Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Self-Triggered Finite Time Pursuit Strategy for a Two-Player Game ⋆

Ameer K. Mulla *

* *Department of Electrical Engineering, Indian Institute of Technology Dharwad, Dharwad, Karnataka, India (e-mail: ameer@iitdh.ac.in)*

Abstract: A continuous-time two player pursuit-evasion game is considered. The players have double-integrator dynamics with bounded acceleration inputs. Unlike, conventional pursuit strategies, it is assumed that, the pursuer does not have continuous access to the states of the players. In this paper, we propose a self-triggered pursuit strategy, in which, the pursuer can choose when the state-information needs to be updated next. The proposed strategy is based on the time-optimal pursuit strategy for a game in which state information is available continuously to both the players. When the bound on acceleration of the evader is smaller than that of the pursuer, the proposed strategy guarantees capture in finite time, with finite number of information updates.

Keywords: Self-triggered Control, Control under Communication Constraints, Pursuit-Evasion Games, Game Theory, Networked Systems

1. INTRODUCTION

Not long after the game theory was first proposed in (Von Neumann and Morgenstern (1953)), Isaacs initiated the study of differential games (Iscaacs (1954–1956)). Pursuit–Evasion games form a widely studied class of differential games, starting from (Pontryagin (1962); Ho et al. (1965)). The initial research in such games was motivated by the theoretical interests (Basar and Olsder (1999)) and application to air combats (Isaacs (1999); Neuman (1990)). In the recent times, with the development of robotics and autonomous vehicles, the theory of pursuit–evasion games finds increasing applications in the areas like surveillance (Grocholsky et al. (2006); Dhillon and Chakrabarty (2003)), search and rescue missions (Chung et al. (2011)), urban security (Tokekar et al. (2014); Dames et al. (2017); Vinod et al. (2018)), wild-life monitoring (Dunbabin and Marques (2012); Tokekar et al. (2013)), and so on.

A pursuit–evasion game is a game of a kind, played between of two players, a *pursuer* and an *evader*. The objective of a pursuer is to *capture* the evader, who is aiming to escape. In the continuous time framework, the dynamics of the players are represented by differential equations (Ho et al. (1965)). Time optimal pursuit–evasion game is a game of degree where the pursuer tries to capture the evader in the shortest possible time, while the objective of the evader is to delay the capture as long as possible, or escape (Ho et al. (1965)).

There are many variations of pursuit-evasion games studied in the literature. Analytical solutions have been proposed for two player games with first-order and second order dynamics (Isaacs (1999); Pontryagin (1962)). Typically, the papers aim to either obtain a saddle point solution, which involves computing equilibrium strategies for both players (Isaacs (1999)) or derive pursuit strategies using worst-case analysis (Guibas et al. (1999); Mulla and Chakraborty (2018)). With development in the area of networked control, multi-player games have fetched attention in the recent years (Vinod et al. (2018); Oyler (2016)). There is a wide spectrum of pursuit–evasion games formulated in the context of various applications (see (Chung et al. (2011); Basar and Zaccour (2018)) for a review of recent work).

In this paper, a two player continuous time pursuit-evasion game is considered, with, each player modelled as a double integrator with bounded acceleration inputs. Unlike common assumption in the literature (Basar and Olsder (1999); Basar and Zaccour (2018)), the players do not have continuous information about the states or control actions of each other. Instead, the aim of this paper is to design a *need based feedback control* strategy for the pursuer, so that the pursuer can autonomously decide when to take a feedback of the states and update the control action, so as to capture the evader as quickly as possible.

The information structure plays a key role in the design of control strategies. Usually, the control strategies for pursuit–evasion game are derived assuming continuous information availability (Isaacs (1999); Basar and Olsder (1999)). Some of the papers like (Guibas et al. (1999); Mulla and Chakraborty (2018); Bopardikar et al. (2007)) consider visibility based information. The assumption of continuous communication, with or without sensing limitations, can be costly (e.g. in robotic systems, continuous communication consumes significant battery power, reducing total operation time), or sometimes unrealistic (continuous communication may be limited by the bandwidth or channel capacity) (Aleem et al. (2015b); Ding

⋆ This work is partially supported by Department of Science & Technology, Government of India, and Science & Engineering Research Board, Government of India.

et al. (2018)). Moreover, since the control is implemented using microprocessors which uses the periodically sampled data, sampling frequency is a compromise between the performance and communication and computational load (Anta and Tabuada (2010)). Event-triggered control algorithms are used as need based control algorithms to reduce the use of computation resource (Åström and Bernhardsson (1999)), which are quite robust, but need continuous monitoring of the states of the players. Self-triggered control algorithms (Heemels et al. (2012); Anta and Tabuada (2010)) are similar to the event triggered control, in the sense that, they also update the control signal based on occurrence of certain events. However, unlike event-triggered algorithms, they do not monitor the states continuously. The player estimates the time at which the next event may occur and sample only at those time instances.

Consistent event-triggered control and self-triggered control policies have been shown to perform better than periodic control in terms of communication overheads and performance for different applications like control of systems subjected to noise and disturbances (Antunes and Balaghi I. (2020)), networked control systems (Balaghi I. et al. (2019)), consensus algorithms for heterogeneous systems (Hu et al. (2017)) etc. A self-trigger pursuit algorithm is proposed in (Aleem et al. (2015a)), which considers the players with single integrator linear dynamics with fixed speeds.

In this paper, a self-triggered pursuit strategy is proposed for a pursuit-evasion game with players having double integrator dynamics with bounded accelerations. Designing pursuit strategies for systems with double integrator dynamics is more complicated compared to the single integrator case, since, for capture, not only the positions, but also the velocities of the players need to match. Due to the requirement of matching in both positions and velocities, capture cannot be guaranteed in finite time with periodic sampling, since, the players may miss the information that the capture has occurred and move away from each other, before next sampling instant. The main contribution of this paper is design of a self-triggered pursuit strategy, using the time-optimal pursuit strategy for a game with continuous state information (Mulla et al. (2014)). The proposed strategy guarantees capture in finite time with finite number of samples. It may be noted that, min-max time capture is possible if and only if continuous information is available to the pursuer for a specific duration time.

This paper is organised as follows: In Section 2, a brief review of time optimal pursuit strategy with continuous feedback is given, followed by the formulation of the problem considered in this paper. In Section 3, a self-triggered pursuit strategy with a state-based self-trigger function is proposed. The results are demonstrated through simulation examples in Section 4, followed by conclusion in Section 5.

2. PRELIMINARIES AND PROBLEM DEFINITION

Consider a two player pursuit-evasion game. The dynamics of the pursuer and the evader are, respectively, given by

$$\dot{\mathbf{x}}_p(t) = A\mathbf{x}_p(t) + bu_p(t); \text{ and}$$
$$\dot{\mathbf{x}}_e(t) = A\mathbf{x}_e(t) + bu_e(t) \tag{1}$$

where, $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $\mathbf{x}_p(t) = \begin{bmatrix} x_p(t) \\ \dot{x}_p(t) \end{bmatrix}$ and $\mathbf{x}_e(t) = \begin{bmatrix} x_e(t) \\ \dot{x}_e(t) \end{bmatrix}$. Here, $x_p(t) \in \mathbb{R}$ and $\dot{x}_p(t) \in \mathbb{R}$ denote the position and speed of the pursuer respectively. Similarly, $x_e(t) \in \mathbb{R}$ and $\dot{x}_e(t) \in \mathbb{R}$, respectively denote the position and speed of the evader. The initial conditions are $\mathbf{x}_p(0) = \mathbf{x}_{p0} = [x_{p0} \ \dot{x}_{p0}]^T$ and $\mathbf{x}_e(0) = \mathbf{x}_{e0} = [x_{e0} \ \dot{x}_{e0}]^T$. The acceleration inputs to the players, $u_p(t) \in \mathbb{R}$ and $u_e(t) \in \mathbb{R}$, are bounded as $|u_p| \leq \alpha$ and $|u_e| \leq \beta$, for some $\alpha > \beta > 0$. The game terminates when the pursuer *capture*s the evader. Note that, When players have double-integrator dynamics, for capture, it is not enough that the distance between them is close to zero, but their velocities also need to be close to each other. If the velocities of the players are not close to each other, the players will pass by each other again increasing the distance between them. Thus, the *capture* is defined as follows:

Definition 1. The evader is considered to be captured by the pursuer at time t_c, if $|x_p(t_c) - x_e(t_c)| \leq \varepsilon$, and $|\dot{x}_p(t_c) - \dot{x}_e(t_c)| \leq \delta$, for some predefined $\varepsilon, \delta > 0$. Such a time, t_c, is called *capture time*.

If $\epsilon = \delta = 0$, it is called a *perfect capture*.

In a time-optimal pursuit-evasion game, the objective of the pursuer is to capture the evader in the least possible time, while the evader aims to avoid the capture, or delay the capture for as long as possible. Under the assumption of continuous measurement of the states, the time-optimal strategies of the players are computed by considering the relative dynamics the players (Mulla et al. (2014)). The feedback control strategies of the players are summarised next for the sake of completeness.

2.1 Time Optimal Pursuit with Continuous Feedback (Mulla et al. (2014))

Using (1), the relative dynamics between the players is given by

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + bu(t) \tag{2}$$

where, $\mathbf{x}(t) = [x \ \dot{x}]^T = \mathbf{x}_p(t) - \mathbf{x}_e(t)$. The initial conditions translate to $\mathbf{x}(0) = \mathbf{x}_0 = [x_0 \ \dot{x}_0]^T = \mathbf{x}_{p0} - \mathbf{x}_{e0}$, and the control input, $u(t) = u_p(t) - u_e(t)$. The state trajectory of (2) is expressed as

$$\mathbf{x}(t) = e^{At} + \int_0^t e^{A(t-z)} bu(z)dz \tag{3}$$

The input, $u(t)$ can vary between $\pm(\alpha + \beta)$. However, because of the conflicting objectives of the players, the optimal strategies of the players are computed using $u(t)$ bounded between $\pm(\alpha - \beta)$ (Section 9.5 of Bryson and Ho (1975)). In the framework of relative dynamics, capture condition translates as $x(t_c) \in [-\varepsilon, \varepsilon]$ and $\dot{x}(t_c) \in [-\delta, \delta]$, more specifically, for perfect capture, $x(t_c) = \dot{x}(t_c) = 0$.

Lemma 2. Under the assumption of continuous state feedback, the optimal feedback strategies of the players are given by $u_p^f(t) = \frac{\alpha}{\alpha - \beta} u^*(t)$ and $u_e^f(t) = \frac{\beta}{\alpha - \beta} u^*(t)$.

Here, $u^*(t)$ is the feedback time-optimal control law for driving the states of (2) from the given initial conditions to the origin.

Lemma 3. The feedback time optimal control $u^*(t)$, is bang-bang, switching between the extreme values $\pm(\alpha-\beta)$, at most once according to the following:

$$u^*(\mathbf{x}) = -\text{sgn}(\dot{x})(\alpha-\beta), \quad \text{if } 2(\alpha-\beta)x + \text{sgn}(\dot{x})\dot{x}^2 = 0;$$
$$= -\text{sgn}(2(\alpha-\beta)x + \text{sgn}(\dot{x})\dot{x}^2)(\alpha-\beta), \text{ otherwise.} \quad (4)$$

When both the players use their time optimal strategies, then the capture time,

$$t_c^* = \frac{\pm\dot{x}_0 + \sqrt{2\dot{x}_0^2 \pm 4x_0(\alpha-\beta)}}{\alpha-\beta}. \quad (5)$$

The sign depends on the sign of $2(\alpha-\beta)x + \text{sgn}(\dot{x})\dot{x}^2$. If the pursuer sticks to its time optimal strategy u_p^f and the evader unilaterally deviates from u_e^f, then the capture time $t_c \leq t_c^*$.

2.2 Some Additional Notations

In this section, we introduce a few more notations used in this paper. Refer to Fig. 1 for reference.

When both the players use their time-optimal strategies, the effective input to (2) is u^*. For some initial condition $\mathbf{x}_0$, the state trajectory $\mathbf{x}(t)$, travels along the black dashed curve shown in Fig. 1, as long as $2(\alpha-\beta)x+\text{sgn}(\dot{x})\dot{x}^2 < 0$. As soon as $2(\alpha-\beta)x + \text{sgn}(\dot{x})\dot{x}^2 = 0$, the input $u^*(t)$ switches the sign and $\mathbf{x}(t)$ travels along the curve $\{2(\alpha-\beta)x + \text{sgn}(\dot{x})\dot{x}^2 = 0$. This curve is known as the *switching curve*. The set of points on this curve is denoted by

$$\mathcal{S} := \{\mathbf{x} : 2(\alpha-\beta)x + \text{sgn}(\dot{x})\dot{x}^2 = 0\}.$$

Based on the sign of $\dot{x}$, the switching curve can be divided in two parts, $\mathcal{S}^+$ and $\mathcal{S}^-$, both meeting at the origin.

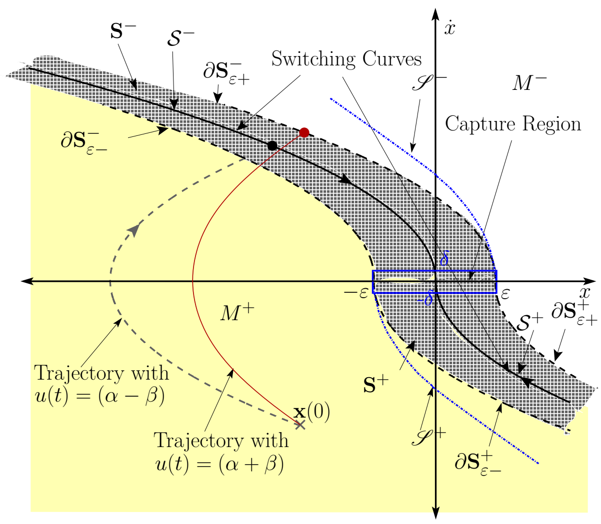

Fig. 1. State-space of the relative dynamics between the players

The switching curve divides the state-space $\mathbb{R}^2$ is two sets:

$$M^\pm := \{\mathbf{x} : u^*(\mathbf{x}) = \pm 1\} \quad (6)$$

When the condition for capture is relaxed from perfect capture to Definition 1, the sign of $u(t)$ may change anytime as long as $\mathbf{x} \in \mathbf{S}^+ \cup \mathbf{S}^-$, so that, for any initial condition $\mathbf{x}_0 \in \mathbb{R}^2$, the capture occurs using bang-bang input with at most one switch, irrespective of u_e. We refer $\mathbf{S} := \mathbf{S}^+ \cup \mathbf{S}^-$ as *switching region* and define $\mathbf{S}^\pm$ as:

$$\mathbf{S}^- := \{\mathbf{x} : \dot{x} \geq 0;\ 2(\alpha-\beta)(x+\epsilon) - (\dot{x}^2 - \delta^2) \geq 0 \text{ and}$$
$$2(\alpha-\beta)(x-\epsilon) + \dot{x}^2 \leq 0\},$$
$$\mathbf{S}^+ := \{\mathbf{x} : \dot{x} \leq 0;\ 2(\alpha-\beta)(x+\epsilon) - (\dot{x}^2 - \delta^2) \geq 0 \text{ and}$$
$$2(\alpha-\beta)(x-\epsilon) + (\dot{x}^2 - \delta^2) \leq 0\},$$

The boundary curves of $\mathbf{S}^\pm$ are

$$\partial\mathbf{S}^-_{\varepsilon-} := \{\mathbf{x} : 2(\alpha-\beta)(x+\epsilon) + (\dot{x}^2 - \delta^2) = 0\};$$
$$\partial\mathbf{S}^-_{\varepsilon+} := \{\mathbf{x} : 2(\alpha-\beta)(x-\epsilon) + \dot{x}^2 = 0\};$$
$$\partial\mathbf{S}^+_{\varepsilon-} := \{\mathbf{x} : 2(\alpha-\beta)(x+\epsilon) - \dot{x}^2 = 0\} \text{ and}$$
$$\partial\mathbf{S}^+_{\varepsilon+} := \{\mathbf{x} : 2(\alpha-\beta)(x-\epsilon) - (\dot{x}^2 - \delta^2) = 0\} \quad (7)$$

Observe, from Fig. 1, that, the capture region is completely enclosed in $\mathbf{S}$.

Two more curves, $\mathscr{S}^\pm$ are indicated in Fig. 1, which are defined as:

$$\mathscr{S}^- := \quad \{\mathbf{x} : 2(\alpha+\beta)(x-\epsilon) + (\dot{x})\dot{x}^2 = 0\} \text{ and}$$
$$\mathscr{S}^- := \quad \{\mathbf{x} : 2(\alpha-\beta)(x+\epsilon) - (\dot{x})\dot{x}^2 = 0\} \quad (8)$$

The context and significance of these curves will be evident in the context Section 3.1.

2.3 Problem Definition

In this paper, we relax the assumption of continuous feedback, and consider a different information structure, where the pursuer, can autonomously sample the states of the players, based on the necessity, so as to capture the evader. Let $t_k \in \mathbb{R}$, $k = 0, 1, 2, ...$ denote the k^{th} sampling instant with $t_0 = 0$ and $t_i > t_j$ for $i > j$. Clearly, the control action of the pursuer, $u_p(t)$ can be updated only when the information is updated. For any time, $t \in [t_k, t_{k+1})$, $u_p(t) = u_p(t_k)$.

We aim to identify a self-trigger function for the pursuer, that determines when it needs a subsequent updated state-information. Further, we devise a pursuit strategy, that updates its control action, u_p, whenever new information is available, so as to capture E.

Formally, the problem can be stated as follows:

Problem 4. For a pursuit-evasion game with the players having dynamics (1),

(1) Identify a function, $\tau(\mathbf{x}(t_k))$, such that,
$$t_{k+1} = t_k + \tau(\mathbf{x}(t_k)) \text{ for } k = 0, 1, 2, ...$$
(2) Design a pursuit strategy, such that, using $u_p(\mathbf{x}(t_k))$, the evader is captured in finite time with finite number of information updates.

3. SELF-TRIGGERED PURSUIT

The optimal strategies, u_p^f and u_e^f, described in Section 2.1, are the time-optimal control strategies of the players only when continuous state feedback is available. For different information structure, the optimal control strategies of the players may be different than $u_p^f(t)$ and $u_e^f(t)$.

3.1 Self-Triggered Pursuit Strategy

For designing a self-trigger function, we need to identify the events due to which the pursuer needs to update its control input so as to capture the evader.

Inspired by the feedback time-optimal pursuit strategy discussed in Lemmas 2 and 3, we propose the following self-triggered pursuit strategy:

$$\begin{aligned} u_p^s(t_k) &= u_p^f(\mathbf{x}(t_k)), \text{ for } k = 0, 1, 2, ... \\ u_p^s(t) &= u_p^s(t_k) \qquad \text{for } t \in [t_k, t_{k+1}) \end{aligned} \tag{9}$$

where, the sampling instants t_k are computed using self-trigger function $\tau(\mathbf{x}(t_k))$ as $t_{k+1} = t_k + \tau(\mathbf{x}_k)$ with $t_0 = 0$.

3.2 Self-Trigger Function

The pursuer triggers sampling of state-information, when it may be required to update $u_k(t)$ so as to ensure capture. Using the definitions given in Section 2.2, the state-space of (2) is divided in four regions. A self-trigger function is designed based on the region in which, the state lies.

Region 1. Enclosed between $\mathscr{S}^+$ and $\mathcal{S}^-$, i.e.

$\{\mathbf{x} : 2(\alpha + \beta)(x + \varepsilon) - \dot{x}^2 < 0 \ \& \ 2(\alpha - \beta)x + \dot{x}^2 < 0)\}$

It may be verified, using (9), that if $\mathbf{x}(t_k)$ lies in this region, $u_p^s(t_k) = \alpha$, irrespective of u_e. Thus, depending on the value of u_e, $u(t) \in [\alpha - \beta, \alpha + \beta]$. Further, $u_p^s(t)$ should change at some $t = t_{k+1}$, such that, $\mathbf{x}(t_{k+1}) \in \mathbf{S}$.

Observe that, $\mathbf{S}^-$ is the switching region shown as shaded area above x-axis in Fig. 1. Moreover, for any $\mathbf{x}(t_k)$ in Region 1, when $u_p(t) = \alpha$, irrespective of $u_e(t)$, the state trajectory $\mathbf{x}(t)$, $t \in [t_k, t_{k+1})$ lies between the two extreme trajectories with $u(t) = \alpha - \beta$ and $u(t) = \alpha + \beta$ (shown with black dashed curve and red curve respectively in Fig. 1). For $t \geq t_k$, based on $u_e(t)$, and hence on $u(t)$, the state trajectory $\mathbf{x}(t)$ travels through $\mathbf{S}^-$ at different times. The pursuer should change the input sign before $\mathbf{x}(t)$ leaves $\mathbf{S}^-$. Thus, we choose

$$t_{k+1} = \min_u \arg \max_t \mathbf{x}(t) \in \mathbf{S}^-$$

i.e., t_{k+1} is the shortest time in which $\mathbf{x}(t)$ leaves $\mathbf{S}^-$, when $u_p(t) = \alpha$, irrespective of $u_e(t)$. Note that, t_{k+1} is the time at which, the state trajectory, using constant input $u(t) = \alpha + \beta$ to intersect the farthest boundary of $\mathbf{S}^-$, i.e. $\partial\mathbf{S}^-_{\varepsilon+}$. Hence, for Region 1, the self-trigger function $\tau(\mathbf{x}(t_k)) := t_{k+1} - t_k$ is obtained by finding the intersection of curves $\mathbf{x}(t)$ with $u(t) = \alpha + \beta$ and $\partial\mathbf{S}^-_{\varepsilon+}$, and given by the following closed form expression,

$$\tau(\mathbf{x}(t_k)) = -\frac{\dot{x}(t_k)}{\alpha + \beta} + \frac{1}{\alpha + \beta}\sqrt{\frac{\alpha - \beta}{2\alpha}(\dot{x}(t_k)^2 - 2(\alpha + \beta)(x(t_k) - \varepsilon))}.$$

Region 2. Enclosed between $\mathcal{S}^-$ and $\mathscr{S}^-$, i.e.

$\{\mathbf{x} : 2(\alpha - \beta)x + \dot{x}^2 > 0) \ \& \ 2(\alpha + \beta)(x - \varepsilon) + \dot{x}^2 < 0\}$

When $\mathbf{x}(t_k)$ lies in Region 2, from (9), $u_p^s(t_k) = -\alpha$, irrespective of u_e. Thus, for $t \geq t_k$, the state trajectory $\mathbf{x}(t)$ either lies in Region 2 or enters Region 1. From (9), $u_p^s(t_k)$ needs to change sign before $\mathbf{x}(t)$ goes out of the switching region $\mathbf{S}^-$ in Region 1. Thus, the next sampling instance t_{k+1} is chosen as the *first* time instant at which the state trajectory $\mathbf{x}(t)$ may intersect the farthest boundary of $\mathbf{S}^-$, i.e. $\partial\mathbf{S}^-_{\varepsilon-}$, with $u_p(t) = -\alpha$. Thus, t_{k+1} may be computed using the intersections of the curves $\mathbf{x}(t)$ with $u(t) = -(\alpha + \beta)$ and $\partial\mathbf{S}^-_{\varepsilon-}$, which gives the self-trigger function $\tau(\mathbf{x}(t_k)) := t_{k+1} - t_k$ as

$$\tau(\mathbf{x}(t_k)) = \frac{\dot{x}(t_k)}{\alpha + \beta} - \frac{1}{\alpha + \beta}\sqrt{\frac{\beta - \alpha}{2\beta}(\dot{x}(t_k)^2 + 2(\alpha + \beta)(x(t_k) + \varepsilon)) + \frac{\alpha + \beta}{2\beta}\delta^2}$$

To avoid repetition, the self-trigger functions for the remaining regions are stated directly. The same may be computed using symmetry, and computation of the self-trigger function for Region 1 and Region 2 respectively.

Region 3. Enclosed between $\mathscr{S}^-$ and $\mathcal{S}^+$, i.e.

$\{\mathbf{x} : 2(\alpha + \beta)(x - \varepsilon) + \dot{x}^2 > 0 \ \& \ 2(\alpha - \beta)x - \dot{x}^2 > 0)\}$

$$\tau(\mathbf{x}(t_k)) = \frac{\dot{x}(t_k)}{\alpha + \beta} + \frac{1}{\alpha + \beta}\sqrt{\frac{\alpha - \beta}{2\alpha}(\dot{x}(t_k)^2 + 2(\alpha + \beta)(x(t_k) + \varepsilon))}.$$

Region 4. Enclosed between $\mathcal{S}^+$ and $\mathscr{S}^+$, i.e.

$\{\mathbf{x} : 2(\alpha - \beta)x - \dot{x}^2 > 0) \ \& \ 2(\alpha + \beta)(x - \varepsilon) + \dot{x}^2 > 0\}$

$$\tau(\mathbf{x}(t_k)) = -\frac{\dot{x}(t_k)}{\alpha + \beta} - \frac{1}{\alpha + \beta}\sqrt{\frac{\beta - \alpha}{2\beta}(\dot{x}(t_k)^2 - 2(\alpha + \beta)(x(t_k) - \varepsilon)) + \frac{\alpha + \beta}{2\beta}\delta^2}$$

Remark 5. For the states close to the capture region, for $\mathbf{x}(t_k)$ in Region 2 (or Region 4), if

$$\frac{\beta - \alpha}{2\beta}(\dot{x}(t_k)^2 - 2(\alpha + \beta)(x(t_k) - \varepsilon)) + \frac{\alpha + \beta}{2\beta}\delta^2 < 0$$

(or $\frac{\beta-\alpha}{2\beta}(\dot{x}(t_k)^2 - 2(\alpha + \beta)(x(t_k) - \varepsilon)) + \frac{\alpha+\beta}{2\beta}\delta^2 < 0$), then

$$\tau(\mathbf{x}(t_k)) := \frac{\dot{x}(t_k) + \delta}{\alpha + \beta}$$

(or $\tau(\mathbf{x}(t_k)) := \frac{-\dot{x}(t_k)+\delta}{\alpha+\beta}$).

3.3 Self-Triggered Finite Time Pursuit

Theorem 6. The self-triggered pursuit strategy proposed in (9), with the self-triggered function proposed in Section 3.1 above, achieves capture in finite time using only a finite number of information updates.

Proof: Consider the initial conditions in Region 1.

The theorem may be proved in two stages:

Stage 1: $\mathbf{x}(t_k) \notin \mathcal{S}^+$

When $\mathbf{x}(t_k)$ lies in Region 1 but, $\mathbf{x}(t_k) \notin \mathcal{S}^+$, $u_p^s(t) = \alpha$. Based on u_e, the input to the relative dynamics, $u \in [\alpha - \beta, \alpha + \beta]$ and hence $\mathbf{x}(t)$ follows some trajectory that lies between the red curve (corresponding to $u = \alpha + \beta$) and black-dashed curve (corresponding to $u = \alpha - \beta$), shown in Fig. 1. From (3), it may be verified that, for any such input u, the state trajectory evolves such that, for $t > t'$, $\dot{x}(t) > \dot{x}(t')$, in fact, as u_p^s is constant between t_k and t_{k+1},

$$\dot{x}(t_{k+1}) = \dot{x}(t_k) + u_p^s(t_k)\tau(\mathbf{x}(t_k)).$$

Further, it can also be verified that

$$\frac{d\tau(\mathbf{x})}{d\dot{x}} < 0.$$

Moreover, observe that, for some t_k, if $\mathbf{x}(t_k) \in \mathcal{S}^-$, then $\tau(\mathbf{x}(t_k)) = -\frac{\dot{x}(t_k)}{\alpha+\beta} + \frac{1}{\alpha+\beta}\sqrt{\dot{x}(t_k)^2 - \frac{\alpha^2-\beta^2}{\alpha}\varepsilon} > 0$, for $\varepsilon > 0$. Since there is a finite increment in the value of $\dot{x}$, during each interval between the sampling times, and the total change in $\dot{x}$ from the initial condition to the case when $\mathbf{x}$ is pushed in $\mathbf{S}^- \cap$ Region 2, is also finite, the number of samples triggered by the pursuer, are also finite.

Stage 2: $\mathbf{x}(t_k) \in \mathbf{S}^- \cap$ Region 2

In this case, according to (9), $u_p^s = -\alpha$, depending on u_e, $\mathbf{x}$ may continue to be in $\mathbf{S}^- \cap$ Region 2, or may be pushed back to Region 1. However, irrespective of u_e, note that, $\frac{dx}{dt} > 0$. Further, it can be verified that, for $\mathbf{x} \in \mathcal{S}^-$,

$$\frac{d\tau(\mathbf{x})}{dx} > 0.$$

That is, as $\mathbf{x}$ approaches the origin, the interval between the samples actually increases. Further, using Remark 5, it may be shown that, for $\delta > 0$, the number of information updates needed before capture is finite, and so is the capture time.

If the initial conditions lie in Region 2, the arguments identical to Stage 2 above, prove the theorem.

The theorem may be proved for the initial conditions in Region 3 and 4 using symmetry. ■

4. SIMULATIONS

Consider a pursuit-evasion game starting from the initial conditions of the players such that $\mathbf{x}(0) = \mathbf{x}_{p0} - \mathbf{x}_{e0} = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$. The capture is defined as in Definition 1, with $\varepsilon = 0.1$ and $\delta = 10^-3$. We demonstrate the performance of the self-triggered pursuit algorithm is three situations.

(1) The evader employs feedback worst-case strategy, such that, $u_e(t) = -\text{sgn}(2x(t) + \text{sgn}(\dot{x}(t))\dot{x}(t)^2)$. The trajectory of the difference between the states of the players, $\mathbf{x}(t)$ is demonstrated in Fig. 2. In this situation, $\mathbf{x}(0) \notin S$. Using the pursuit strategy (9), $\mathbf{x}(t)$ enters S in $t = 4$, the pursuer updates the information 10 times before $\mathbf{x}(t)$ enters S. Once $\mathbf{x}(t) \in S$, due to the control strategies of the players, $\mathbf{x}(t)$ never leaves S and the capture occurs at $t = 6$, which is the min-max time-to-capture with continuous feedback. The pursuer triggered 62 times in total for information updates before capture.

(2) With the proposed control strategy of the pursuer, a constant strategy of the evader $u_e(t) = 1$ for all t causes more delay in capture compared to the worst case feedback strategy. As demonstrated in Fig. 3, every time $\mathbf{x}(t) \in S$, the trajectory $\mathbf{x}(t)$ is driven maximally away from S till the next sampling instant. This increases the time-to-capture to 5.6388. In this case, the pursuer samples the states 81 times.

(3) The evader uses some random inputs $u_e(t)$ for all the time. In the instance demonstrated in Fig. 4, the pursuer captures the evader in $t = 3.6157$. Unlike previous two situations, $\mathbf{x}(t)$ enters S in $t = 1.5545$ with only four data updates. Once $\mathbf{x}(t) \in S$, based on the evader's control action, sometimes $\mathbf{x}(t)$ stays in S, while at some others instants, it is pushed out of S. The pursuer is triggered 56 times for sampling to update its information.

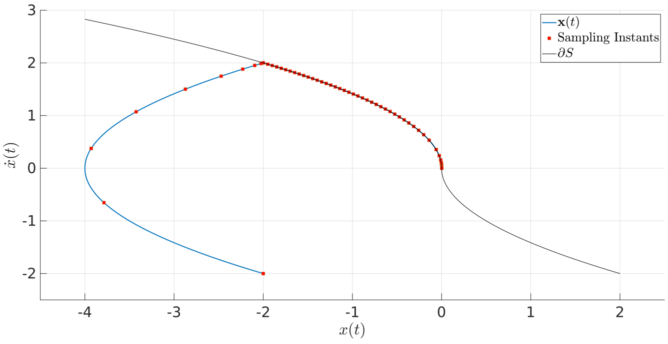

Fig. 2. $\mathbf{x}(t)$ when $u_e(t) = u_e^f(t)$.

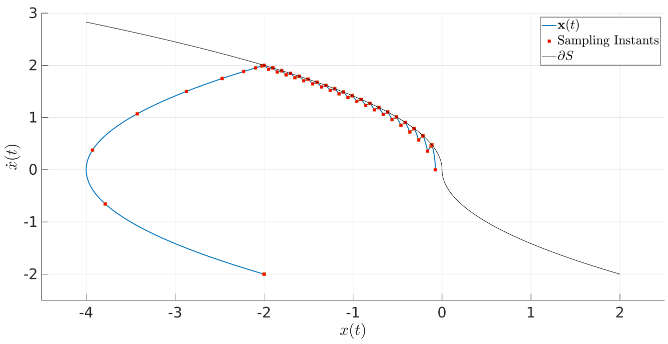

Fig. 3. $\mathbf{x}(t)$ when $u_e(t) = 1$.

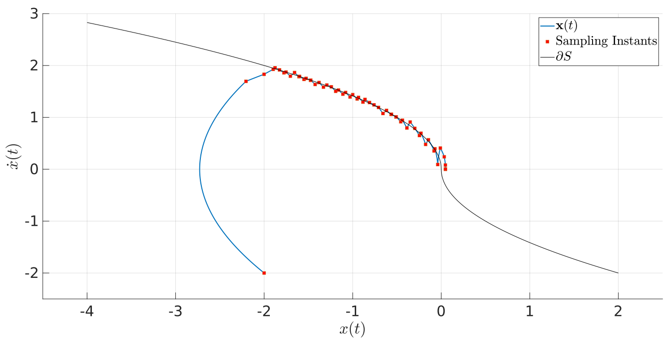

Fig. 4. $\mathbf{x}(t)$ when $u_e(t)$ is random.

Remark 7. In this simulation, a relatively high value of ε is chosen to demonstrate the behaviour of the state trajectory properly. For smaller values of ε, the pursuer is triggered more frequently for information and control action update.

5. CONCLUSION

A self-triggered pursuit policy is proposed so as to enable the pursuer to sample the information on need basis and reduce communication overhead. The proposed strategy guarantees capture in finite time, using only a finite number of information updates. It is conjectured that, the proposed self-triggered pursuit strategy is a min-max time pursuit strategy, under the given information structure. An immediate extension of this work may be to adapt the pursuit strategy to develop self-triggered consensus tracking problems, which is a work in progress. Extension of the proposed algorithm to multi-player games may also be considered.

REFERENCES

Aleem, S.A., Nowzari, C., and Pappas, G.J. (2015a). Self-triggered pursuit of a single evader. In *2015 54th IEEE Conference on Decision and Control (CDC)*, 1433–1440. IEEE.

Aleem, S.A., Nowzari, C., and Pappas, G.J. (2015b). Self-triggered pursuit of a single evader with uncertain information. *arXiv preprint arXiv:1512.06184*.

Anta, A. and Tabuada, P. (2010). To sample or not to sample: Self-triggered control for nonlinear systems. *IEEE Transactions on automatic control*, 55(9), 2030–2042.

Antunes, D.J. and Balaghi I., M.H. (2020). Consistent event-triggered control for discrete-time linear systems with partial state information. *IEEE Control Systems Letters*, 4(1), 181–186.

Åström, K.J. and Bernhardsson, B. (1999). Comparison of periodic and event based sampling for first-order stochastic systems. In *Proceedings of the 14th IFAC World congress*, volume 11, 301–306.

Balaghi I., M.H., Antunes, D.J., and Heemels, W.P.M.H.M. (2019). An l2-consistent data transmission sequence for linear systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2622–2627.

Basar, T. and Olsder, G.J. (1999). *Dynamic noncooperative game theory*, volume 23. SIAM.

Basar, T. and Zaccour, G. (2018). *Handbook of Dynamic Game Theory*. Springer.

Bopardikar, S.D., Bullo, F., and Hespanha, J.P. (2007). Cooperative pursuit with sensing limitations. In *2007 American Control Conference*, 5394–5399. IEEE.

Bryson, A. and Ho, Y.C. (1975). *Applied Optimal Control: Optimization, Estimation and Control*. CRC Press.

Chung, T.H., Hollinger, G.A., and Isler, V. (2011). Search and pursuit-evasion in mobile robotics. *Autonomous robots*, 31(4), 299.

Dames, P., Tokekar, P., and Kumar, V. (2017). Detecting, localizing, and tracking an unknown number of moving targets using a team of mobile robots. *The International Journal of Robotics Research*, 36(13-14), 1540–1553.

Dhillon, S.S. and Chakrabarty, K. (2003). Sensor placement for effective coverage and surveillance in distributed sensor networks. In *2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003.*, volume 3, 1609–1614. IEEE.

Ding, L., Han, Q.L., Ge, X., and Zhang, X.M. (2018). An overview of recent advances in event-triggered consensus of multiagent systems. *IEEE transactions on cybernetics*, 48(4), 1110–1123.

Dunbabin, M. and Marques, L. (2012). Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics & Automation Magazine*, 19(1), 24–39.

Grocholsky, B., Keller, J., Kumar, V., and Pappas, G. (2006). Cooperative air and ground surveillance. *IEEE Robotics & Automation Magazine*, 13(3), 16–25.

Guibas, L.J., Latombe, J.C., LaValle, S.M., Lin, D., and Motwani, R. (1999). A visibility-based pursuit-evasion problem. *International Journal of Computational Geometry & Applications*, 9(04n05), 471–493.

Heemels, W., Johansson, K.H., and Tabuada, P. (2012). An introduction to event-triggered and self-triggered control. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 3270–3285. IEEE.

Ho, Y., Bryson, A., and Baron, S. (1965). Differential games and optimal pursuit-evasion strategies. *IEEE Transactions on Automatic Control*, 10(4), 385–389.

Hu, W., Liu, L., and Feng, G. (2017). Output consensus of heterogeneous linear multi-agent systems by distributed event-triggered/self-triggered strategy. *IEEE Transactions on Cybernetics*, 47(8), 1914–1924.

Isaacs, R. (1999). *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization*. Courier Corporation.

Iscaacs, R. (1954–1956). Differential Games I, II, III, IV. *RAND Corporation Research Memorandum RM-1391, 1399, 1411, 1468.*

Mulla, A.K., Bhatt, S., Patil, D.U., and Chakraborty, D. (2014). Min-max time consensus tracking over directed trees. In *The 21st International Symposium on Mathematical Theory of Networks and Systems, Groningen, The Netherlands.*

Mulla, A.K. and Chakraborty, D. (2018). Min–max time consensus tracking with communication guarantee. *IEEE Transactions on Automatic Control*, 63(1), 132–144.

Neuman, F. (1990). On the approximate solution of complex combat games. *Journal of Guidance, Control, and Dynamics*, 13(1), 128–136.

Oyler, D.W. (2016). *Contributions To Pursuit-Evasion Game Theory*. Ph.D. thesis, University of Michigan.

Pontryagin, L.S. (1962). *Mathematical theory of optimal processes*. New York–Interscience.

Tokekar, P., Branson, E., Vander Hook, J., and Isler, V. (2013). Tracking aquatic invaders: Autonomous robots for monitoring invasive fish. *IEEE Robotics & Automation Magazine*, 20(3), 33–41.

Tokekar, P., Isler, V., and Franchi, A. (2014). Multi-target visual tracking with aerial robots. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3067–3072. IEEE.

Vinod, A.P., HomChaudhuri, B., Hintz, C., Parikh, A., Buerger, S.P., Oishi, M.M., Brunson, G., Ahmad, S., and Fierro, R. (2018). Multiple pursuer-based intercept via forward stochastic reachability. In *2018 Annual American Control Conference (ACC)*, 1559–1566. IEEE.

Von Neumann, J. and Morgenstern, O. (1953). *Theory of games and economic behavior*. Princeton Univ. Press.