain an equivalent path γ' that never traverses the essential cuts.

A. Representatives of Star Regions

By definition, γ intersects every $R(v_i)$. In general, each $R(v_i) \cap \gamma$ is a disconnected region and contains more than one point. Consequently, there is more than one point in γ that g can reach to prevent I from breaking the LOS after reaching s_{v_i}. Lemma 1 shows that regardless of the size of each intersection $R(v_i) \cap \gamma$, and regardless of its number of connected components, we just need to consider a single point in each $R(v_i) \cap \gamma$ that must be visited by g when I approaches s_{v_i} such that this restriction does not increase $\overline{v}_g(G)$. Given the underlying path of a watchman route γ, $V(P)$ and $\overline{v}_g(G)$, we define the collection of sets of intersection points $S_{V(P),\gamma} = \{s_{i,\gamma} : R(v_i) \cap \gamma\}$, and a set of representative points $S_{p,\gamma}$ of $S_{V(P),\gamma}$ as follows. For each $p_i \in S_{p,\gamma}$, there is a $s_{i,\gamma} \in S_{V(P),\gamma}$, such that $p_i \in s_{i,\gamma}$ and $|S_{p,\gamma} \cap s_{i,\gamma}| = 1$. In the interest of space, the proof of Lemma 1 is presented in the addendum [17].

Lemma 1: There is a set of representative points $S_{p,\gamma}$ of $S_{V(P),\gamma}$, such that the minimum sufficient speed to guarantee persistent tracking when g is forced to visit each $p_i \in S_{p,\gamma}$ to prevent I from breaking the LOS when approaching s_{v_i}, is equal to $\overline{v}_g(G)$.

B. Equivalent Tree

Based on Lemma 1, we redefine the graph G that represents γ. Let $V(G)$ include vertices that correspond to the representatives of star regions (grouped in $V_{track}(G) = \{v_{p_i} : \exists p_i \in S_{p,\gamma}\}$). Then we show that for any γ represented as a graph G, there is a tree G' such that if g is constrained to move along γ' (that corresponds to G'), the minimum sufficient speed that guarantees persistent tracking is not greater than $\overline{v}_g(G)$. This result allows us to reduce the problem of designing any $p-$route to the problem of designing a path that can be represented as a tree, so there is only one path between any pair of representative points of star regions. Algorithm 1 shows a procedure to find a graph $\hat{G}$ from G such that G' can be trivially obtained from $\hat{G}$. Lemma 2 proves that the minimum sufficient speed to guarantee persistent tracking when g is constrained to move along a path that corresponds to $\hat{G}$ is equal to $\overline{v}_g(G)$, and Theorem 1 proves the same for G'. In the interest of space, the proofs for Lemma 2 and Theorem 1 are provided in the addendum [17].

In Algorithm 1 $\mathbb{S}'_\gamma = \{\gamma^k_{i,j} : p_i, p_j \in S_{p,\gamma}$ and $\exists k$ such that $\gamma^k_{i,j} \subseteq \gamma\}$. Moreover, given γ and a path $\gamma^j_{i,j} \in \mathbb{S}'_\gamma$, we define $S_G(\gamma^k_{i,j})$ as the set of all possible subgraphs G_{sub} of G such that the only difference between each G_{sub} and G is that the path in G_{sub} that corresponds to $\gamma^k_{i,j}$ (called $G^k_{i,j}$) does not exist. Hence, for each G_{sub}, a subset of edges in $E(G^k_{i,j})$ is absent. Algorithm 1 is exhaustive. It considers all possible graphs G_{sub}, and relies on obtaining the minimum $\overline{v}_g(G_{sub})$. There is a recursive procedure of eliminating such paths in G until it becomes a disconnected graph. The subgraph that corresponds to the minimum sufficient speed to guarantee persistent tracking along with the minimum speed is returned. Algorithm 1 is used only to prove the existence of a p-route equivalent to γ so its computational complexity is irrelevant.

Lemma 2: For any γ represented as a graph G, Algorithm 1 returns a subgraph $\hat{G}$ (and the corresponding route $\hat{\gamma}$) such that $\overline{v}_g(\hat{G}) = \overline{v}_g(G)$ when g is constrained to move along $\hat{\gamma}$.

Theorem 1: For any γ, represented as a graph G, there is an acyclic subgraph G' (and the corresponding path γ') such that $\overline{v}_g(G') = \overline{v}_g(G)$ when g is constrained to move along γ'.

Theorem 1 uses $\hat{G}$ from Algorithm 1, which is a subgraph in which the value $z(\gamma^k_{i,j})$ for any $(v_i, v_j) \in V_{track}(\hat{G})$ is at most equal to $\overline{v}_g(\hat{G})$, to get a tree equivalent to G. This property of $\hat{G}$ allows a "safe" removal of some edges of $\hat{G}$ to obtain an equivalent G' as proved in Theorem 1. The results from lemmas 1, 2 and Theorem 1 imply the existence of an equivalent $p-$route $\hat{\gamma}$ such that its corresponding graph $\hat{G}$ is a tree. The results do not depend on the metric defined in Section III. Thus, the results apply when trying to find γ^* and its corresponding graph G^*. From Lemma 1 and Theorem 1, for any watchman route with an underlying path γ^* represented as G^*, there is always an equivalent path γ' represented as a tree G'. Moreover, γ^* has a set of representative points S_{p,γ^*} such that the minimum sufficient speed to guarantee persistent tracking when g is forced to visit each $p_i \in S_{p,\gamma}$ to prevent I from breaking the LOS is $\overline{v}^*_g(G^*)$. Hence, the problem of designing γ^* reduces to the problem of designing γ' represented as a tree, such that there is a representative point of γ' inside each star region. Assume that the representative points are already known and all the vertices in $V(G')$ correspond to the representative points. Thus, the problem is to find the edges of G' such that $\overline{v}_g(G')$ is minimum. Let G_{com} be a complete graph such that $V(G_{com}) = G'$. Designing γ' is then equivalent to find the spanning tree G' of G_{com} such that $\overline{v}_g(G')$ is minimum, an exponential problem in nature [18].

Algorithm 1 Equivalent Graph

```
Input: G
Output: Ĝ, v̄_g
if G is disconnected then
    return Ĝ ← G and v̄_g(Ĝ) ← ∞
end if
γ̂ ← arg max_{γ^k_{i,j} ∈ S'_γ} z(γ^k_{i,j})
v̄_g(Ĝ) ← v̄_I z(γ̂)
for each G_sub ∈ S_G(γ̂) do
    G_tem, v_tem ← call Equivalent Graph(G_sub)
    if v_tem < v̄_g(Ĝ) then
        v̄_g(Ĝ) ← v_tem and Ĝ ← G_tem
    end if
end for
return v̄_g(Ĝ) and Ĝ
```

V. Construction of γ

Based on the results from the previous section, we propose a technique to construct an approximate path γ on which the sufficient speed to guarantee persistent tracking can be decreased. Algorithm 2 returns a path γ with its corresponding graph G. The path is constructed incrementally. At each iteration, Algorithm 2 considers a different star region, so it finds a point inside it and connects it to the current p-route such that the overall upper bound on the speed to guarantee persistent tracking (for the current p-route) is minimized (local minimization). Next, Algorithm 3 tries to improve the required speed of the guard by moving some representative points of the sub-path that yields the minimum sufficient speed. The structure of the path remains unchanged, only the location of those vertices, which are allowed to move along any of the edges incident to them may change. This process continues until no improvement in the speed is achieved.

Given $R(v_i)$ and $R(v_j)$, with $v_i, v_j \in V^{rf}(P)$, we define $d_R(i,j) = \min\{d(p_i, p_j) : p_i \in R(v_i)$ and $p_j \in R(v_j)\}$. Let $S_d = \{d_R(i,j)/d(v_i, v_j) : v_i \in V^{rf}(P), v_j \in V^{rf}(P)$ and $i \neq j\}$. Let S^{sort}_d be a copy of S_d but with its elements sorted in a non-decreasing order. Also, given a

p-route and its corresponding graph, we define $\gamma(e_l)$ as the sub-path in γ that corresponds to the edge e_l of the associated graph.

Algorithm 2 returns a path γ with the corresponding graph G that represents it. However, it only finds the minimum value when the representative points can only be selected along the line segments that correspond to the edges in G, but there may be cases in which $\overline{v}_g(G)$ can be improved even further by not restricting the representative points to be defined only along the line segments already defined in γ. Thus, Algorithm 2 is refined by loosening the aforementioned constraint. Algorithm 3 "perturbs" some of the points p along a given direction until $\overline{v}_g(G)$ does not decrease, so it is based on a discretized adjustment of the location of such points while maintaining the connections between them.

Figure 2 shows an example implementation of Algorithm 2. $S_d^{sort} = \{z_{3,4}, z_{1,3}, z_{1,4}, z_{2,3}, z_{2,4}, z_{1,2}\}$. In the first iteration, p_4 is arbitrarily chosen as a point inside $R(v_4)$. In Figure 2 (a), the original location of p_4 is shown in addition to the location of p_3, Algorithm 2 finds the minimum distance between p_3 and p_4, and the corresponding path. In the same figure, the new location of p_4 is shown after its location is adjusted according to Algorithm 3 since it allows p_3 and p_4 to move along the path between them in order to minimize $z(\gamma_{3,4})$. The next vertex to be selected by the **For** loop (line 7) is v_{p_1} (Figure 2 (b)). p_1 is found as the point in $R(v_1)$ that lies closer to the edge between p_3 and p_4. Thus, $\overline{v}_g(G) = \overline{v}_I z(\gamma_{1,4})$. After the execution of Algorithm 3, p_4 moves to the left until $z(\gamma_{1,4}) = z(\gamma_{4,3})$. Next, the only remaining point is p_2 which is added to the current γ (Figure 2 (c)). Algorithm 3 finds the location of p_2 and the location where the edge incident to p_2 intersects the edge between p_1 and p_4 such that $\max\{z(\gamma_{2,1}), z(\gamma_{2,3}), z(\gamma_{2,4})\}$ is minimized. However, connecting p_2 to the current γ does not affect $\overline{v}_g(G) = \overline{v}_I z(\gamma_{1,4}) = \overline{v}_I z(\gamma_{4,3})$.

VI. Motion Strategy

In this section, we describe the motion strategy followed by g to guarantee persistent tracking of I. For the persistent tracking task, we need to know $x_I(0)$ to avoid an initial location $x_g(0)$ for which x_I is not inside the visibility polygon of g. Moreover, for each $v_i \in V_{track}(G)$, $x_g(0)$ must guarantee that g can reach each point $p_i \in \gamma$ before (or at the same time) that I reaches $s_{v_i}(x_g(0))$ from $x_I(0)$.

Since $\overline{v}_g(G) \geq \overline{v}_I \frac{d(x_g(0), p_i)}{d(x_I(0), s_{v_i})}$, for each $v_i \in V_{track}(G)$, $d(x_g(0), p_i) \leq \frac{\overline{v}_g(G)}{\overline{v}_I} d(x_I(0), s_{v_i})$ for each $v_i \in V_{track}(G)$. Hence, to determine $x_g(0)$ we require to compute the sub-paths $\gamma_i(\overline{v}_g(G)) \subseteq \gamma$ defined as the set of points along γ such that the distance between each $p \in \gamma_i(\overline{v}_g(G))$ and p_i is no greater than $\frac{\overline{v}_g(G)}{\overline{v}_I} d(x_I(0), s_{v_i})$. The set $\gamma_{init} = \bigcap_{v_i \in V_{track}(G)} \gamma_i(\overline{v}_g(G))$ trivially contains all the points such that $x_g(0) \in \gamma_{init}$ guarantees that g can persistently track I. $\gamma_{init} \neq \emptyset$ since otherwise a point $p^I_{init} \in P$ and a point $p^g_{init} \in \gamma$ such that for $x_I(t) = p^I_{init}$ and $x_g(t) = p^g_{init}$ there is a p_j with $v_j \in V_{track}(G)$, and $\bigcap_{v_i \in V_{track}(G) \setminus \{v_j\}} \gamma_i(\overline{v}_g(G)) \neq \emptyset$ but $\bigcap_{v_i \in V_{track}(G)} \gamma_i(\overline{v}_g(G)) = \emptyset$ would exist. Thus, there

Algorithm 2 Approximate Route

1: **Input**: $S_d^{sort}, P, \overline{v}_I$
2: **Output**: G, γ
3: $V(G) \leftarrow \emptyset, E(G) \leftarrow \emptyset, \gamma \leftarrow \emptyset, G_{temp} \leftarrow G, \gamma_{temp} \leftarrow \gamma$
4: **while** S_d^{sort} is non-empty **do**
5: $\quad p_i, p_j \leftarrow$ points that correspond to the first element $z_{i,j} = d_R(i,j)/d(v_i, v_j) \in S_d^{sort}$, $S_d^{sort} \leftarrow S_d^{sort} \backslash \{d_R(i,j)/d(v_i, v_j)\}$
6: $\quad S_v \leftarrow$ vertices in $\{v_{p_i}, v_{p_j}\}$ that are not in $V(G_{temp})$
7: $\quad$ **for** each $v_{p_k} \in S_v$ **do**
8: $\quad\quad \overline{v}_g(G_{temp}) \leftarrow \infty, \gamma' \leftarrow \gamma_{temp}$, $G' \leftarrow G_{temp}$
9: $\quad\quad$ **for** each $e_l \in E(G')$ **do**
10: $\quad\quad\quad p_m, p_k \leftarrow \underset{p_m \in \gamma(e_l), p_k \in R(v_k)}{\arg\min} \max\{\frac{d(p_a, p_k)}{d(s_{v_a}, s_{v_k})} :$ $\gamma_{m,k} \cap \gamma' = \{p_m\}$ and $v_{p_a} \in V_{track}(G')\}$
11: $\quad\quad\quad \gamma_{aux} \leftarrow \gamma' \cup \gamma_{m,k}$, $G_{aux} \leftarrow$ graph representing γ_{aux}, $(G_{aux}, \gamma_{aux}) \leftarrow$ call Algorithm 3
12: $\quad\quad\quad$ **if** $\overline{v}_g(G_{aux}) \leq \overline{v}_g(G_{temp})$ **then**
13: $\quad\quad\quad\quad G_{temp} \leftarrow G_{aux}$, $\gamma_{temp} \leftarrow \gamma_{aux}$
14: $\quad\quad\quad$ **end if**
15: $\quad\quad$ **end for**
16: $\quad$ **end for**
17: **end while**
18: $G \leftarrow G_{temp}$, $\gamma \leftarrow \gamma_{temp}$

Algorithm 3 Approximate Location of p_i

Input: $G_{aux}, \gamma_{aux}, P, \overline{v}_I$
Output: updated G_{aux}, γ_{aux}
$continue \leftarrow true$
while $continue = true$ **do**
$\quad \hat{G} \leftarrow G_{aux}$, $\hat{\gamma} \leftarrow \gamma_{aux}$, $\gamma_{b,c} \subseteq \hat{\gamma} \leftarrow$ path such that $\overline{v}_I z(\gamma_{b,c}) = \overline{v}_g(\hat{G})$ and $v_{p_b}, v_{p_c} \in V_{track}(\hat{G})$, $S_p \leftarrow \{p_b, p_c\} \cup \{p_\alpha \in \gamma_{b,c} : v_{p_\alpha} \in V(\hat{G}) \backslash V_{track}(\hat{G})\}$
$\quad$ **for** each $p_i \in S_p$ **do**
$\quad\quad G_{best} \leftarrow \hat{G}$, $\gamma_{best} \leftarrow \hat{\gamma}$
$\quad\quad$ **for** each $e_l \in E(\gamma_{aux})$ incident to v_{p_i} **do**
$\quad\quad\quad G_z \leftarrow G_{best}$, $\gamma_z \leftarrow \gamma_{best}$
$\quad\quad\quad$ **if** $v_{p_i} \in V_{track}(G_z)$ **then**
$\quad\quad\quad\quad \overline{v}_g(G_z) \leftarrow \min_{\hat{i}} \max\{\overline{v}_I \frac{d(p_{\hat{i}}, p_a)}{d(s_{v_{p_{\hat{i}}}}, s_{v_{p_a}})} :$ $p_{\hat{i}} \in \gamma(e_l) \cap R(v_i)$ and $v_{p_a} \in V_{track}(G_z)\}$
$\quad\quad\quad$ **else**
$\quad\quad\quad\quad \overline{v}_g(G_z) \leftarrow \min_{\hat{i}} \max\{\overline{v}_I \frac{d(p_{\hat{i}}, p_a)}{d(s_{v_{p_{\hat{i}}}}, s_{v_{p_a}})} :$ $p_{\hat{i}} \in \gamma(e_l)$ and $v_{p_a} \in V_{track}(G_z)\}$
$\quad\quad\quad$ **end if**
$\quad\quad\quad$ **if** $\overline{v}_g(G_z) < \overline{v}_g(\hat{G})$ **then**
$\quad\quad\quad\quad \hat{G} \leftarrow G_z$, $\hat{\gamma} \leftarrow \gamma_z$
$\quad\quad\quad$ **end if**
$\quad\quad$ **end for**
$\quad$ **end for**
$\quad$ **if** $\overline{v}_g(\hat{G}) < \overline{v}_g(G_{aux})$ **then**
$\quad\quad G_{aux} \leftarrow \hat{G}$, $\gamma_{aux} \leftarrow \hat{\gamma}$, $continue \leftarrow true$
$\quad$ **else**
$\quad\quad continue \leftarrow false$
$\quad$ **end if**
end while

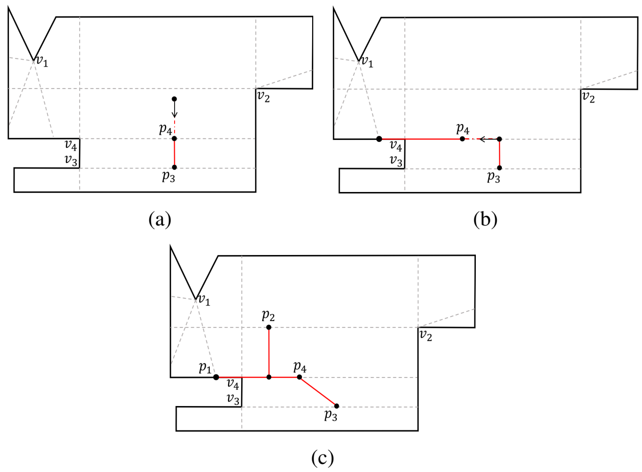

Fig. 2: (a) p_3 is found. (b) p_1 is found and connected to p_4. (c) p_2 is found and the resulting path is shown.

is a $p_k \in \gamma$ for which $v_k \in V_{track}(G)$ and $j \neq k$ such that $\overline{v}_I \frac{d(p_j,p_k)}{d(s_{v_j},x_I(t))+d(s_{v_k},x_I(t))} > \overline{v}_g(G)$, which is not possible since, by definition, $\overline{v}_I \frac{d(p_j,p_k)}{d(s_{v_j},s_{v_k})} \leq \overline{v}_g(G)$ and $\frac{d(p_j,p_k)}{d(s_{v_j},x_I(t))+d(s_{v_k},x_I(t))} \leq \frac{d(p_j,p_k)}{d(s_{v_j},s_{v_k})}$ from triangular inequality.

We define the intruders velocity as $\dot{x}_I = v_I \hat{u}$, where $\hat{u}$ is a unit vector. The orientation of the vector (representing the heading direction of $\hat{u}$ is denoted by $\theta \in [0, 2\pi)$. We arbitrarily define a Cartesian global reference frame in P with X and Y as the horizontal and vertical axes respectively. θ is defined with respect to the X axis. x_g changes when $d(x_I, s_{v_i}) < \frac{\overline{v}_I}{\overline{v}_g} d(x_g, p_i)$ for any $v_i \in V_{track}(G)$. Otherwise, $v_g = 0$. Let $V'_{track} \subseteq V_{track}(G)$ be the set of vertices such that for each $v_i \in V'_{track}$, $d(x_I, s_{v_i}) < \frac{\overline{v}_I}{\overline{v}_g} d(x_g, p_i)$ and $\hat{u} \cdot \overrightarrow{s_{I,v_i}} > 0$, where s_{I,v_i} is the first segment of the shortest path between x_I and s_{v_i}, and $\overrightarrow{s_{I,v_i}}$ is the vector representing s_{I,v_i} with x_I as its origin. After V'_{track} is obtained we arbitrarily select any $v_i \in V'_{track}$. Based on the direction of the velocity vector of I and its current location, we know that at the next instant, $d(x_I, s_{v_i})$ is smaller than $\frac{\overline{v}_I}{\overline{v}_g} d(x_g, p_i)$. Let $\gamma_{x_g,p_i} \subseteq \gamma$ be the path between x_g and p_i. Thus, after $v_i \in V'_{track}$ is selected, $\dot{x}^i_g = \overline{v}_g$, where $\dot{x}^i_g$ is the speed of g along γ_{x_g,p_i}.

VII. Conclusions

In this work we addressed the problem in which a single observer must maintain an unpredictable intruder within its sensing range all the time. The observer is constrained to move along a fixed trajectory. By leveraging results from the watchman's route problem to guarantee mobile coverage of the environment, we proposed a strategy to design a $p-$route, which is the fixed path of the observer. The strategy builds a path that minimizes the speed required to maintain persistent tracking. We showed that finding the optimal $p-$route is computationally intractable, so we proposed a procedure to find an approximate one. To this end, a target-tracking metric to estimate the speed required by the observer given the geometry of the environment was proposed. Finally, a reactive motion strategy for the observer given its corresponding $p-$route was shown. As a future work, we plan to improve the design procedure to find a $p-$route closer to the optimal.

References

[1] S. Witwicki, J. C. Castillo, J. Messias, J. Capitan, F. S. Melo, P. U. Lima, and M. Veloso, "Autonomous surveillance robots: A decision-making framework for networked multiagent systems," *IEEE Robotics & Automation Magazine*, vol. 24, no. 3, pp. 52–64, Sept 2017.

[2] P. Bazydło, J. Kacprzyk, and K. Lasota, "Global path planning for a specialized autonomous robot for intrusion detection in wireless sensor networks (wsns) using a new evolutionary algorithm," in *Trends in Advanced Intelligent Control, Optimization and Automation*, W. Mitkowski, J. Kacprzyk, K. Oprzkedkiewicz, and P. Skruch, Eds. Springer International Publishing, 2017, pp. 503–513.

[3] S. M. LaValle, H. H. Gonzalez-Banos, C. Becker, and J. C. Latombe, "Motion strategies for maintaining visibility of a moving target," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, Apr 1997, pp. 731–736.

[4] L. E. Parker and B. A. Emmons, "Cooperative multi-robot observation of multiple moving targets," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, vol. 3, Apr 1997, pp. 2082–2089 vol.3.

[5] R. K. Williams and G. S. Sukhatme, "Observability in topology-constrained multi-robot target tracking," in *Proceeding of IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1795–1801.

[6] I. S. Kim, H. S. Choi, K. M. Yi, J. Y. Choi, and S. G. Kong, "Intelligent visual surveillance — a survey," *International Journal of Control, Automation and Systems*, vol. 8, no. 5, pp. 926–939, Oct 2010.

[7] S. Bhattacharya and S. Hutchinson, "A cell decomposition approach to visibility-based pursuit evasion among obstacles," *The International Journal of Robotics Research*, vol. 30, no. 14, pp. 1709–1727, Sep. 2011.

[8] R. Zou and S. Bhattacharya, "On optimal pursuit trajectories for visibility-based target-tracking game," *IEEE Transactions on Robotics*, 2018.

[9] R. Zou and S. Bhattacharya, "Visibility-based finite-horizon target tracking game," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 399–406, Jan 2016.

[10] R. Murrieta-Cid, L. Munoz-Gomez, M. Alencastre-Miranda, A. Sarmiento, S. Kloder, S. Hutchinson, F. Lamiraux, and J. Laumond, "Maintaining visibility of a moving holonomic target at a fixed distance with a non-holonomic robot," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Aug 2005, pp. 2687–2693.

[11] G. T. Toussaint, "Computational geometric problems in pattern recognition," in *Pattern Recognition Theory and Applications*, J. Kittler, K. S. Fu, and L.-F. Pau, Eds. Dordrecht: Springer Netherlands, 1982, pp. 73–91.

[12] G. J. Laguna, R. Zou, and S. Bhattacharya, "Target tracking on triangulation graphs," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 460–465.

[13] W. pang Chin and S. Ntafos, "Optimum watchman routes," *Information Processing Letters*, vol. 28, no. 1, pp. 39 – 44, 1988.

[14] W.-P. Chin and S. Ntafos, "Shortest watchman routes in simple polygons," *Discrete & Computational Geometry*, vol. 6, no. 1, pp. 9–31, Mar 1991.

[15] M. Jenkin and G. Dudek, "The paparazzi problem," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3. IEEE, 2000, pp. 2042–2047.

[16] S. Carlsson, H. Jonsson, and B. J. Nilsson, "Finding the shortest watchman route in a simple polygon," *Discrete & Computational Geometry*, vol. 22, no. 3, pp. 377–402, Oct 1999.

[17] G. Laguna and S. Bhattacharya. (2019) Addendum for IEEE/RSJ IROS 2019 publication. [Online]. Available: https://bit.ly/2Iz7w2z

[18] P. W. Shor, "A new proof of Cayley's formula for counting labeled trees," *Journal of Combinatorial Theory, Series A*, vol. 71, no. 1, pp. 154–158, 1995.