2015 IEEE 54th Annual Conference on Decision and Control (CDC)
December 15-18, 2015. Osaka, Japan

Self-triggered Pursuit of a Single Evader

Saad A. Aleem, Cameron Nowzari, and George J. Pappas

Abstract— **This paper studies a continuous-time pursuit-evasion problem involving a single pursuer and a single evader on a plane. In contrast to other works that study this problem, we are interested in developing a pursuit strategy that doesn't require continuous, or even periodic, information about the position of the evader. To this end, we propose a self-triggered control strategy such that the pursuer can autonomously decide, based on out-dated information, when new samples of the evader's position is required in order to satisfy desired performance metrics. Our proposed algorithm guarantees capture of the evader in finite time with a finite number of sporadic updates without sacrificing any performance in terms of guaranteed time to capture as compared to classic algorithms that assume continuous information is available at all times. Simulations illustrate our results.**

I. Introduction

Pursuit and evasion strategies are widely observed in nature and play an important part in shaping predator-prey behaviors [1]. In engineering, such problems have been the subject of much attention in combat games [2] and more recently in the study of robotic systems for search and rescue missions and motion planning involving adversarial elements (see [3] for an overview of recent results). This paper studies a continuous-time pursuit-evasion problem involving a single pursuer and a single evader, where the goal of the pursuer is to capture the evader. In the past, treatment of pursuit-evasion problems usually involved continuous or periodic sensing/control updates for the agents. Instead, we want to relax this requirement by taking actions (e.g., control updates and sensing) only when necessary. Our objective is to design a self-triggered update policy for the pursuer that allows it to decide autonomously when fresh, up-to-date information about the evader's location is required in order to guarantee its capture.

Literature review: In the literature, pursuit-evasion problems have been studied extensively in the context of differential games [4], [5]. The treatment of differential games as optimal control problems involves optimal pursuit strategies by finding instantaneous control actions for the pursuer that continuously track the evader in order to capture it time-efficiently [6], [7]. In [8], sufficient conditions are derived for a single pursuer to capture an evader where the agents satisfy certain initial conditions, have equal maximum speeds and are constrained to move within the nonnegative quadrant of $\mathbb{R}^2$. In [9], upper and lower bounds on the time-to-capture have been discussed where the agents are constrained in a circular environment. These pursuit strategies have been generalized and extended by the authors of [10] to guarantee capture using multiple pursuers in an unbounded environment $\mathbb{R}^n$, as long as the evader is initially located inside the convex hull of the pursuers. In the context of robotic systems, visibility-based pursuit-evasion has garnered a lot of interest [11], [12], [13], [14]. In these problems, the pursuer is visually searching for an unpredictable evader that can move arbitrarily fast in a simply connected polygonal environment. Similar problems have been studied in [15], [16], [17], where visibility limitations are introduced for the pursuers and the evader. A related problem has been discussed in [18], where the agents can move in $\mathbb{R}^2$ but each agent has limited range of spatial sensing.

A common theme of the analysis in the mentioned works is the assumption of continuous tracking of the evader providing location information at all times. Unfortunately, the availability of continuous information and control updates is unrealistic, especially in the context of robotics or cyber-physical systems. In contrast to previous methods, we are interested in guaranteeing capture while relaxing this requirement. To this end, our aim is to opportunistically compensate for the reduced sensing effort while still achieving the desired objective. Recently, studies related to triggered control laws have received great interest in the control of networked dynamical systems. These methods are aimed at analyzing the cost to make up for less computation or communication effort on part of the agents, while achieving a desired task with a guaranteed level of performance of the system. An overview of the recent results can be found in [19]. Of particular relevance to this paper are works that study self-triggered [20], [21], [22], [23] or event-triggered [24], [25], [26], [27] implementations of decentralized strategies. In contrast to conventional time-driven approaches, strategies based on triggered control schemes study how information could be sampled for control purposes where the agents act in an opportunistic fashion to meet their desired objective [28].

Contribution: In this work, we apply the framework of triggered control to design a self-triggered pursuit policy which guarantees capture of the evader. Based on the latest observation of the evader, the pursuer computes the (sleep) duration for which it can follow its current trajectory without having to sense the location of the evader. Our analysis naturally accounts for the worst-case evader strategies, so the pursuer does not need to access the evader's dynamics in order to capture it. We then study the trade-off between more greedy strategies that can generally result in faster capture of the evader at the cost of more frequent samples of its position.

The authors are with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA, {aleems,cnowzari,pappasg}@seas.upenn.edu

Organization: The problem formulation and its mathematical model are presented in Section II. In Section III, we present the design of self-triggered update duration for the pursuer. This is followed by trade-off analysis between the number of updates and time-to-capture in Section IV and the results of simulations in Section V. The readers are encouraged to go over the detailed analysis of our problem in the Appendix.

Notation: $\mathbb{R}^n$ denotes $n-$dimensional Euclidean space and $\|\cdot\|$ is the Euclidean distance. We let $\mathbb{R}_{>0}$, $\mathbb{R}_{\geq 0}$ and $\mathbb{Z}_{\geq 0}$ to be the sets of positive real, nonnegative real and nonnegative integer numbers, respectively.

II. Problem statement

We consider a system with a single pursuer P and a single evader E. At any given time t, the position of the evader is given by $r_e(t) \in \mathbb{R}^2$ and its velocity is given by $u_e(t) \in \mathbb{R}^2$ with $\|u_e(t)\| \leq v_e$, where $v_e > 0$ is the maximum speed of the evader. Similarly, the position and velocity of the pursuer are given by $r_p(t)$ and $u_p(t)$ with $\|u_p(t)\| \leq v_p$, where $v_p > v_e$ is the maximum speed of the pursuer. The system evolves as

$$\dot{r}_p = u_p,$$
$$\dot{r}_e = u_e.$$

In our problem, the goal of the pursuer is to capture the evader. We define *capture* of the evader as the instance when the pursuer is within some pre-defined positive capture radius ε of the evader. Assuming that the pursuer has exact information about the evader's state at all times, it is well known that the time-optimal strategy for the pursuer is to move with maximum speed in the direction of the evader [4]. Such a strategy, known as classical pursuit, is given by the control law

$$u_p(t) = \frac{v_p}{\|r_e(t) - r_p(t)\|} \left(r_e(t) - r_p(t)\right). \qquad (1)$$

The issue with the control law (1) is that it requires continuous access to the evader's state at all times and instantaneous updates of the control input. Instead, we want to guarantee capture of the evader without tracking it at all times and only updating the controller sporadically. We do this by having the pursuer decide in an opportunistic fashion when to sample evader's position, and update its control input. Under this framework, the pursuer only knows the position of the evader at the time of its last observation. Let $\{t_k\}_{k \in \mathbb{Z}_{\geq 0}} \subset \mathbb{R}_{\geq 0}$ be a sequence of times at which the pursuer receives information about the evader's position. In between updates, the pursuer implements a zero-order hold of the control signal computed at the last time of observation using (1) which is given by

$$u_p(t) = \frac{v_p}{\|r_e(t_k) - r_p(t_k)\|} \left(r_e(t_k) - r_p(t_k)\right), \qquad (2)$$

for $t \in [t_k, t_{k+1})$.

In this paper, our purpose is to identify a function for the self-triggered update duration ϕ for the pursuer that determines the next time at which the updated information is required. In other words, each time the pursuer receives updated information about the evader at some time t_k, we want to find the duration $\phi\left(D_k, v_e, v_p\right)$ until the next update such that

$$t_{k+1} = t_k + \phi\left(D_k, v_e, v_p\right), \qquad (3)$$

where $D_k \triangleq \|r_e(t_k) - r_p(t_k)\|$ is the separation between the agents at time t_k. Our goal is to design the triggering function ϕ such that the pursuer is guaranteed to capture the evader while also being aware of the number of samples of the evader required.

III. Design of Self-triggered Update Law

We study the pursuit and evasion problem consisting of a single pursuer and a single evader on a plane ($\mathbb{R}^2$), where both agents are modelled as single integrators with constant speeds[1]. Let $r_p = (x_p, y_p)$ and $r_e = (x_e, y_e)$. Additionally, the pursuer is moving along θ_p and the relative angle between the agents' headings is denoted by θ_e (see Fig. 1). Without loss of generality, we normalize the speed of the pursuer to $v_p = 1$ and the evader moves with a constant positive speed of $v_e = \nu < 1$. The dynamics of the pursuer and the evader are given by

$$\begin{aligned} \dot{x}_p &= \cos\theta_p, \quad \dot{x}_e = \nu\cos(\theta_e + \theta_p), \\ \dot{y}_p &= \sin\theta_p, \quad \dot{y}_e = \nu\sin(\theta_e + \theta_p). \end{aligned} \qquad (4)$$

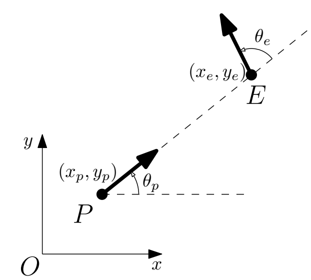

Fig. 1. Figure shows the pursuer P at $r_p = (x_p, y_p)$ and the evader E at $r_e = (x_e, y_e)$ in $\mathbb{R}^2$. The pursuer is moving along θ_p and the relative angle between agents' headings is denoted by θ_e. The arrows indicate the velocity vectors.

A. Self-triggered Update Policy for Pursuer

Suppose at time t_k the pursuer observes the evader a distance D_k away and starts moving towards it. The pursuer does not have access to the evader's evasion strategy. We are interested in the duration for which the pursuer can maintain its course of trajectory without observing the evader. More specifically, we are interested in the first instance at which the separation between the agents can possibly increase, thus prompting the pursuer to sample the evader's position and update its trajectory. Let $r(t)$ denote the separation between the pursuer and the evader at time t. We denote the one half times the square of the separation by R, which is given by

$$R = \frac{1}{2}r^2 = \frac{(x_e - x_p)^2 + (y_e - y_p)^2}{2}.$$

[1]It is not necessary to assume that evader is moving with constant speed at all times. We can bound the evader speed by $v_e^{\max}$ such that $v_e^{\max} < v_p$ and the analysis will remain unchanged.

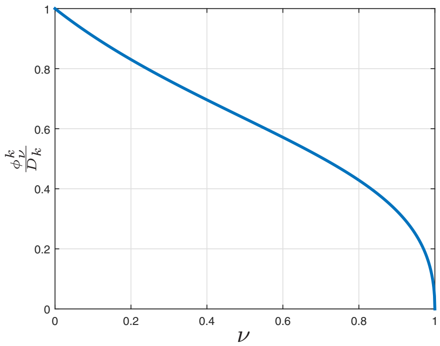

Fig. 2. Figure shows a plot of normalized update time $\frac{\phi_\nu^k}{D_k}$ against evader speed $\nu \in [0, 1)$, where ϕ_ν^k is given by (6).

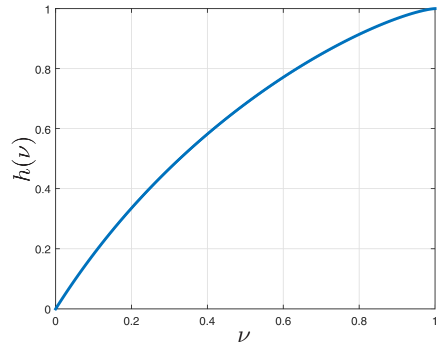

Fig. 3. Figure shows the graph of $h(\nu)$ against evader speed $\nu \in [0, 1)$, as given by the expression (7). It shows $h(\nu)$ is monotonically increasing with ν which intuitively means that greater evader speeds result in potentially smaller decrease in separation between updates.

Note that the time at which $\dot{R}$ becomes nonnegative is same as the time at which $\dot{r}$ becomes nonnegative. Using (4), the derivative of R (see Appendix for details) is given by

$$\dot{R}\left(\tau, x_e, y_e, \theta_e\right) = \nu(x_e - \tau)\cos\theta_e + \nu y_e \sin\theta_e + \tau - x_e, \tag{5}$$

where $\tau \in [0, t_{k+1} - t_k)$. We are interested in the first instance at which the separation between the agents can possibly increase. As the pursuer does not have access to evader's dynamics, we need to maximize $\dot{R}$ in (5) over the evader parameters (x_e, y_e, θ_e) subject to reachable set of the evader.

For fixed τ, we denote $\dot{R}$ in (5) by $\dot{R}_\tau(x_e, y_e, \theta_e)$. Let $g(\tau) = \sup_{x_e, y_e, \theta_e} \dot{R}_\tau(x_e, y_e, \theta_e)$, subject to the reachable set of the evader at τ. For the dynamics in (4), where $v_p = 1$ and $v_e = \nu$, we denote update duration in (3) by $\phi_\nu^k \triangleq \phi(D_k, v_p, v_e)$ and it is defined as

$$\phi_\nu^k = \inf\left\{\tau \in \mathbb{R}_{>0} | g(\tau) = 0\right\}.$$

Note that, for fixed evader speed ν, ϕ_ν^k is a function of D_k (as we have normalized v_p to 1). For notational brevity, we will drop the argument of ϕ_ν^k whenever it is clear from the context. For the agents modelled by (4), our self-triggered update duration is obtained by solving $g(\tau) = 0$ (see Appendix for derivation) and is given by

$$\phi_\nu^k = \begin{cases} \frac{D_k \nu\sqrt{1-\nu^2} - D_k(1-\nu^2)}{2\nu^2 - 1}, & \text{if } \nu \neq \frac{1}{\sqrt{2}} \\ \frac{D_k}{2}, & \text{if } \nu = \frac{1}{\sqrt{2}} \end{cases}. \tag{6}$$

The graph of normalized update time $\frac{\phi_\nu^k}{D_k}$ against evader speed ν is shown in Fig. 2. From the plot, we observe that increasing the evader speed decreases the self-triggered update duration for the pursuer. So, if the evader moves faster, our law prescribes more frequent updates of it to guarantee capture.

Recall that the primary objective of the pursuer is to *capture* the evader. The underlying principle in the design of the self-triggered policy is that at each instance of fresh observation the separation between the agents must have decreased. The following proposition characterizes this result.

Proposition III.1 (Decreasing separation between updates) *Let the pursuer and evader dynamics be given by* (4)*, where the agents are separated by D_k at time t_k. If the pursuer updates its trajectory using the self-triggered update policy ϕ_ν^k in* (6)*, then the distance between the agents at time t_{k+1} has strictly decreased, i.e. $D_{k+1} < D_k$ for $t_{k+1} = t_k + \phi_\nu^k$.*

Proof: Given the separation D_k at time t_k, the new separation between the agents, after a duration of ϕ_ν^k, is D_{k+1}. To see that the separation is strictly decreasing, note that

$$D_{k+1} \leq D_{\max}^k = D_k - (1-\nu)\phi_\nu^k \triangleq D_k h(\nu),$$

where $D_{\max}^k$ is the maximum possible separation between the agents after the duration ϕ_ν^k (see Appendix for details) and $h(\nu)$ is given by

$$h(\nu) = 1 - \frac{\nu(1-\nu)\sqrt{1-\nu^2} - (1-\nu)(1-\nu^2)}{2\nu^2 - 1}. \tag{7}$$

Note that $h(\nu) \in [0, 1)$ for $\nu \in [0, 1)$ (see Fig. 3). Thus, for any evader speed $\nu \in [0, 1)$, we have $D_{k+1} < D_k$ and the proof is complete. ■

B. Capture Time & Number of Samples

Using the self-triggered update policy in (6) and some pre-defined positive capture radius $\varepsilon < D_0$, the pursuer is guaranteed capture in finite time with finite updates. More specifically, we can find the maximum number of samples in terms of capture radius ε and evader speed ν and use it to guarantee finite time-to-capture. This is summarized in the following theorem.

Theorem III.2 (Guaranteed capture with finite updates) *Let the pursuer and evader dynamics be given by* (4)*, where the agents are initially separated by D_0. Given some pre-defined positive capture radius $\varepsilon < D_0$, the self-triggered update policy ϕ_ν^k in* (6) *ensures capture with finite observations in finite time.*

Proof: According to Proposition III.1, the separation between the agents is strictly decreasing between successive updates. In fact, the new separation between the agents satisfies the condition $D_{k+1} \leq D_k h(\nu)$, where $h(\nu) \in [0, 1)$ for $\nu \in [0, 1)$ as described by the expression (7). This implies that after n observations of the evader, the separation between the agents satisfies the inequality

$$D_n \leq D_0 h^n(\nu), \tag{8}$$

where D_0 is the initial separation between the agents. Using the inequality in (8), the maximum number of samples can be calculated by setting $D_0 h^n(\nu) \leq \varepsilon$. Thus,

$$n_{\max} = \left\lceil \frac{\log\left(\frac{\varepsilon}{D_0}\right)}{\log\left(h(\nu)\right)} \right\rceil. \tag{9}$$

The expression in (9) shows that for any pre-defined positive capture radius $\varepsilon < D_0$, the pursuer is guaranteed to capture the evader with finite number of samples. This completes the first part of the proof.

For self-triggered pursuit policy ϕ_ν^k in (6), the sequence of times at which the pursuer samples the evader position, denoted by $\{t_k\}_{k \in \mathbb{Z}_{\geq 0}}$, follows the criteria $t_{k+1} = t_k + \phi_\nu^k$. This means that after N updates, the total duration of pursuit (denoted by T_N) is given by

$$T_N = t_0 + \sum_{k=0}^{N} \phi_\nu^k. \tag{10}$$

Without loss of generality, we can assume $t_0 = 0$. Since the pursuer is guaranteed to capture the evader with finite number of maximum samples $n_{\max}$, the time-to-capture (denoted by T_{cap}) is bounded by

$$T_{\text{cap}} \leq \sum_{k=0}^{n_{\max}} \phi_\nu^k = f(\nu) \sum_{k=0}^{n_{\max}} D_k,$$

where $f(\nu)$ is given by

$$f(\nu) \triangleq \frac{\phi_\nu^k}{D_k} = \frac{\nu\sqrt{1-\nu^2} - (1-\nu^2)}{2\nu^2 - 1} \tag{11}$$

and satisfies the relationship $f(\nu) = \frac{1-h(\nu)}{1-\nu}$. Thus, we have

$$\begin{aligned} T_{\text{cap}} \leq f(\nu) \sum_{k=0}^{n_{\max}} D_k &\leq D_0 f(\nu) \sum_{k=0}^{n_{\max}} h^k(\nu) \\ &< D_0 f(\nu) \sum_{k=0}^{\infty} h^k(\nu) \\ &= \frac{D_0 f(\nu)}{1-h(\nu)} = \frac{D_0}{1-\nu}. \end{aligned}$$

It shows that, for any evader speed $\nu \in [0, 1)$, T_{cap} is strictly less than $\frac{D_0}{1-\nu}$ which is finite. This completes the proof. ■

Remark III.3 Note that in proving finite time-to-capture in Theorem III.2, we showed that time-to-capture is strictly less than $\frac{D_0}{1-\nu}$. However, given a pre-defined capture radius $\varepsilon < D_0$, it can be shown that the time-to-capture satisfies the inequality

$$T_{\text{cap}} \leq \frac{D_0 - \varepsilon}{1-\nu}, \tag{12}$$

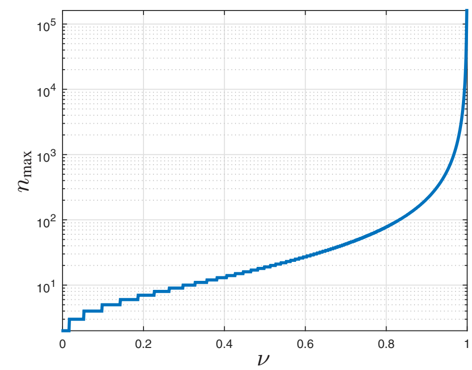

Fig. 4. Figure shows the variation of the maximum number of samples ($n_{\max}$) against evader speed as given by the expression in (9) where ε is chosen as $\frac{D_0}{10^3}$.

if the evader is actively moving away from the pursuer at all times. This is the same upper bound for time-to-capture in classical pursuit strategy. Recall that in classical pursuit, the pursuer is sensing the evader at all times and heading towards it with maximum possible speed v_p. The worst-case scenario is that the evader is moving directly away from the pursuer with its maximum possible speed v_e, at all times (classical evasion). Given that the agents are initially separated by D_0 along with a pre-defined capture radius $\varepsilon < D_0$, the time-to-capture is bounded by

$$T_{\text{cap}} \leq \frac{D_0 - \varepsilon}{v_p - v_e}, \tag{13}$$

where $v_p > v_e$. Note that $\frac{D_0-\varepsilon}{v_p-v_e}$ is an upper bound on total capture time, since any deviation from the classical evasion strategy on the part of the evader will allow capture in shorter time [4]. Therefore, by using self-triggered pursuit policy, we guarantee capture with finite number of updates without incurring any increase in the maximum time-to-capture. •

The expression in (9) guarantees capture with finite samples of evader's state. Fig. 4 shows a graph between the maximum number of samples that guarantee capture and the evader speed for pre-defined capture radius $\varepsilon = \frac{D_0}{10^3}$. The number of samples increase quite sharply as ν approaches 1. This makes intuitive sense as the maximum number of evader observations should increase as evader approaches the maximum speed of the pursuer.

IV. Trade-off Analysis

In Section III, the underlying principle in the design of self-triggered pursuit policy was to avoid increase in separation between the agents. The self-triggered update duration in (6) was defined as the first instance when the derivative of the separation becomes nonnegative. Instead, we want to model 'greedy' pursuit strategies that ensure decrease in separation by some amount between successive updates. In this section, we are interested in studying the trade-off between number of samples and the time-to-capture as a function of parameterized rate of decrease in separation. For each update step, we introduce a nonnegative parameter δ_k in order to parameterize the rate of decrease in separation between the agents and modify the design of self-triggered policy to obtain

$$\phi^k_{\delta_k,\nu} \triangleq \phi^k_\nu(D_k, \delta_k) = \inf\left\{\tau \in \mathbb{R}_{>0} | g(\tau) = -\delta_k\right\}, \quad (14)$$

where $g(\tau) = \sup_{x_e, y_e, \theta_e} \dot{R}_\tau(x_e, y_e, \theta_e)$. Solving $g(\tau) = -\delta_k$ results in

$$\phi^k_{\delta_k,\nu} = \frac{D_k\left(\frac{\delta_k}{D_k} - (1-\nu^2) \pm \sqrt{\nu^2(1-\nu^2) + (\frac{\delta_k}{D_k})^2 - \frac{2\nu^2\delta_k}{D_k}}\right)}{2\nu^2 - 1}. \quad (15)$$

Note that $\phi^k_{\delta_k,\nu}$ in (15) depends on the term $\frac{\delta_k}{D_k}$. We slightly modify our parameter by setting $\delta_k := \alpha D_k$, where $\alpha \in \mathbb{R}_{\geq 0}$. Note that setting δ_k to αD_k is one possible design choice of the parameter which helps us find a closed-form expression for update duration and study the trade-off between number of samples and the time-to-capture as a function of the parameter α. This yields

$$\phi^k_{\alpha,\nu} = \frac{D_k\left(\alpha - (1-\nu^2) \pm \sqrt{\nu^2(1-\nu^2) + \alpha^2 - 2\alpha\nu^2}\right)}{2\nu^2 - 1},$$

where, with a slight abuse of notation, we have used $\phi^k_{\alpha,\nu}$ instead of $\phi^k_\nu(D_k, \alpha) = \inf\{\tau \in \mathbb{R}_{>0} | g(\tau) = -\alpha D_k\}$. To find the permissible domain of α, we invoke the criteria $\phi^k_{\alpha,\nu} \in \mathbb{R}_{>0}$ for $\nu \in [0, 1)$. This yields $\alpha \in [0, \frac{1-\nu^2}{2})$.
As a result, the parameterized self-triggered update duration is given by

$$\phi^k_{\alpha,\nu} = \begin{cases} \frac{D_k\left(\alpha + \sqrt{\nu^2(1-\nu^2)+\alpha^2-2\alpha\nu^2} - (1-\nu^2)\right)}{2\nu^2-1}, & \text{if } \nu \neq \frac{1}{\sqrt{2}} \\ \frac{D_k(1-4\alpha)}{2(1-2\alpha)}, & \text{if } \nu = \frac{1}{\sqrt{2}} \end{cases}, \quad (16)$$

for $\alpha \in [0, \frac{1-\nu^2}{2})$ and $\nu \in [0, 1)$. Note that by choosing a fixed α, the rate at which the separation between the agents decreases is not constant as δ_k is proportional to the last observed separation D_k. This means that as the separation between the agents decreases, δ_k also decreases. However, we are guaranteed some amount of decrease in separation between successive updates.
Using similar analysis, as in Section III, it is straightforward to verify that by using parameterized self-triggered update duration $\phi^k_{\alpha,\nu}$, the separation between agents is strictly decreasing. We denote the new separation between the agents, after a duration of $\phi^k_{\alpha,\nu}$, by D_{k+1} which satisfies the inequality

$$D_{k+1} \leq D_k - (1-\nu)\phi^k_{\alpha,\nu} \triangleq D_k h_\alpha(\nu).$$

Here $h_\alpha(\nu)$[2] $\in [0, 1)$ for $\nu \in [0, 1)$ and $\alpha \in [0, \frac{1-\nu^2}{2})$. This shows that $D_{k+1} < D_k$ for $\nu \in [0, 1)$. Similarly, given an initial separation of D_0 between agents and a pre-defined positive capture radius $\varepsilon < D_0$, the maximum number of evader updates can be calculated by setting $D_0 \left(h_\alpha(\nu)\right)^n \leq \varepsilon$, which results in

$$n^\alpha_{\max} = \left\lceil \frac{\log\left(\frac{\varepsilon}{D_0}\right)}{\log\left(h_\alpha(\nu)\right)} \right\rceil. \quad (17)$$

[2] $h_\alpha(\nu) = 1 - \frac{(1-\nu)\left(\alpha + \sqrt{\nu^2(1-\nu^2)+\alpha^2-2\alpha\nu^2} - (1-\nu^2)\right)}{2\nu^2-1}$.

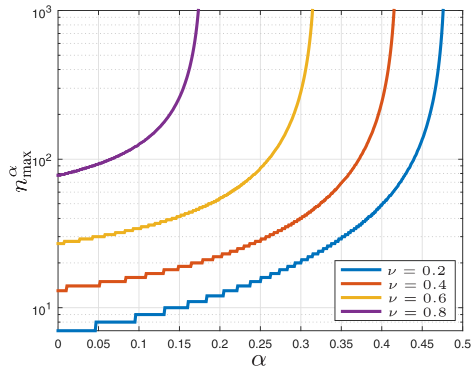

Fig. 5. Figure shows the variation of the maximum number of samples ($n^\alpha_{\max}$), as given by the expression (17), against α where $\alpha \in [0, \frac{1-\nu^2}{2})$ for different values of evader speed ν. Capture radius is chosen as $\varepsilon = \frac{D_0}{10^3}$.

For fixed evader speed ν, the number of samples for guaranteed capture *increases* as the parameter α is increased (see Fig. 5). This shows that, for any $\nu \in [0, 1)$, we need to sample the evader's state more frequently in order to ensure decrease in separation between the agents at a relatively higher rate. Thus, for fixed evader speed ν, $\alpha = 0$ corresponds to the *minimum class* of maximum number of updates that guarantee capture. In general, for fixed evader speed ν, α parameterizes a class of more greedy pursuit strategies that force more samples in order to capture the evader.

Remark IV.1 In Section III, we showed that the upper bound for time-to-capture is same as that of classical pursuit strategy. Since the worst-case behavior of the evader remains unchanged in our framework, the upper bound on time-to-capture is independent of the parameter α and remains the same. Note that although the upper bound on capture time remains unchanged, the actual time-to-capture depends on the evader trajectory. If the evader deviates from the classical evasive strategy, then it is possible to to *decrease* the actual capture time by increasing α. Thus, by introducing the parameter α, we can study the trade-off between the number of evader observations and the actual time-to-capture. •

V. Simulations

In this section, we provide simulations for our parameterized self-triggered pursuit policy as outlined in Section IV. We study the trade-off between the number of observations to catch the evader and the time-to-capture as we change the self-triggered update parameter α. For fixed evader speed ν, we showed in Section IV that we require $\alpha \in [0, \frac{1-\nu^2}{2})$. We model our agents according to the dynamics in (4), where the speed of the evader is less than that of pursuer. The evader is restricted to move in any of 4 directions: right, left, up and down and chooses the best direction to actively move away from the pursuer at all times. We initialize the agents with a separation $D_0 = 10$ units and set the capture radius as $\varepsilon = 10^{-2}$ units. Fig. 6 shows the variation of number of samples and Fig. 7 shows the time-to-capture T_{cap} for different evader speeds. From these results we can see that increasing the value α decreases the capture time at the cost of increased number of samples for any $\nu \in [0, 1)$, which agrees with our findings in Section IV.

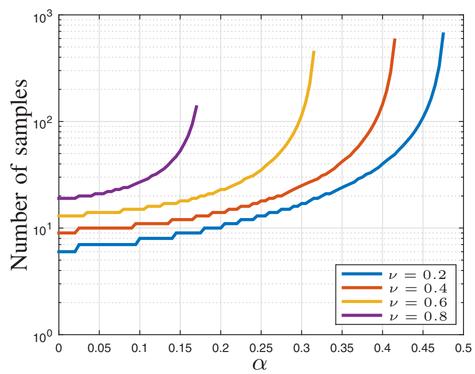

Fig. 6. Figure shows the variation of number of samples to capture the evader against $\alpha \in [0, \frac{1-\nu^2}{2})$ for different values of evader speed $\nu \in \{0.2, 0.4, 0.6, 0.8\}$. For any ν, the number of samples increase with increasing α.

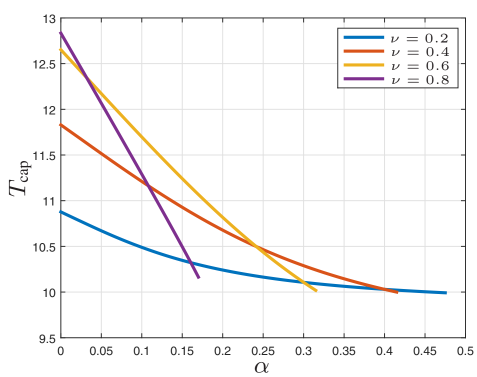

Fig. 7. Figure shows the variation of capture time T_{cap} against $\alpha \in [0, \frac{1-\nu^2}{2})$ for different values of evader speed $\nu \in \{0.2, 0.4, 0.6, 0.8\}$. For any ν, T_{cap} decreases with increasing α.

VI. Conclusions

The framework presented in this paper offers a novel perspective on various pursuit-evasion problems in the existing literature. We have proposed a self-triggered pursuit policy which allows the pursuer to autonomously decide when fresh information about the evader's location is required. This is in contrast to a majority of works that assume continuous, or at least periodic, information about the evader is available at all times. Our analysis guarantees that our worst-case time-to-capture is the same as that of classical pursuit strategies that assume continuous sensing of the evader, while only requiring sporadic updates about the evader's position. In addition to minimum updates to guarantee capture, we have developed a class of more greedy strategies that force more samples in order to capture the evader in less time. Our simulations have illustrated the theoretical results of the trade-off between number of samples and time-to-capture. In the future, we are interested in extending our methods in designing robust schemes for noisy observations, cooperative strategies involving multiple agents, and bounded sensing ranges.

References

[1] D. Pais and N. E. Leonard, "Pursuit and evasion: evolutionary dynamics and collective motion," in *AIAA Guidance, Navigation and Control Conference*, (Toronto, ON), pp. 1–14, 2010.

[2] F. Neuman, "On the approximate solution of complex combat games," *Journal of Guidance, Control, and Dynamics*, vol. 13, no. 1, pp. 128–136, 1990.

[3] T. H. Chung, G. A. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics - a survey," *Autonomous Robots*, vol. 31, no. 4, pp. 299–316, 2011.

[4] R. Isaacs, *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*. Courier Corporation, 1999.

[5] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, vol. 23. SIAM, 1999.

[6] Y. Ho, A. E. Bryson, and S. Baron, "Differential games and optimal pursuit-evasion strategies," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 385–389, 1965.

[7] S. H. Lim, T. Furukawa, G. Dissanayake, and H. F. Durrant-Whyte, "A time-optimal control strategy for pursuit-evasion games problems," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, (New Orleans, LA), pp. 3962–3967, 2004.

[8] J. Sgall, "Solution of David Gale's lion and man problem," *Theoretical Computer Science*, vol. 259, no. 1-2, pp. 663–670, 2001.

[9] L. Alonso, A. S. Goldstein, and E. M. Reingold, "Lion and man: upper and lower bounds," *ORSA Journal on Computing*, vol. 4, no. 4, pp. 447–452, 1992.

[10] S. Kopparty and C. V. Ravishankar, "A framework for pursuit evasion games in $\mathbb{R}^n$," *Information Processing Letters*, vol. 96, no. 3, pp. 114–122, 2005.

[11] J. L. Guibas, J. Latombe, S. M. LaValle, D. Lin, and R. Motwani, "Visibility-based pursuit-evasion in a polygonal environment," in *Algorithms and Data Structures*, pp. 17–30, Springer, 1997.

[12] S. LaValle and J. E. Hinrichsen, "Visibility-based pursuit-evasion: The case of curved environments," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, pp. 196–202, 2001.

[13] S. Sachs, S. M. LaValle, and S. Rajko, "Visibility-based pursuit-evasion in an unknown planar environment," *The International Journal of Robotics Research*, vol. 23, no. 1, pp. 3–26, 2004.

[14] V. Isler, S. Kannan, and S. Khanna, "Randomized pursuit-evasion in a polygonal environment," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 875–884, 2005.

[15] I. Suzuki and M. Yamashita, "Searching for a mobile intruder in a polygonal region," *SIAM Journal on Computing*, vol. 21, no. 5, pp. 863–888, 1992.

[16] B. P. Gerkey, S. Thrun, and G. Gordon, "Visibility-based pursuit-evasion with limited field of view," *The International Journal of Robotics Research*, vol. 25, no. 4, pp. 299–315, 2006.

[17] I. Isler, S. Kannan, and S. Khanna, "Randomized pursuit-evasion with local visibility," *SIAM Journal on Discrete Mathematics*, vol. 20, no. 1, pp. 26–41, 2006.

[18] S. D. Bopardikar, F. Bullo, and J. P. Hespanha, "Cooperative pursuit with sensing limitations," in *Proceedings of the IEEE American Control Conference*, (New York, NY), pp. 5394–5399, 2007.

[19] W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *Proceedings of the IEEE Conference on Decision and Control*, (Maui, HI), pp. 3270–3285, 2012.

[20] M. Velasco, J. Fuertes, and P. Marti, "The self triggered task model for real-time control systems," in *Proceedings of the 24^{th} IEEE Real-Time Systems Symposium*, vol. 384, pp. 67–70, 2003.
[21] A. Anta and P. Tabuada, "To sample or not to sample: Self-triggered control for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2030–2042, 2010.
[22] C. Nowzari and J. Cortés, "Self-triggered coordination of robotic networks for optimal deployment," *Automatica*, vol. 48, no. 6, pp. 1077–1087, 2012.
[23] C. Nowzari and J. Cortés, "Self-triggered and team-triggered control of networked cyber-physical systems," in *Event-Based Control and Signal Processing* (M. Miskowicz, ed.), Embedded Systems, 2015.
[24] D. V. Dimarogonas and K. H. Johansson, "Event-triggered control for multi-agent systems," in *Proceedings of the IEEE Conference on Decision and Control*, (Shangai, China), pp. 7131–7136, 2009.
[25] A. Eqtami, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Event-triggered control for discrete-time systems," in *Proceedings of the IEEE American Control Conference*, (Baltimore, MD), pp. 4719–4724, 2010.
[26] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2012.
[27] C. Nowzari and J. Cortés, "Distributed event-triggered coordination for average consensus on weight-balanced digraphs," 2014. Submitted.
[28] M. Mazo and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2456–2461, 2011.

APPENDIX

The agents are modelled by the dynamics in (4). Suppose, at time t_k, the pursuer observes the evader at a distance D_k. Let $r_p^k \triangleq r_p(t_k)$ and $r_e^k \triangleq r_e(t_k)$. Without loss of generality, we make the relative vector between pursuer and the evader parallel to the $x-$axis such that $y_p(t_k) = 0$ and $y_e(t_k) = 0$. Additionally, as a matter of convenience, we assume that $0 = x_p(t_k) < x_e(t_k) = D_k$. This is elaborated in Fig. 8.

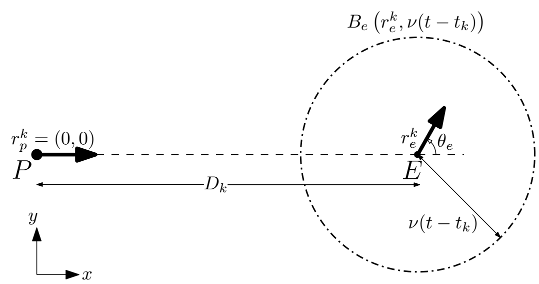

Fig. 8. Figure shows the pursuer and the evader, at $r_p^k = (0,0)$ and $r_e^k = (D_k, 0)$ respectively, separated by D_k at time t_k. $B_e\left(r_e^k, \nu(t-t_k)\right)$ is the ball centered at r_e^k with radius $\nu(t-t_k)$ and indicates the reachable set of the evader for $t \in [t_k, t_{k+1})$.

Thus, our pursuit trajectory is parallel to the $x-$axis as $\theta_p(t_k) = 0$. The pursuer does not observe the evader for the duration $t_{k+1} - t_k$, therefore it does not change its trajectory for $t \in [t_k, t_{k+1})$ and $\theta_p(t) = \theta_p(t_k) = 0$. The modified dynamics are given by

$$\begin{aligned} \dot{x}_p &= 1, \quad \dot{x}_e = \nu \cos\theta_e, \\ \dot{y}_p &= 0, \quad \dot{y}_e = \nu \sin\theta_e. \end{aligned} \tag{18}$$

Using (18), the position of the pursuer is given by $r_p(t) = (t - t_k, 0)$ for $t \in [t_k, t_{k+1})$. Recall that r denotes the separation and R denotes one half of square of separation between the agents. Thus, $\dot{R} = (x_e - x_p)(\dot{x}_e - \dot{x}_p) + (y_e - y_p)(\dot{y}_e - \dot{y}_p)$. As a result, between successive updates, $\dot{R}$ is given by

$$\dot{R} = \nu(x_e - t + t_k)\cos\theta_e + \nu y_e \sin\theta_e + t - t_k - x_e.$$

We introduce a change of variable by denoting $\tau \triangleq t - t_k$. In terms of τ, the derivative of R is given by

$$\dot{R}\left(\tau, x_e, y_e, \theta_e\right) = \nu(x_e - \tau)\cos\theta_e + \nu y_e \sin\theta_e + \tau - x_e. \tag{19}$$

The update time is given by the instance at which the separation begins to increase or the derivative of $\dot{R}$ becomes nonnegative. The reason for choosing square of separation is that it makes the analysis convenient. Note that the pursuer does not have access to evader dynamics. All the pursuer knows is that the reachable set of the evader is given by a ball of radius $\nu\tau$, centered at r_e^k, for any $\tau \in [0, t_{k+1} - t_k)$. Thus, $r_e \in B_e(r_e^k, \nu\tau)$. In (19), $\dot{R}$ is a function of evader parameters. For fixed τ, we denote $\dot{R}$ in (19) by $\dot{R}_\tau(x_e, y_e, \theta_e)$. We want to maximize $\dot{R}$, subject to the constraint $r_e \in B_e(r_e^k, \nu\tau)$. Let $g(\tau) = \sup_{x_e, y_e, \theta_e} \dot{R}_\tau(x_e, y_e, \theta_e)$, subject to the reachable set of the evader. Denoting the update duration in (3) by $\phi_\nu^k = t_{k+1} - t_k$, our self-triggered update duration is defined as

$$\phi_\nu^k = \inf\left\{\tau \in \mathbb{R}_{>0} | g(\tau) = 0\right\}.$$

In order to maximize $\dot{R}$ over evader parameters, we are interested in solving the following optimization problem

$$\begin{aligned} \sup_{x_e, y_e, \theta_e} \quad & \dot{R}_\tau(x_e, y_e, \theta_e), \\ \text{subject to} \quad & (x_e, y_e) \in B_e(r_e^k, \nu\tau). \end{aligned} \tag{20}$$

Note that, in $\mathbb{R}^2$, $B_e(r_e^k, \nu\tau)$ is given by $(x_e - D_k)^2 + y_e^2 \leq (\nu\tau)^2$. Thus the constraint of the problem (20) is independent of θ_e. In order to maximize the objective function in (20) with respect to θ_e, we perform unconstrained optimization by fixing x_e and y_e and solving $\frac{\partial \dot{R}_\tau}{\partial \theta_e} = 0$.

$$\frac{\partial \dot{R}_\tau}{\partial \theta_e} = -\nu(x_e - \tau)\sin\theta_e + \nu y_e \cos\theta_e. \tag{21}$$

Setting (21) to zero, we get

$$\theta_e^* = \arctan\left(\frac{y_e}{x_e - \tau}\right). \tag{22}$$

Note that for $\tau \in [0, \phi_\nu^k)$, $x_e - \tau \geq 0$. To see this, suppose $x_e - \tau < 0$. Setting $\theta_e = 0$ in the expression for $\dot{R}$ in (19), we get

$$\dot{R}_\tau = -(1 - \nu)(x_e - \tau) > 0,$$

for $\nu \in [0, 1)$. Thus, for $x_e - \tau < 0$, we can find a set of evader parameters (x_e, y_e, θ_e) for which $\dot{R}_\tau > 0$. This contradicts the definition of our self-triggered update duration as $\dot{R}_\tau \leq 0$ for any $\tau \in [0, \phi_\nu^k)$.

Due to symmetry of the problem, we can assume $y_e \geq 0$. Thus, from (22) and $x_e - \tau \geq 0$, we have $\theta_e^* \in [0, \frac{\pi}{2}]$. Substituting (22) in (19), we get

$$\dot{R}_\tau(x_e, y_e, \theta_e^*) = \nu\sqrt{y_e^2 + (x_e - \tau)^2} + \tau - x_e. \tag{23}$$

Using the result in (23), the problem in (20) simplifies to

$$\sup_{x_e,y_e} \quad \dot{R}_\tau(x_e,y_e) = \nu\sqrt{y_e^2+(x_e-\tau)^2}+\tau-x_e \tag{24}$$
$$\text{subject to} \quad (x_e,y_e)\in B_e(r_e^k,\nu\tau).$$

From the constraint $(x_e,y_e)\in B_e(r_e^k,\nu\tau)$ we get

$$y_e^2 \le (\nu\tau)^2-(x_e-D_k)^2. \tag{25}$$

In order to maximize the objective function in (24) with respect to y_e, we need y_e to be as large as possible. From the constraint (25), the objective function of (24) is maximized when y_e lies at the boundary of the ball $B_e(r_e^k,\nu\tau)$. Thus, for fixed x_e, y_e^* is given by

$$y_e^* = \sqrt{(\nu\tau)^2-(x_e-D_k)^2}. \tag{26}$$

Substituting (26) in (23), we get

$$\dot{R}_\tau(x_e) = \nu\sqrt{(x_e-\tau)^2+(\nu\tau)^2-(x_e-D_k)^2}+\tau-x_e. \tag{27}$$

This reduces the problem in (24) to

$$\sup_{x_e} \quad \dot{R}_\tau(x_e) \tag{28}$$
$$\text{subject to} \quad x_e\in[D_k-\nu\tau, D_k+\nu\tau].$$

Note that we can relax the problem in (28) by omitting the constraint $x_e\in[D_k-\nu\tau,D_k+\nu\tau]$. The relaxation of (28) results in an unconstrained optimization problem. Ignoring the constraints of problem (28), let $\widetilde{g}(\tau)\triangleq\sup_{x_e}\dot{R}_\tau(x_e)$ and $\widetilde{\phi}_\nu^k=\inf\{\tau\in\mathbb{R}_{>0}|\widetilde{g}(\tau)=0\}$. As a result of the relaxation of the problem (28), we have $\widetilde{\phi}_\nu^k\le\phi_\nu^k$ (this is because of omitting the constraint $x_e\in[D_k-\nu\tau,D_k+\nu\tau]$). Let $\widetilde{x}_e^*:=\operatorname{argmax}\dot{R}_\tau(x_e)$ for the unconstrained problem. To perform unconstrained maximization of $\dot{R}_\tau(x_e)$, we solve $\frac{\partial\dot{R}_\tau}{\partial x_e}=0$, where $\frac{\partial\dot{R}_\tau}{\partial x_e}$ is given by

$$\frac{\partial\dot{R}_\tau}{\partial x_e}=\frac{\nu(D_k-\tau)}{\sqrt{(x_e-\tau)^2+(\nu\tau)^2-(x_e-D_k)^2}}-1. \tag{29}$$

Setting (29) to zero, we get

$$\widetilde{x}_e^*=\frac{D_k^2\nu^2+D_k^2-2\tau D_k\nu^2-\tau^2}{2(D_k-\tau)}. \tag{30}$$

Substituting (30) in (27), we can evaluate $\widetilde{g}(\tau)$ which is given by

$$\widetilde{g}(\tau)=\tau-\frac{D_k^2\nu^2+D_k^2-2\tau D_k\nu^2-\tau^2}{2(D_k-\tau)}+\nu^2(D_k-\tau). \tag{31}$$

Solving for $\widetilde{g}(\tau)=0$, yields

$$\widetilde{\phi}_\nu^k=\begin{cases}\frac{D_k\nu\sqrt{1-\nu^2}-D_k(1-\nu^2)}{2\nu^2-1}, & \text{if } \nu\ne\frac{1}{\sqrt{2}}\\ \frac{D_k}{2}, & \text{if } \nu=\frac{1}{\sqrt{2}}\end{cases}. \tag{32}$$

Recall that $\widetilde{\phi}_\nu^k\le\phi_\nu^k$, where $\widetilde{\phi}_\nu^k$ was obtained from relaxing the constraint in problem (28) and $\phi_\nu^k=\inf\{\tau\in\mathbb{R}_{>0}|g(\tau)=0\}$. Our claim is that at the instance of update, the maximizer $\widetilde{x}_e^*$ of the relaxed problem is a feasible solution of the problem (28), i.e. $\widetilde{x}_e^*(\tau)\in[D_k-\nu\tau,D_k+\nu\tau]$ for $\tau=\widetilde{\phi}_\nu^k$. To see this, at $\tau=\widetilde{\phi}_\nu^k$, the maximizer $\widetilde{x}_e^*$ in (30), $D_k+\nu\widetilde{\phi}_\nu^k$ and $D_k-\nu\widetilde{\phi}_\nu^k$ are given by

$$D_k-\nu\widetilde{\phi}_\nu^k=D_k\varphi_1(\nu),$$
$$\widetilde{x}_e^*(\widetilde{\phi}_\nu^k)=D_k\varphi_2(\nu),$$
$$D_k+\nu\widetilde{\phi}_\nu^k=D_k\varphi_3(\nu),$$

where

$$\varphi_1(\nu)=1-\frac{\nu^2\sqrt{1-\nu^2}-\nu(1-\nu^2)}{2\nu^2-1},$$
$$\varphi_2(\nu)=\frac{3\nu^3+\sqrt{1-\nu^2}-2\nu-2\nu^4\sqrt{1-\nu^2}}{(\nu-\sqrt{1-\nu^2})(2\nu^2-1)},$$
$$\varphi_3(\nu)=1+\frac{\nu^2\sqrt{1-\nu^2}-\nu(1-\nu^2)}{2\nu^2-1}.$$

For $\nu\in[0,1)$, $\varphi_1(\nu)\le\varphi_2(\nu)\le\varphi_3(\nu)$. This shows that $\widetilde{x}_e^*(\widetilde{\phi}_\nu^k)$ satisfies the constraints in the problem (28) at $\tau=\widetilde{\phi}_\nu^k$. This means, at the instance of update, that the maximizer $\widetilde{x}_e^*$ of the relaxed problem is a feasible solution of the problem (28) and hence it is optimal solution x_e^* for (28). Thus $\widetilde{g}(\tau)=g(\tau)$ and as a result $\widetilde{\phi}_\nu^k=\phi_\nu^k$. Note that ϕ_ν^k is continuous in the parameter ν as $\lim_{\nu\to\frac{1}{\sqrt{2}}}\frac{D_k\nu\sqrt{1-\nu^2}-D_k(1-\nu^2)}{2\nu^2-1}=\frac{D_k}{2}$.

If the pursuer updates its trajectory using the self-triggered update policy described in (32), then the maximum distance between the agents, between successive updates, is given by

$$D_{\max}^k=D_k-(1-\nu)\phi_\nu^k.$$

To see this, after the duration ϕ_ν^k, the pursuer moves a distance of ϕ_ν^k units (as it is moving with unitary speed). The evader can be anywhere inside a ball of radius $\nu\phi_\nu^k$ centered at r_e^k (as it is moving with speed $\nu<1$). This is shown in Fig. 9. The maximum separation between the pursuer and the evader is denoted by $D_{\max}^k$ and is given by $D_k-(1-\nu)\phi_\nu^k$.

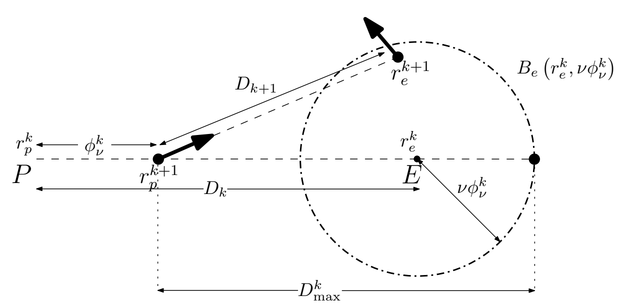

Fig. 9. At time t_k, the pursuer P is initially at r_p^k and the evader E is initially at r_e^k. After the duration ϕ_ν^k, the agents have moved to r_p^{k+1} and r_e^{k+1} respectively. D_{k+1} indicates the new separation between the agents. $B_e(r_e^k,\nu\phi_\nu^k)$ outlines the boundary of the reachable set of the evader after ϕ_ν^k. $D_{\max}^k$ denotes the maximum possible separation between the pursuer and the evader at $t=t_k+\phi_\nu^k$.