Time-Optimal Motion Strategies for Capturing an Omnidirectional Evader using a Differential Drive Robot

Ubaldo Ruiz, Rafael Murrieta-Cid, *Member, IEEE* and Jose Luis Marroquin

***Abstract*—In this paper, we consider the problem of capturing an omnidirectional evader using a Differential Drive Robot (DDR) in an obstacle-free environment. At the beginning of this game the evader is at a distance $L > l$ (the capture distance) from the pursuer. The goal of the evader is to keep the pursuer farther than this capture distance as long as possible. The goal of the pursuer is to capture the evader as soon as possible. In this work, we make the following contributions: We present closed-form representations of the motion primitives and time-optimal strategies for each player; these strategies are in Nash Equilibrium, meaning that any unilateral deviation of each player from these strategies does not provide to such player benefit towards the goal of winning the game. We propose a partition of the playing space into mutually disjoint regions where the strategies of the players are well established. This partition is represented as a graph which exhibits properties that guarantee global optimality. We also analyze the decision problem of the game and we present the conditions defining the winner.**

***Index Terms*—Differential Games, Pursuit-Evasion, Capturing, Nonholonomic Constraints**

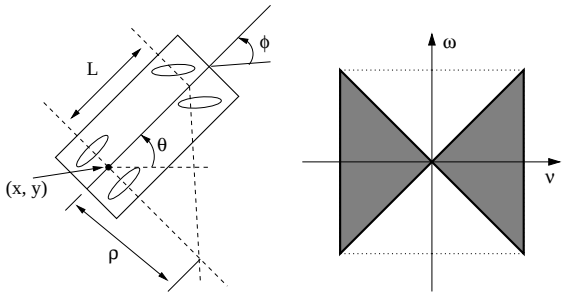
I. Introduction

This paper addresses a pursuit-evasion game. A great deal of previous research exists in the area of pursuit-evasion, particularly in the area of dynamics and control in free space (without obstacles) [1], [2]. The pursuit-evasion problem is often framed as a problem in non-cooperative dynamic game theory [2]. A pursuit-evasion game can be defined in several ways. For example, one or more pursuers could be given the task of *finding* an evader [3], [4], [5] in an environment with obstacles. To solve this problem [3], [5], [6], [7], [8], [9], the pursuer(s) must sweep the environment so that the evader is not able to eventually sneak into an area that has already been explored. A recent survey of this kind of problem is presented in [10]. Other related problem, which has been extensively studied, consists in *maintaining* visibility of a moving evader in an environment with obstacles [11], [12], [13], [14]. In [15] we have specifically considered the case in which both the pursuer and the evader *are omnidirectional*; that work led to a sufficient escape condition for the evader.

The problem that we address in this work is closely related to the classical differential game, called the homicidal chauffeur problem [1], [16]. In that game a faster pursuer (w.r.t. the evader) has as its objective to get closer than a given distance (the capture condition) from a slower but more agile evader, in order to run him over. The pursuer is a vehicle with a bounded minimal turning radius. The game takes place in the Euclidean plane without obstacles, and the evader aims to avoid the capture condition. In our problem, we also consider a faster pursuer and a more agile evader moving in an environment without obstacles. The game is defined in the same way: the objective of the pursuer is to capture the evader, while the evader aims to avoid the capture. However, there is an important difference between the problem described in this paper and the homicidal chauffeur one. In this work the pursuer is a Differential Drive Robot (DDR), i.e. the pursuer can rotate in place. Furthermore, the DDR can move backward and forward while in the Homicidal Chauffeur problem the pursuer can only move forward. In Subsection I-A we present a more detailed comparison between the Homicidal Chauffeur problem and the problem addressed in this paper.

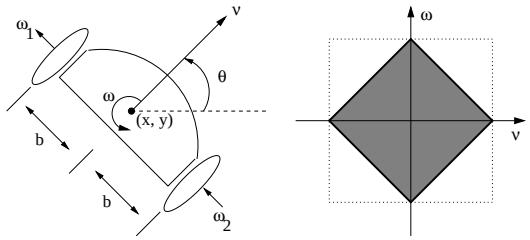
In [17] we have presented a solution for the problem of maintaining surveillance of an omnidirectional mobile evader at *constant* distance with a Differential Drive Robot, assuming that the state of the system (pose of the Differential Drive Robot and position of the evader) is known by both players and the instantaneous velocity vector of the evader is known by the pursuer. Note that if the instantaneous velocity of the evader is not known by the pursuer, it is not possible to maintain a constant distance between both players. In that work, we obtained optimal motion strategies, in the sense that they require the minimal speed for both players for winning, and the long term solution for the game. The main distinguishing features of our current work compared with our previous research in [17] are: the games are different, in this paper, the pursuer wants to reduce the distance between both players in minimum time while in [17] the pursuer wants to maintain a constant distance using minimal velocity. In this work, the criterion to be optimized is different; the proposed motion strategies for both players are time-optimal while in [17] they are not. In this case, the players only know their positions, i.e. the instantaneous velocity vector of the evader is not needed as in [17]. In [17], the solution was obtained using geometrical methods while in this case the solution is based on the theory of optimal control and differential games. An early version [18] of part of this work has appeared in the IEEE Intl. Conf. on Robotics and Automation 2012. In this document, we have extended the previous work in the following ways: We have included a comparison between the Homicidal Chauffeur problem and the game solved in this work. We stress the fact

An early version [18] of part of this work has been presented in the IEEE Intl. Conf. on Robotics and Automation 2012. U. Ruiz, R. Murrieta-Cid and J.L. Marroquin are with Centro de Investigación en Matemáticas, Guanajuato México (email: ubaldo@cimat.mx; murrieta@cimat.mx; jlm@cimat.mx;)



(a) Model (b) Control space

Fig. 1. Car-like



(a) Model (b) Control space

Fig. 2. Differential Drive Robot

that it is not possible to represent a DDR with the model for a simple car used in the Homicidal Chauffeur problem, and therefore, it is also not possible to obtain the solution for a DDR pursuer from the solution for the Homicidal Chauffeur problem. We propose a graph representation of the reduced state space partition which exhibits properties that guarantee global optimality. We explain the importance of this properties in the solution of the game. We show a trajectory followed by the system in the reduced space when the evader avoids capture. The trajectories for both players in the realistic space are also included. We have also included simulations where the players follow greedy (gradient descent) strategies and we compare the results against the optimal case.

A. Comparison between our contributions and the Homicidal Chauffeur problem solution

This section has as its goal to present three main differences between our contributions and the solution presented in [1] and [16] for the Homicidal Chauffeur problem: (1) In contrast to the solution proposed for the Homicidal Chauffeur problem, for our problem, we propose a graph representation of the reduced state space partition which exhibits properties that guarantee global optimality (refer to Subsection VIII-B). (2) We present a concrete example of a trajectory followed by the evader when it wins, i.e., it avoids capture indefinitely (refer to Section IX). (3) We show that the model for a simple car cannot be used to represent a DDR. In particular, we show that when the distance between the front and rear axles of the car tends to zero and the car steering angle tends to $\frac{\pi}{2}$, both implying that the turning radius tends to zero, then the car-like model approaches to an omnidirectional system and not a DDR.

Consider the kinematic model for a simple car in [19] shown in Fig. 1(a), (x, y, θ) is the configuration of the system, with the origin at the center of rear axle, and the x-axis pointing along the main axis of the car. Let v denote the translation velocity of the car, and ϕ denote the steering angle. Recall, that the distance between the front and rear axles is represented as L. If the steering angle is fixed at ϕ, the car travels in a circular motion, in which the radius of the circle is $\rho = L/\tan\phi$. Suppose that the translational velocity v and the steering angle ϕ are directly specified. The transition equation for a simple car is

$$\dot{x} = v\cos\theta,\ \ \dot{y} = v\sin\theta,\ \ \dot{\theta} = \frac{v}{L}\tan\phi \tag{1}$$

where $v \in [-v^{\max}, v^{\max}]$ and $\phi \in [-\pi/2, \pi/2]$. The last expression can be rewritten as $\dot{\theta} = (v/\rho)u_n$, where $u_n \in [-1, 1]$. This small variation of the transition equation, considering also that $v \in [0, v^{\max}]$, was used in [1], [16] to model the motion of the car in the Homicidal Chauffeur Problem. Assuming $\dot{\theta} = \omega$, in Fig. 1(b) we can observe the set of admissible controls in the $v - \omega$ space. From $\dot{\theta} = (v/\rho)u_n$, we have that as $\rho \to 0$ then $\dot{\theta} \to \infty$. Therefore, we have that as the turning radius ρ approaches zero, the simple car approaches an omnidirectional vehicle, which can instantaneously change its motion direction. In contrast, the kinematic model for a DDR [20] is given by (see Fig. 2(a)):

$$\dot{x} = v\cos\theta,\ \ \dot{y} = v\sin\theta,\ \ \dot{\theta} = \omega \tag{2}$$

where v is the robot's translational velocity and ω its angular velocity . With a suitable choice of units, we have that

$$v = \frac{\omega_1 + \omega_2}{2}, \quad \omega = \frac{\omega_2 - \omega_1}{2b} \tag{3}$$

where ω_1 and ω_2 are the wheel angular velocities. b is the distance between the center of the robot and the wheel location. For a DDR, assuming $v^{\max} > 0$, we have that

$$|\dot{\theta}| = |\omega| \leq \frac{1}{b}(v^{\max} - |v|) \tag{4}$$

The angular velocity is inversely proportional to the translation velocity. Comparing the spaces of admissible controls for both models (see Figs. 1(b) and 2(b)) we can see that there are values for v and w that can be valid for one model but not for the other. This leads to different time-optimal motion primitives. It is also important to note a fundamental assumption in the model for a simple car; its four wheels share the same rotational direction and speed, which is not the case for a DDR. Note that for the car-like robot it is possible to set *simultaneously* v and w at their maximal values (saturated values) while for the DDR is not possible to set at the same time v and w at their maximal values. Taking into consideration the arguments given above, we have that it is not possible to model a DDR using the model for a simple car, and therefore, it is also not possible to obtain the solution for a DDR pursuer from the solution for the Homicidal Chauffeur Problem.

B. Related work within robotics

Recent years have seen a growing interest in related problems within the robot motion planning community. In [11], game theory was proposed as a framework to formulate the tracking problem, and an on-line algorithm was presented. In [11], an algorithm was presented that operates by maximizing the probability of future visibility of the evader. The approach presented in [23] computes a motion strategy by maximizing the shortest distance to escape, i.e., the shortest distance the evader needs to move in order to escape the pursuer's visibility region. In [12], a technique is proposed to track an evader without the need of a global map. In [24], the problems of maintaining visibility and capturing an evader in an environment with obstacles are studied. In this case, both players are omnidirectional systems, and the authors show how to efficiently (low-degree polynomial time) compute an optimal path for the pursuer that counteracts *a given evader motion*. In [25], a robot has to track an unpredictable target with bounded speed. The robot's sensors are manipulated to record general information about the target's movements and avoid the necessity of detailed information about the target's position being available if the robot's sensors are accessed by other agent that can damage the target. Almost all existing work focuses on the 2-D version of the problem of maintaining visibility of an evader, but there are few works that deal with the 3-D version of it. In [26], the authors present an on-line algorithm for 3-D target tracking among obstacles, using local geometric information available to a robot's visual sensors. The robot motions are calculated minimizing a risk function. [14] addresses the problem of maintaining visibility of the evader in an environment containing obstacles. In [14], the authors prove the existence of strategies that are in Nash equilibrium: the pursuer wants to maintain visibility of the evader for the maximum possible amount of time, and, the evader wants to escape the pursuer's sight as soon as possible. But notice that in that work the pursuer is an *omnidirectional robot* while in the current paper the pursuer is an nonholonomic system (DDR). In fact, our problem is also related to the problem of finding optimal paths for nonholonomic robots [20], [21], [22]. The work in [14] and our current work are based on techniques from Optimal Control theory [27]. Only few pursuit-evasion problems modeled using optimal control tools can be solved analytically. For example, the Homicidal Chauffeur problem [1], [16], the tracking problem in [14] and some examples presented in [1]. In most of the cases, the use of numerical methods is necessary, particularly, in problems with state spaces of high dimensions (four or more). Several techniques have been proposed for those cases, some of them are based on the Level Set methods [28], and others in the Cell Mapping method [29], [30]. Typically, although those methods can obtain good results they have some limitations. The computational time required to find an accurate solution can be high, and the selection of the parameters, e.g., the resolution of the state space discretization, can be difficult.

Other techniques typically used for solving the type of problems addressed in the paper are: Numerical optimization of the Hamiltonian. See for instance [15]. This technique has as a drawback that in general, it does not guarantee global optimality unless the search space is convex. Other option is to use numerical dynamic programming (DP) [33], since this technique is exhaustive it guarantees global optimality. However the decision problem must be solved by other means previously to the use of DP, since DP finds the optimal trajectories backwards, starting from the terminal condition, which in our problem corresponds to the usable part. This technique has as an advantage that it can deal more easily with obstacles in the environment. In this work, we found that it is possible to obtain an analytical solution for the problem. We focus our attention in finding closed-form representations of the motion primitives and strategies for the players.

An interesting version of the problem of pursuit-evasion involves multiple participants of each kind (several evaders and pursuers) [31], [13], [32]. Pursuit-evasion has been found to be of use in interesting applications. For example, in [33], the authors noticed the similarity between pursuit-evasion games and mobile-routing for networking. Applying this similarity, they proposed motion planning algorithms for robotic routers to maintain connectivity between a mobile user and a base station. That work also includes a proof-of-concept implementation.

II. Problem formulation

A Differential Drive Robot (DDR), the pursuer, and an omnidirectional evader move on a plane without obstacles. The DDR tries to capture the evader. The game is over when the distance between the DDR and the evader is smaller than a critical value l. Both players have maximum bounded speeds $V_p^{\max}$ and $V_e^{\max}$, respectively. The DDR is faster than the evader, $V_p^{\max} > V_e^{\max}$, but it can only change its direction of motion at a rate that is inversely proportional to its translational speed [20]. We consider here a purely kinematic problem, and neglect any effects due to dynamic constraints (e.g., acceleration bounds). *The DDR wants to minimize the capture time t_f while the evader wants to maximize it. The objective is to find the optimal strategies that are in Nash Equilibrium and may be used by both players to achieve their goals.*

III. Model

A. Realistic space

The kinematics of the game can be described in a global coordinate system (refer to Fig. 3(a)). (x_p, y_p, θ_p) represents the pose of the DDR and (x_e, y_e) is the position of the omnidirectional evader, both at time t. The state of the system can be expressed as $(x_p, y_p, \theta_p, x_e, y_e) \in \mathbb{R}^2 \times S^1 \times \mathbb{R}^2$. The evolution of the system is described by the following equations of motion

$$\begin{aligned} \dot{x}_p &= \left(\frac{u_1+u_2}{2}\right)\cos\theta_p, \;\; \dot{y}_p = \left(\frac{u_1+u_2}{2}\right)\sin\theta_p \\ \dot{\theta}_p &= \left(\frac{u_2-u_1}{2b}\right), \;\; \dot{x}_e = v_1\cos\psi_e, \;\; \dot{y}_e = v_1\sin\psi_e \end{aligned} \tag{5}$$

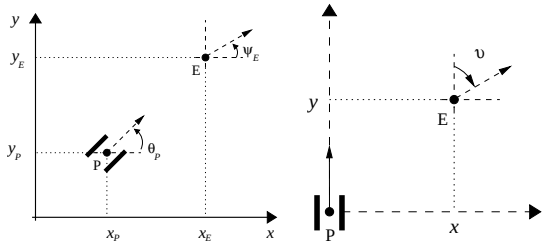
where $u_1, u_2 \in [-V_p^{\max}/r, V_p^{\max}/r]$ are the controls of the DDR, and they correspond to the angular velocities of its wheels. r is the radius of the wheels, in this problem we assume $r = 1$. Let u_1 be the angular velocity of the left wheel

and u_2 of the right wheel. If both controls have the same magnitude and are either positive or negative, respectively, the robot moves forward or backward in a straight line, and with a suitable choice of units [20], the translational speed is equal to $V_p = \frac{1}{2}(u_1 + u_2)$. If u_1 and u_2 have the same magnitude but opposite signs the robot rotates in place either clockwise or counter-clockwise [20]. The evader controls its speed $v_1 \in [0, V_e^{\max}]$ and its direction of motion $\psi_e \in [0, 2\pi)$. We present two useful definitions for the rest of the paper, $\rho_v = V_e^{\max}/V_p^{\max}$ is the ratio between the maximum translational speed of both players, and $\rho_d = b/l$ is the ratio of the distance between the center of the robot and the wheel location b and the capture distance l. We must have that $l \geq b$, otherwise the capture distance would be located inside the robot.

B. Reduced space

Usually it is more convenient to analyze the problem and perform all the computations in a space of reduced dimension. In our case, the problem can be stated in a coordinate system that is fixed to the body of the DDR (see Fig. 3(b)). The state of the system now can be expressed as $\mathbf{x}(t) = (x, y) \in \mathbb{R}^2$. All the orientations in this system are measured with respect to the positive y-axis, in particular, the direction of motion of the evader v_2. Using the coordinate transformation given by

$$\begin{aligned} x &= (x_e - x_p)\sin\theta_p - (y_e - y_p)\cos\theta_p \\ y &= (x_e - x_p)\cos\theta_p + (y_e - y_p)\sin\theta_p \\ v_2 &= \theta_p - \psi_e \end{aligned} \tag{6}$$



(a) Realistic space (b) Reduced space

Computing the time-derivative of x and y in Eq. (6) and substituting the expressions for $\dot{x}_p$, $\dot{y}_p$, $\dot{\theta}_p$, $\dot{x}_e$ and $\dot{y}_e$ in Eq. (5), and the expressions for x and y in Eq. (6), the following model of the kinematics in the DDR-fixed coordinate system is obtained

$$\begin{aligned} \dot{x} &= \left(\frac{u_2 - u_1}{2b}\right) y + v_1 \sin v_2 \\ \dot{y} &= -\left(\frac{u_2 - u_1}{2b}\right) x - \left(\frac{u_1 + u_2}{2}\right) + v_1 \cos v_2 \end{aligned} \tag{7}$$

where $u_1, u_2 \in [-V_p^{\max}, V_p^{\max}]$ are again the controls of the DDR, $v_1 \in [0, V_e^{\max}]$ is the control associated to the speed of the evader and $v_2 \in [0, 2\pi)$ is the control associated to the direction of motion of the evader in the new coordinate system. This set of equations can be expressed in the form $\dot{\mathbf{x}} = f(\mathbf{x}, u, v)$, where $u = (u_1, u_2) \in \widehat{U} = [-V_p^{\max}, V_p^{\max}] \times [-V_p^{\max}, V_p^{\max}]$ and $v = (v_1, v_2) \in \widehat{V} = [0, V_e^{\max}] \times [0, 2\pi)$.

IV. Concepts from differential games and overview of the methodology applied in our problem

In this section, we provide an overview of the methodology applied in our problem and describe some concepts from differential games that will be used in its solution. We use Isaacs' methodology (IM) [1], [2] to find the solution of our pursuit-evasion problem. The IM is based on an extension of the Hamilton-Jacobi-Bellman (HJB) equation [2] (see Subsection A in Appendix A). This equation provides sufficient conditions for the existence of saddle-point equilibrium strategies [2]. Solving the HJB allows one to know the value function $V(\mathbf{x})$ over the entire space; however, obtaining a closed-form solution directly from the HJB is not possible in most cases. As an alternative, the IM uses an extension of the Pontryagin's Maximum Principle (PMP) [27] (refer to Subsection B in Appendix A) which provides a constructive manner for computing saddle-point equilibrium strategies. The PMP only provides necessary conditions for these strategies; Together with the PMP it is necessary to define a partition of the state space into regions, such that in the interior of each one of them the optimal controls for each player are uniquely determined by the PMP (see Section VIII). Additionally, at the boundaries between regions it may be necessary to choose between two or more trajectories, so that in order to guarantee global optimality, rules for performing these choices must be specified. In IM these rules are based on the type of those boundaries, which are called singular surfaces [1], [2]. In robotics literature, the complete process for constructing a globally optimal control law for steering the representative point from any point of the space to a given target set is called the synthesis problem [21]. Typically, to solve the synthesis problem, first one finds the optimal motion primitives and then a partition of the playing space is obtained to guarantee global optimality [20], [21]. In Section IV-E, we present in more detail the methodology used to solve our problem.

We now define the value of the game which is a central concept in the solution of our problem.

A. Value of the game

In a noncooperative game, the numerical quantity which the players strive to maximize and minimize respectively can assume a variety of forms. A common representation of the payoff is

$$J(\mathbf{x}(t_s), u, v) = \int_{t_s}^{t_f} L(\mathbf{x}(\overline{t}), u(\overline{t}), v(\overline{t}))d\overline{t} + G(\mathbf{x}(t_f)) \tag{8}$$

The time integral extends over the path traversed by $\mathbf{x}(\overline{t})$ during the game; its lower limit (we call it t_s) refers to the starting state $\mathbf{x}(t_s)$; its upper limit is the time t_f to reach the final state $\mathbf{x}(t_f)$. $L(\mathbf{x}(t), u(t), v(t))$ is called the *running cost function* and it is the cost incurred while the game is being played. The term $G(\mathbf{x}(t_f))$ is called the *terminal cost function* and it is the cost incurred for reaching a particular terminal state.

In problems of *minimum time*, as in this work, one of the players (in this problem, the pursuer) wants to minimize the

time it takes to reach the final state $\mathbf{x}(t_f)$ from the initial state $\mathbf{x}(t_s)$, which is represented by the difference $t_f - t_s$. In those problems, usually there is no special cost for reaching a particular terminal state, so that $G(\mathbf{x}(t_f)) = 0$. From Eq. (8), if $L(\mathbf{x}(t), u(t), v(t)) = 1$ and $G(\mathbf{x}(t_f)) = 0$, we can see that the time integral reduces to $t_f - t_s$. Therefore in our game, the payoff is represented as

$$J(\mathbf{x}(t_s), u, v) = \int_{t_s}^{t_f(\mathbf{x}(t_s),u,v)} d\bar{t} = t_f(\mathbf{x}(t_s), u, v) - t_s \qquad (9)$$

Note that $t_f(\mathbf{x}(t_s), u, v)$ depends on the sequence of controls u and v applied to reach the point $\mathbf{x}(t_f)$ from the point $\mathbf{x}(t_s)$.

For a given state of the system $\mathbf{x}(t_s)$, $V(\mathbf{x}(t_s))$ represents the outcome if the players implement their equilibrium strategies starting at the point $\mathbf{x}(t_s)$, and it is called the *value of the game* at $\mathbf{x}(t_s)$

$$V(\mathbf{x}(t_s)) = \min_{u(t)\in\widehat{U},\forall t} \max_{v(t)\in\widehat{V},\forall t} J(\mathbf{x}(t_s), u, v) \qquad (10)$$

where $\widehat{U}$ and $\widehat{V}$ are the set of valid values for the controls for all time t. Note that in Eq. (10), the players have opposite goals, one player maximizes the cost while the other player minimizes it, both through the selection of their controls. $V(\mathbf{x}(t))$ is defined over the entire state space. In this game, Eq. (10) actually corresponds to the time it takes for the DDR to capture the evader, when the players implement their equilibrium strategies starting at $\mathbf{x}(t_s)$. No matter what the evader does, it cannot avoid the capture for longer than $V(\mathbf{x}(t_s))$. If the evader does anything different than its equilibrium strategy, the DDR can capture it in less than $V(\mathbf{x}(t_s))$. Reciprocally, if the DDR does anything different than its equilibrium strategy, the evader can remain avoiding it for a longer time.

B. Open and closed-loop strategies

We start by defining a strategy γ which is a rule that tells each player the control it has to apply at each time instant. If the strategy only depends on time $\gamma(t)$ is called a open-loop strategy, and if it depends on the state of the system $\gamma(\mathbf{x}(t))$ is called a closed-loop strategy [2]. In this work, we find open-loop strategies using the Pontryagin's Maximum Principle (refer to Subsection B in Appendix A) and we define a partition of the entire state space into regions (refer to Section VIII).

C. Open and closed-loop equilibrium strategies

Let $\gamma_p(\mathbf{x}(t))$ and $\gamma_e(\mathbf{x}(t))$ denote the closed-loop strategies of the DDR and the evader, respectively, therefore $u(t) = \gamma_p(\mathbf{x}(t))$ and $v(t) = \gamma_e(\mathbf{x}(t))$. A strategy pair $(\gamma_p^*(\mathbf{x}(t)), \gamma_e^*(\mathbf{x}(t)))$ is in closed-loop (saddle-point) equilibrium if

$$\begin{aligned} J(\mathbf{x}(t_s), \gamma_p^*(\mathbf{x}(t)), \gamma_e(\mathbf{x}(t))) &\leq J(\mathbf{x}(t_s), \gamma_p^*(\mathbf{x}(t)), \gamma_e^*(\mathbf{x}(t))) \\ &\leq J(\mathbf{x}(t_s), \gamma_p(\mathbf{x}(t)), \gamma_e^*(\mathbf{x}(t))), \\ &\forall \gamma_p(\mathbf{x}(t)), \gamma_e(\mathbf{x}(t)) \end{aligned} \qquad (11)$$

where J is the payoff of the game in terms of the strategies. An analogous relation exists for open-loop strategies.

D. Termination situations

Two important concepts for modeling our problem are the terminal surface and the usable part which we define below. Every state of the system in which the distance between both players equals l represents an opportunity for the DDR to capture the evader. This set of points is called *terminal surface* [1] or *target set* [2], which can be characterized by a scalar function $\zeta(\mathbf{x}(t)) = 0$. We will refer to this set simply as ζ.

In our game, termination occurs only when the distance between the DDR and the evader is smaller than a critical value l despite any opposition of the evader. The portion of the terminal surface where the DDR can guarantee termination regardless of the choice of controls of the evader is called the *usable part* (UP) [1]. From [1], we have that the UP of our problem is given by

$$\text{UP} = \left\{ \mathbf{x}(t) \in \zeta : \min_{u(t)\in\widehat{U}} \max_{v(t)\in\widehat{V}} \mathbf{n} \cdot f(\mathbf{x}(t), u(t), v(t)) < 0 \right\} \qquad (12)$$

where $\widehat{U}$ and $\widehat{V}$ are the sets of valid values for the controls, and $\mathbf{n}$ is the normal vector to ζ from point $\mathbf{x}(t)$ on ζ and extending into the playing space. $\mathbf{n} \cdot f(\mathbf{x}(t), u(t), v(t))$ is a projection of the motion directions of both players along the best direction for penetrating ζ and tell us if the strategies of both players will allow crossing the terminal surface or not. Those points of ζ where the expression (12) holds with the inequality reversed are called the *non-usable* part (NUP) and the game will never terminate on the NUP. The set of points that separates these parts is called the *boundary of the usable part* (BUP). The BUP can be computed replacing the inequality in (12) by an equality.

E. Overview of the methodology applied in our problem

We use the IM to obtain the solution to our problem, in this methodology the PMP is used to obtain extremal trajectories that correspond to the application of the optimal controls by the players. However, there may be sets of points in the state space (singular surfaces [1], [2], [34]) where two or more backward-time trajectories converge. The players must decide which trajectory to follow in order to obtain a globally optimal solution. The IM provides a constructive way to find those rules (identification of singular surfaces). In summary, in the IM a complete solution to a pursuit–evasion game involves: 1) Finding the singular surfaces for the particular problem, which induce a partition of the entire state space into regions. 2) Finding the optimal strategies inside each one of these regions by backward-time integration of the corresponding differential equations up to singularities, discarding non-optimal extremal trajectories. 3) Providing rules for selecting the appropriate strategy at those places where more than one choice is possible. Thus, the application of IM for our problem consists of the following steps:

1) Compute the usable part (Subsection V-A). This allows one to find the initial conditions needed to solve the differential equation derived from PMP: the so-called adjoint equation.

2) Construct the Hamiltonian of the system (Subsection V-B) and obtain the *expressions* for the optimal controls satisfying it (Subsection V-C).
3) Find $\nabla V(\mathbf{x})$ solving the adjoint equation backwards in time (retro-time, see Eq. (24)) using the values of $V(\mathbf{x})$ and $\nabla V(\mathbf{x})$ on the UP as initial conditions (Subsections V-D and V-E). $V(\mathbf{x})$ is used to find the controls used by the players.
4) Use the computed controls in the backward integration of the motion equations to find the trajectory followed by the players. (Subsections V-F and V-H).
5) Find the singular surfaces and restart the backward integration of the adjoint and motion equations until the entire space is covered. At those points where multiple backward trajectories converge, choose the globally optimal trajectory based on the type of singular surface. (Subsection V-G and Sections VII and VIII).
6) Find the region of the playing space where capture is possible for the DDR (Section VI).

Steps 1 to 4 corresponds to use the Pontryagin's principle. In step 5, we find the trajectories of the players for the entire space. In step 6, we solve the decision problem corresponding to determining the winner of the game.

V. Optimal motion strategies and trajectories

In this section, we will refer to trajectories in the reduced and realistic spaces. In the realistic space, we will describe trajectories for both the pursuer and the evader over a global reference frame in a Cartesian plane. In the reduced space, we will refer to trajectories of the system, i.e., relative motions of the evader with respect to the pursuer in a local reference frame defined by the pursuer.

A. Computing the usable part and its boundary

In this section, we compute the portion of the space where the pursuer guarantees termination regardless of the choice of controls by the evader. For this problem, the terminal surface ζ is characterized by the distance l between both players. In the reduced space, ζ is a circle of radius l centered at the origin, hence we can parametrize it by the angle s (see Fig. 3), which is the angle between the evader's position and the pursuer's heading *at the end of the game* (recall that all the orientations in the reduced space are measured with respect to the positive y-axis). At the end of the game

$$x = l\sin s,\ y = l\cos s \tag{13}$$

Lemma 1: In this game, the usable part has two regions: 1) The first region corresponds to capturing the evader when the DDR is moving forward following a straight line in realistic space. This region contains all the points on ζ such that $\cos s > \rho_v$ and its boundary is given by those points where $\cos s = \rho_v$. 2) The second region corresponds to capturing the evader when the DDR is moving backward following a straight line in realistic space. This region contains all the points on ζ such that $\cos s < -\rho_v$ and its boundary is given by those points where $\cos s = -\rho_v$.

Proof: The outward normal $\mathbf{n}$ to ζ is defined by

$$\mathbf{n} = [\sin s \ \ \cos s] \tag{14}$$

The usable part after substituting Eq. (7) and Eq. (14) into inequality (12) is given by

$$\begin{aligned}\text{UP} =\{s : \min_{u_1,u_2}\max_{v_1,v_2}\{&\sin s\left[\left(\frac{u_2-u_1}{2b}\right)y + v_1\sin v_2\right]\\ &+\cos s\left[-\left(\frac{u_2-u_1}{2b}\right)x - \left(\frac{u_1+u_2}{2}\right) + v_1\cos v_2\right]\} < 0\}\end{aligned} \tag{15}$$

Substituting Eq. (13) into inequality (15) and after straightforward algebraic manipulation, we find that

$$\text{UP} = \left\{s : \min_{u_1,u_2}\max_{v_1,v_2}\left[v_1\cos(v_2 - s) - \left(\frac{u_1+u_2}{2}\right)\cos s\right] < 0\right\} \tag{16}$$

As the evader is the maximizer player it wants the term $v_1\cos(v_2 - s)$ to be positive, and with the largest value possible. Therefore, $v_1 = V_e^{\max}$ and $v_2 = s$, i.e., the evader is moving at maximum speed with an angle s with respect to the pursuer's heading. Substituting these values into Eq. (16) we have

$$\text{UP} = \left\{s : \min_{u_1,u_2}\left[V_e^{\max} - \left(\frac{u_1+u_2}{2}\right)\cos s\right] < 0\right\} \tag{17}$$

In inequality (17) we have two cases, (1) $\cos s > 0$ or (2) $\cos s < 0$. In order to make inequality (17) minimal, u_1 and u_2 must be equal and saturated (that is, equal to $|V_p^{\max}|$). Hence the pursuer moves in a straight line. If $\cos s > 0$ then $\left(\frac{u_1+u_2}{2}\right) = V_p^{\max} > 0$, the DDR is moving forward and if $\cos s < 0$ then $\left(\frac{u_1+u_2}{2}\right) = -V_p^{\max} < 0$, the DDR is moving backward. Note that this pair of controls corresponds to the best action that the DDR can apply against the evader in the $\min\max$ context of the game, and therefore they give the set of configurations where the DDR captures the evader against any opposition of this player. Note that the same controls u_1 and u_2 are used in both the reduced and realistic spaces. From inequality (17) and considering the two cases described above, it is straightforward to find that the region where the DDR is moving forward contains all the points such that $\cos s > \rho_v$ and the region where the DDR is moving backward contains all the points such that $\cos s < -\rho_v$.

■

B. Hamiltonian

In order to compute the optimal trajectories for both players, one needs to construct the Hamiltonian of the system. As it was mentioned previously, for problems of *minimum time* [2], as in this game, $L(\mathbf{x}(t), u(t), v(t)) = 1$ and $G(\mathbf{x}(t_f)) = 0$. $\nabla V = [V_x \ \ V_y]^T$ where V_x and V_y represent the partial derivatives $\frac{\partial V}{\partial x}$ and $\frac{\partial V}{\partial y}$. Substituting the last expressions and the equations of motion in (7) into Eq. (62), we obtain

$$\begin{aligned}H(\mathbf{x}, \nabla V, u_1, u_2, v_1, v_2) = &V_x\left(\frac{u_2-u_1}{2b}\right)y + V_x v_1\sin v_2\\ &- V_y\left(\frac{u_2-u_1}{2b}\right)x - V_y\left(\frac{u_1+u_2}{2}\right) + V_y v_1\cos v_2 + 1\end{aligned} \tag{18}$$

Lemma 2: The Hamiltonian of the system is separable in the controls of the pursuer and the evader, i.e., we can write it in the form $f_1(\mathbf{x}, \nabla V, u) + f_2(\mathbf{x}, \nabla V, v)$.

Proof: In our case, Eq. (18) can be rewritten in the form

$$H(\mathbf{x}, \nabla V, u_1, u_2, v_1, v_2) = \frac{u_1}{2}\left(\frac{-yV_x}{b} + \frac{xV_y}{b} - V_y\right) + \frac{u_2}{2}\left(\frac{yV_x}{b} - \frac{xV_y}{b} - V_y\right) + v_1(V_x \sin v_2 + V_y \cos v_2) + 1 \tag{19}$$

■

C. Optimal controls

Lemma 3: The time-optimal controls for the DDR that satisfy the Isaacs' equation (Eq. 55 in Subsection A of Appendix A) in the reduced space are given by

$$u_1^* = -\text{sgn}\left(\frac{-yV_x}{b} + \frac{xV_y}{b} - V_y\right) V_p^{\max}$$
$$u_2^* = -\text{sgn}\left(\frac{yV_x}{b} - \frac{xV_y}{b} - V_y\right) V_p^{\max} \tag{20}$$

We have that both controls are always saturated. If they have the same sign the DDR will move in straight line at maximum translational speed in the realistic space and if they have opposite signs the DDR will rotate in place at maximum rotational speed in the realistic space. The controls of the evader in the reduced space are given by

$$v_1^* = V_e^{\max},\ \sin v_2^* = \frac{V_x}{\rho},\ \cos v_2^* = \frac{V_y}{\rho} \tag{21}$$

where $\rho = \sqrt{V_x^2 + V_y^2}$. The evader will also move at maximal speed.

Proof: By Lemma 2 we know that the Hamiltonian of our game is separable in two parts, one in terms of the pursuer's controls and other in terms of the evader's controls. Consider the pursuer first. As the DDR is the minimizer player it wants the Hamiltonian term

$$\frac{u_1}{2}\left(\frac{-yV_x}{b} + \frac{xV_y}{b} - V_y\right) + \frac{u_2}{2}\left(\frac{yV_x}{b} - \frac{xV_y}{b} - V_y\right) \tag{22}$$

to be minimal. Let $A = \frac{-yV_x}{b} + \frac{xV_y}{b} - V_y$ and $B = \frac{yV_x}{b} - \frac{xV_y}{b} - V_y$. There are four cases:

In all cases u_1 and u_2 must be saturated to minimize Eq. (22) and they correspond to the maximal rotational speed of the wheels $V_p^{\max}$ (with a suitable choice of units and assuming an unit radius r of the pursuer's wheels [20], the rotational speeds are equivalent to the translational speeds).

1) If $A < 0$ and $B < 0$ then to minimize Eq. (22), $u_1 = u_2 = V_p^{\max}$, and the pursuer moves forward in a straight line.
2) If $A > 0$ and $B > 0$ then to minimize Eq. (22) $u_1 = u_2 = -V_p^{\max}$, and the pursuer moves backward in a straight line.
3) If $A > 0$ and $B < 0$ then to minimize Eq. (22) $u_1 = -V_p^{\max}$ and $u_2 = V_p^{\max}$, and the pursuer rotates in place counterclockwise.
4) If $A < 0$ and $B > 0$ then to minimize Eq. (22) $u_1 = V_p^{\max}$ and $u_2 = -V_p^{\max}$, and the pursuer rotates in place clockwise.

The DDR switches controls when A or B change signs. When the DDR switches controls A or B are instantaneously zero. One can show that if either A or B become zero, the corresponding time derivatives $\dot{A}$ or $\dot{B}$ will be different from zero, so that A or B are zero only at the switching instant.

Analogously, since the evader is the maximizer player it wants the term $v_1(V_x \sin v_2 + V_y \cos v_2)$ to be maximal. The quantity in round parenthesis is the dot product of the vectors $[V_x\ V_y]$ and $[\sin v_2\ \cos v_2]$, and it is maximal when $[\sin v_2\ \cos v_2]$ lies along $[V_x\ V_y]$ (both vectors are parallel and have the same direction). To maximize $v_1(V_x \sin v_2 + V_y \cos v_2)$, $v_1 = V_e^{\max}$ and $[V_x\ V_y] \parallel [\sin v_2\ \cos v_2]$, from which Eq. (21) follows. ■

In Lemmas 6 and 9, we will present the actual evader trajectories.

D. Adjoint equation

The adjoint equation (Eq. 60 in Subsection B of Appendix A) is a differential equation for the gradient of the value function $V(\mathbf{x})$ along the optimal trajectories in terms of the optimal controls. It is given by

$$\frac{d}{dt}\nabla V[\mathbf{x}(t)] = -\frac{\partial}{\partial x} H(\mathbf{x}, \nabla V, u_1^*, u_2^*, v_1^*, v_2^*) \tag{23}$$

where the components of $\nabla V(\mathbf{x})$ are called adjoint variables. If t_f is the termination time of the game, we define the retro-time as

$$\tau = t_f - t \tag{24}$$

The adjoint equation in retro-time form is

$$\frac{d}{d\tau}\nabla V[\mathbf{x}(\tau)] = \frac{\partial}{\partial x} H(\mathbf{x}, \nabla V, u_1^*, u_2^*, v_1^*, v_2^*) \tag{25}$$

Lemma 4: The expressions in retro-time of the adjoint equation of our system are

$$\frac{d}{d\tau}V_x = -\left(\frac{u_2^* - u_1^*}{2b}\right)V_y,\ \ \frac{d}{d\tau}V_y = \left(\frac{u_2^* - u_1^*}{2b}\right)V_x \tag{26}$$

Proof: Substituting Eq. (19) into Eq. (25) (u_1^*, u_2^*, v_1^* and v_2^* denote the optimal controls of both players) it is straightforward to obtain the expressions above. ■

Remark 1: From Eq. (26), notice that the adjoint equation can take four different expressions depending on the values of u_1^* and u_2^*. Therefore, it is necessary to know when and for how long a particular expression is valid during the game, which corresponds to find the switches for controls of the DDR.

In what follows we will show that the players' optimal motion primitives in the realistic space correspond, for the evader, to straight lines (see Lemmas 6 and 9), and for the pursuer to rotations in place and straight lines, Lemma 7. We will also provide the system trajectories in the reduced space (see Theorems 1 and 2).

E. Integrating the adjoint equation starting at the usable part

We need to establish the initial conditions of the system, in this case, the values of V_x and V_y on the UP of ζ. From Eq. (13) we have that

$$\frac{dx}{ds} = l\cos s,\ \ \frac{dy}{ds} = -l\sin s \tag{27}$$

Since $V(\mathbf{x}) = 0$ on the UP of ζ it follows that

$$V_s = \frac{dV}{ds} = \frac{\partial V}{\partial x}\frac{dx}{ds} + \frac{\partial V}{\partial y}\frac{dy}{ds} = 0 \tag{28}$$

Substituting Eq. (27) into Eq. (28)

$$V_x \cos s = V_y \sin s \tag{29}$$

From Eq. (29) we have that on the UP

$$V_x = \lambda \sin s, \;\; V_y = \lambda \cos s \tag{30}$$

where λ is a constant value.

Lemma 5: The solution of the adjoint equation (26) starting at the usable part is

$$V_x = \lambda \sin s, \;\; V_y = \lambda \cos s \tag{31}$$

Proof: From Lemma 1, we know that at the end of game the pursuer follows a translation. Therefore Eq. (26) takes the form

$$\frac{d}{d\tau}V_x = 0, \;\; \frac{d}{d\tau}V_y = 0 \tag{32}$$

One can directly verify that Eq. (31) satisfies Eq. (32). This solution for the adjoint equation will be valid at the UP and as long as the DDR controls do not change, which corresponds to a DDR motion following a straight line in the realistic space. In Lemma 7, we compute the retro-time instant when the DDR switches controls. At this moment, a new integration of the adjoint equation must be done. ■

Lemma 6: At the end of the game, if the pursuer follows its optimal strategy (i.e. moves in a straight line in the realistic space) the corresponding optimal strategy for the evader is also a straight line in the realistic space, and therefore, the system moves in a straight line in the reduced space.

Proof: From Eq. (31), we know that V_x and V_y have constant values. Substituting those values into the evader's controls in Eq. (21), we find that $v_2^* = s$, the evader's motion direction in the reduced space, is also constant, thus the system follows a straight line in the reduced space at the end of the game.

From Lemma 1, we know that the DDR is moving in straight line in the realistic space at the end of the game. Therefore its motion direction θ_p is constant. From the third equation in the coordinate transformation, Eq. (6), and as v_2 and θ_p are constant, it is straightforward to see that ψ_e, the evader's motion direction in the realistic space, will be constant. ■

Remark 2: From Lemma 3, the controls of the players *are independent*, it would be misleading to conclude that Lemma 6 implies that the evader's controls depend on the pursuer's controls. But in order to show a graphical representation of the trajectories in the realistic space it is necessary to know the controls of the DDR to compute the transformation between the reduced and realistic spaces.

F. Integrating the motion equations starting at the usable part

Theorem 1: The retro-time trajectories of the system in the reduced space leading directly to the end of the game are

$$\begin{aligned} x(\tau) &= -\tau V_e^{\max} \sin s + l \sin s \\ y(\tau) &= \tau(-V_e^{\max} \cos s \pm V_p^{\max}) + l \cos s \end{aligned} \tag{33}$$

the sign $+$ is taken if the pursuer moves forward in the realistic space when it captures the evader and the sign $-$ if it moves backward.

Proof: From Eq. (7), the retro-time version of the equations of motion in the reduced space is

$$\begin{aligned} \frac{d}{d\tau}x &= -\left(\frac{u_2 - u_1}{2b}\right) y - v_1 \sin v_2 \\ \frac{d}{d\tau}y &= \left(\frac{u_2 - u_1}{2b}\right) x + \left(\frac{u_1 + u_2}{2}\right) - v_1 \cos v_2 \end{aligned} \tag{34}$$

Substituting Eq. (31) into the controls expressions in Eq. (20) and Eq. (21), and the resulting expressions into Eq. (34) we obtain

$$\frac{d}{d\tau}x = -V_e^{\max} \sin s, \;\; \frac{d}{d\tau}y = -V_e^{\max} \cos s + V_p^{\max} \tag{35}$$

when the pursuer is translating forward, and

$$\frac{d}{d\tau}x = -V_e^{\max} \sin s, \;\; \frac{d}{d\tau}y = -V_e^{\max} \cos s - V_p^{\max} \tag{36}$$

when the pursuer is translating backward. Integrating Eq. (35) and Eq. (36) with the initial conditions $x = l \sin s$ and $y = l \cos s$ leads to the expressions in Eq. (33) for the trajectories. ■

Remark 3: The trajectories in Eq. (33) are referred as the *primary solution* [1].

G. Transition surface

The solutions in Eq. (31), and Eq. (33) are valid as long as the DDR does not switch controls. The place where a control variable abruptly changes in value, is known as a *transition surface*. In our problem, after a retro-time interval the DDR switches controls and it starts rotating in place in the realistic space.

Lemma 7: The DDR switches controls and it starts a rotation in place in the realistic space, at $\tau_s = |\frac{b \cos s}{V_p^{\max} \sin s}|$. If $s \in [0, \pi]$, u_2^* switches first, otherwise u_1^* does.

Proof: We can compute the time τ_s when the DDR switches controls, substituting Eq. (31) and Eq. (33) into Eq. (20), and verifying which one of the resulting expressions is the first in changing signs. Doing that we find that for $s \in [0, \frac{\pi}{2}]$, u_2^* switches first and it does it at

$$\tau_s = \frac{b \cos s}{V_p^{\max} \sin s} \tag{37}$$

The other cases can be proved using an analogous reasoning. ■

At τ_s, we need to start a new integration of the retro-time version of the adjoint equation (26) and the equations of motion (34). This integration takes as initial conditions the values of V_x, V_y, x, and y at τ_s. We will denote those values as $V_{x_{\tau_s}}$, $V_{y_{\tau_s}}$, x_{τ_s} and y_{τ_s}. The equations in Lemmas 8 and 9, and Theorem 2 were constructed after the DDR switches controls and it starts rotating in place in the realistic space.

Lemma 8: The solution of the adjoint equation (26) starting at τ_s is

$$\begin{aligned} V_x &= \lambda \sin\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \\ V_y &= \lambda \cos\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \end{aligned} \tag{38}$$

for $\tau \geq \tau_s$.

Proof: Computing the retro-time derivative of Eq. (26), we obtain two ordinary linear differential equations of second order with constant coefficients

$$\frac{d^2}{d\tau^2}V_x = -\left(\frac{u_2^* - u_1^*}{2b}\right)^2 V_x, \;\; \frac{d^2}{d\tau^2}V_y = -\left(\frac{u_2^* - u_1^*}{2b}\right)^2 V_y \tag{39}$$

Solving these equations with the values of V_x and V_y at τ_s as initial conditions we obtain the expressions in Eq. (38). ■

Lemma 9: For $\tau > \tau_s$, the optimal controls correspond to the evader following a straight line in the realistic space and the DDR rotating in place.

Proof: Substituting Eq. (38) into Eq. (21) we have that,

$$\begin{aligned} \sin v_2^* &= \sin\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \\ \cos v_2^* &= \cos\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \end{aligned} \tag{40}$$

therefore

$$v_2^* = s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s) \tag{41}$$

As the DDR is rotating in place, its motion direction is given by

$$\theta_p^{'} = \theta_p^s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s) \tag{42}$$

where θ_p^s is the initial motion direction of the DDR in the realistic space. Substituting Eq. (41) and Eq. (42) into the third expression in Eq. (6), we obtain that $\psi_e = \theta_p^s - s$, the evader's motion direction in realistic space. Note that it is a constant value, thus the evader is following a straight line in realistic space. ■

Note again that from Lemma 3, the controls of the players *are independent*. But in order to show a graphical representation of the trajectories in the realistic space it is needed to know the controls of the DDR.

H. Integrating the motion equations starting at the TS

Theorem 2: The retro-time trajectories of the system starting at τ_s are

$$\begin{aligned} x(\tau) =& -y_{\tau_s}\sin\left[\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] + x_{\tau_s}\cos\left[\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \\ & - (\tau - \tau_s)V_e^{\max}\sin\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \\ y(\tau) =& x_{\tau_s}\sin\left[\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] + y_{\tau_s}\cos\left[\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \\ & - (\tau - \tau_s)V_e^{\max}\cos\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \end{aligned} \tag{43}$$

Proof: Substituting Eq. (38) into Eq. (21), and the resulting expressions into Eq. (34) we obtain

$$\begin{aligned} \frac{d}{d\tau}x &= -\left(\frac{u_2^* - u_1^*}{2b}\right)y - V_e^{\max}\sin\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \\ \frac{d}{d\tau}y &= \left(\frac{u_2^* - u_1^*}{2b}\right)x - V_e^{\max}\cos\left[s - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right] \end{aligned} \tag{44}$$

Computing the retro-time derivative of Eq. (44) and solving the resulting expressions with the initial conditions x_{τ_s} and y_{τ_s} using an analogous reasoning to the one applied in the proof of Lemma 8, we obtain the solution in Eq. (43). ■

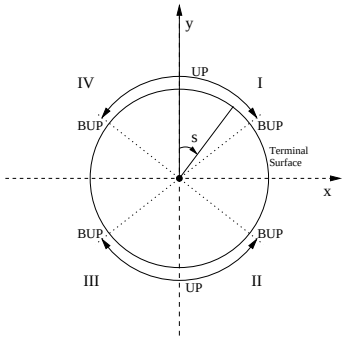


Fig. 3. Representation of the terminal surface, usable part and its boundary in the reduced space.

VI. Decision problem

A game of kind is a game in which we are interested in what conditions lead to a winning for each one of the players. In our case, this corresponds to find the conditions that make capture possible for the DDR or escape for the evader.

A. The barrier

There is a surface called the *barrier* [1], which separates the set of starting positions in those that result in capture and those that result in escape for the evader. From starting points on the barrier, optimal behavior leads to a contact of the terminal surface without crossing it. The outcome of following the barrier is called neutral, and it can be understood as intermediate between capture and escape. The techniques we have used in the calculation of the optimal strategies and their corresponding trajectories, are also applied in the construction of the barrier, which can be interpreted as a *neutral trajectory of the system*. The answer to the capture-escape question relies on whether or not the barrier divides the playing space into two parts.

B. Construction of the barrier

Let $\mathbf{x}$ be a point initially on ζ, the terminal surface. As we previously mentioned, the portion of the terminal surface where the DDR can guarantee termination regardless of the choice of controls of the evader is called the usable part (UP), and its boundary (BUP) is characterized by

$$\text{BUP} = \left\{ \mathbf{x}(t) \in \zeta : \min_{u(t)\in\widehat{U}} \max_{v(t)\in\widehat{V}} \mathbf{n}\cdot f(\mathbf{x}(t), u(t), v(t)) = 0 \right\} \tag{45}$$

where $\mathbf{n}$ is the normal vector to ζ from point $\mathbf{x}(t)$ on ζ and extending into the playing space.

For such points, when each player applies its optimal strategies $\mathbf{x}$ moves tangentially to ζ. As the BUP separates the points on ζ where immediate capture occurs from those where it does not, it is used as initial condition for the barrier. The barrier is constructed integrating the adjoint equation (26) and the equations of motion (34), starting at the BUP. The resulting surface may or may not divide the playing space into two parts, one of them contiguous to the UP.

Suppose the barrier separates the playing space into two parts. If $\mathbf{x}$ is in the outer side, the one that is not contiguous to the UP, then the DDR cannot force the capture because the UP is not accessible. If the barrier fails to separate the playing space, then capture can always be attained by the DDR. However, from starting points in each side of the barrier (in local sense) the DDR must adopt different strategies.

C. Symmetry of the problem

Figure 3 shows a representation of the terminal surface, the usable part (UP) and its boundary (BUP) in the reduced space. The system exhibits some symmetries with respect to the x and y-axis in this representation. An analysis for the trajectories in the first quadrant will be provided. This analysis can be extended to the remaining quadrants using an analogous reasoning.

D. Solving the decision problem

We present two useful properties appearing in some trajectories reaching the UP.

Lemma 10: The retro-time trajectories starting at the UP in the first quadrant (see Fig. 3) reach the y-axis before the system switches controls if $l/V_e^{\max} \leq \tau_s$.

Proof: When the retro-time trajectories reach the y-axis we have that $x = 0$. From Eq. (33)

$$-\tau V_e^{\max} \sin s + l \sin s = 0 \tag{46}$$

By straightforward algebraic manipulation, we find that $\tau = l/V_e^{\max}$. This is the retro-time it takes to reach the y-axis if the system is following Eq. (33), and it will be denoted as $\tau_c = l/V_e^{\max}$. We know that the DDR switches controls at $\tau = \tau_s$. After that, the system starts following Eq. (43). If $\tau_c \leq \tau_s$ the system will reach the y-axis before switching controls. ■

Lemma 11: The trajectories in Eq. (33) that reach the y-axis in the first quadrant, reach it at $y = l/\rho_v$.

Proof: From Lemma 10, we have that $\tau_c = l/V_e^{\max}$ is the retro-time it takes to reach the y-axis when the system is following Eq. (33). Substituting τ_c into Eq. (33) we have that

$$y = \frac{l}{V_e^{\max}}(-V_e^{\max} \cos s + V_p^{\max}) + l \cos s \tag{47}$$

After straightforward algebraic manipulation, we find that $y = l/\rho_v$. This value will be denoted as y_c. ■

Lemma 12: The barrier consists of a straight line segment, and it intersects the y-axis in the first quadrant if $\rho_v \geq |\tan S|/\rho_d$ where $S = \cos^{-1}(\rho_v)$ is the angle at the BUP (see Fig. 3).

Proof: In our game, the barrier is constructed by substituting the value S that satisfies $\cos S = \rho_v$ into Eq. (33). The expression in Eq. (33) is valid as long as the DDR does not switch controls. After a retro-time interval τ_s the DDR should switch controls and start rotating in place in the realistic space. Then the system should follow the trajectory described by Eq. (43) in the reduced space. Figure 4 shows both trajectories. The trajectory given by Eq. (43) intersects the initial segment of the barrier and it comes back to the UP in the reduced space. According to [1], the barrier is not crossed by any trajectory followed by the system during optimal play, in particular, it cannot cross itself. Therefore the portion of the trajectory given by Eq. (43), (the arc in Fig. 4) must be discarded. The barrier reaches the terminal surface with $S = \cos^{-1}(\rho_v)$, and it consists only of a straight line in the reduced space given by Eq. (33) that ends when $\tau = \tau_s$. From Lemma 10, it is straightforward to verify that the barrier will reach the y-axis if $\tau_c \leq \tau_s$. Substituting the values of τ_c and τ_s in the last inequality, we find that it can be expressed as $\frac{V_e^{\max}}{V_p^{\max}} \geq \frac{l|\tan S|}{b}$, which can be rewritten as $\rho_v \geq |\tan S|/\rho_d$. ■

Remark 4: Note that if the system follows the trajectory composed by the arc from point 1 to point 2 (see Fig. 4), and the straight line from point 2 to point 3, the DDR loses the game. The distance between both players equals l over the target set, however, the pursuer will not be able to get closer from the evader than this value and capture cannot be attained (since in the reduced space, the system is pointing tangentially to the terminal surface and it cannot be crossed). In contrast, if the system follows the straight line motion from point 1 to point 4, the system reaches the usable part and it can be crossed. The distance between the players can be reduced by the pursuer and it wins. Hence, the arc given by Eq. (43) must be discarded. Indeed, it can be proved that traveling the arc from point 1 to point 2, and the straight line trajectory from point 2 to point 3 takes more time than traveling the straight line from point 1 to point 4.

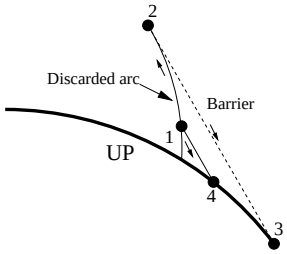


Fig. 4. The barrier

Theorem 3: If $\rho_v < |\tan S|/\rho_d$ the DDR can capture the evader from any initial configuration in the playing space. Otherwise the barrier separates the playing space into two regions, one of them contiguous to the UP. The DDR can only force the capture in the region contiguous to UP, in which case, the DDR follows a straight line in the realistic space when it captures the evader.

Proof: It follows from the definition of the barrier and Lemma 12. Note that the segment of the barrier corresponding to a rotation in place of the DDR in the realistic space has been discarded and all the trajectories between the barrier and the UP are straight lines reaching the y-axis (refer to Lemma 11 and see Fig. 6). ■

Remark 5: For the rest of this work, we assume that the barrier does not intersect the y-axis and therefore capture in all the playing space can be attained by the DDR. In Section IX, we make an exception including simulations where the barrier separates the playing space into two regions and showing the strategy followed by the evader to avoid capture.

VII. Singular surfaces

The singular surfaces and their construction in the context of our game will be described in the next paragraphs giving a solution of the game for the complete playing space.

A. Definition of a singular surface

Following the definition given in [2], a singular surface is a manifold on which: 1) The equilibrium strategies are not uniquely determined by the necessary conditions of Theorem 5, or 2) the value function is not continuously differentiable, or 3) The value function is discontinuous.

Below, we describe the singular surfaces that we have found in this game.

1) Transition surface (TS): The place where a control variable abruptly changes in value, is known as a transition surface. The procedure for locating a transition surface is fairly straightforward, since it follows from the adjoint and motion equations (see Lemma 15 for their construction in this game).

2) Universal surface (US): A surface to which optimal trajectories enter from both sides –called the tributary trajectories– and then stay, is called a universal surface. In differential games, one can think of such a surface as a union of especially advantageous paths. Optimal play will demand that the state of the system be brought to the universal surface and thereafter remain on it (see Lemma 16 for their construction in this game).

3) Dispersal surface (DS): A dispersal surface is defined in [1], [2] as the locus of initial conditions along which the optimal strategy of one player or the optimal strategies of both players are not unique. They are often found as the retro-time intersection of two distinct families of optimal paths, for each of which the Isaacs equation is satisfied. At the intersection, the optimal time-to-go is the same for either pair of strategies. In our game, they appear due to symmetries in the reduced space, see subsection VIII-B and Fig. 6.

In this game we have found only these three types of singular surfaces. However there are other known types in the literature [2].

B. Identification and construction of the singular surfaces

In this section, we describe the construction of the singular surfaces appearing in this game.

Lemma 13: The retro-time trajectories reaching the y-axis in the first quadrant have an orientation $s \in [0, \tan^{-1}(\rho_v \rho_d)]$ at the UP (see Fig. 3).

Proof: From Lemma 10, the retro-time trajectories that reach the y-axis are those where $\tau_c \leq \tau_s$. The last one that can reach it will have $\tau_c = \tau_s$. Substituting the corresponding values

$$\frac{l}{V_e^{\max}} = \frac{b \cos s}{V_p^{\max} \sin s} \tag{48}$$

From the last expression we find that

$$\tan s = \rho_v \rho_d \tag{49}$$

The trajectory given by $s = 0$ coincides with the y-axis. Therefore, the trajectories reaching the y-axis will have an angle $s \in [0, \tan^{-1}(\rho_v \rho_d)]$ at the UP. ■

Lemma 14: The straight lines trajectories that have an orientation $s \in (\tan^{-1}(\rho_v \rho_d), \cos^{-1}(\rho_v)]$ in the UP of the first quadrant terminate when the DDR switches controls.

Proof: From Lemma 13 we know that the last trajectory reaching the y-axis has an orientation $s_c = \tan^{-1}(\rho_v \rho_d)$. If $s > s_c$ the DDR switches controls before reaching the y-axis and the system starts following the trajectories given by Eq. (43). The value $s = \cos^{-1}(\rho_v)$ corresponds to the barrier and it consists of a straight line in the reduced space. Thus the straight line trajectories reaching the UP at $s \in (\tan^{-1}(\rho_v \rho_d), \cos^{-1}(\rho_v)]$ terminate when a switch of the DDR controls occurs. ■

1) Transition surface (TS):

Lemma 15: The points $\mathbf{x}$ in the reduced space where $\tau = \tau_s$ constitute a TS in the first quadrant. Here, the expression $yV_x - xV_y - bV_y = 0$ is satisfied. This surface is bounded by the barrier and the y-axis.

Proof: From Lemma 14, we know that the trajectories ending at $s \in (\tan^{-1}(\rho_v \rho_d), \cos^{-1}(\rho_v))$ have a switch when $\tau = \tau_s$. The points $\mathbf{x}$ where this happen constitute the transition surface (TS). For the first quadrant, in those points u_2 changes sign. Note that $s = \tan^{-1}(\rho_v \rho_d)$ generates a straight line trajectory of the system in the reduced space reaching the y-axis just before switching controls. If $s = \cos^{-1}(\rho_v)$, the trajectory corresponds to the barrier, which is a straight line in the reduced space ending also just before switching controls. Thus the y-axis and the barrier bound the TS. ■

Remark 6: The TS indicates the points in the reduced space where the DDR switches controls, and it is not a trajectory followed by the system.

2) Universal surface (US):

Lemma 16: The positive y-axis contains a US where the pursuer follows the evader with its heading directly aligned to it.

Proof: In a universal surface (US), optimal play demands that the system be brought to the surface and remain on it. Hence a necessary condition for a US, is that in this surface there are no switches and the controls of the players remain constant. We find that in this game, this occurs when the system is moving along the positive y-axis in the reduced space.

From Lemma 13, one time-optimal trajectory for the system starting at the UP and reaching the point y_c corresponds to a straight line with $s = 0$, i.e., the evader's relative position aligned with the pursuer's heading. From Lemma 5 we know that along this trajectory starting at UP and reaching y_c, the following equations hold

$$V_x = \lambda \sin s, \quad V_y = \lambda \cos s \tag{50}$$

where λ is a constant value. At y_c and as $s = 0$,

$$V_x = 0, \quad V_y = \lambda \tag{51}$$

Substituting those values into the pursuer's controls Eq. (20), we find that the expressions inside the sign functions

$$-y\frac{V_x}{b} + x\frac{V_y}{b} - V_y = -\lambda \tag{52}$$

and

$$y\frac{V_x}{b} - x\frac{V_y}{b} - V_y = -\lambda \tag{53}$$

are constant. Therefore, the DDR does not need to switch controls. Substituting Eq. (51) into the evader's controls Eq. (21), we find that the motion direction of the evader is given by

$$\sin v_2^* = 0, \quad \cos v_2^* = 1 \tag{54}$$

which corresponds to a constant motion direction. Hence, the system is moving in straight line over the y-axis in the reduced space.

In the realistic space, both players are moving following a straight line while the system is moving over the positive y-axis in the reduced space. ■

Remark 7: The tributary trajectories entering the US are generated by a different combination of the player's optimal controls to the ones used over the surface [1].

Lemma 17: The tributary trajectories entering the US associated to the first quadrant correspond to a rotation in place of the DDR and straight line for the evader in the realistic space.

Proof: Over the US associated to the first quadrant in the reduced space, we have that the DDR always captures the evader moving forward, therefore the tributary trajectories will correspond to a rotation in place of the DDR, in the realistic space. For the first quadrant, we have that $u_1^* = V_p^{\max}$ and $u_2^* = -V_p^{\max}$ (the DDR rotates clockwise to align its heading with the evader's motion direction in the realistic space). Taking these controls, the trajectories in the reduced space can be computed using an analogous reasoning to the one applied in Theorem 2. In fact, they satisfy Eq. (43) taking $u_1^* = V_p^{\max}$, $u_2^* = -V_p^{\max}$ and $\tau_s = d/V_e^{\max}$, where $d \geq y_c$ is the distance to the UP along the y-axis. ■

VIII. Partition of the space

In this section, we present a partition of the first quadrant into three regions. An analogous reasoning can be used for the other three quadrants yielding a partition of the entire reduced space. All the points in each region can be reached by a particular combination of the motion strategies of the players. This partition will contain some singular surfaces. *The complete set of trajectories for each region is sufficient to cover the space*.

A. Construction of the partition

We construct each one of the regions in quadrant I and we verify that the space is covered by them. The complete construction is shown in Fig. 6.

1) Region I: We denote as *region I* the set of points that can reach the UP with a single straight line trajectory in the reduced space, which corresponds to a straight line motion of both, the DDR and the evader, in the realistic space. From Lemmas 12, 13, 14 and 15, we have that the straight line trajectories ending at the UP in the first quadrant are bounded by the y-axis, the barrier (labeled as BS) and the TS. The trajectories in region I can be classified into two types: the ones reaching the y-axis at y_c and the ones reaching the transition surface TS. Both types of trajectories in this region are given by Eq. (33). Examples of trajectories in this region (solid lines) are shown in Fig. 6.

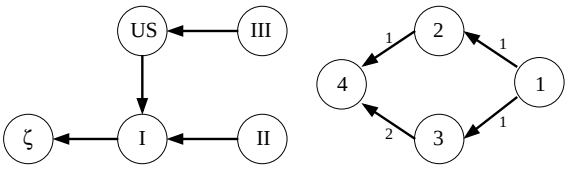
2) Region II: We denote as *region II* the set of points that reach the TS by following a trajectory given by Eq. (43) in the reduced space, which corresponds to a rotation in place for DDR and straight line trajectory for the evader, both in the realistic space. From Lemmas 14, 15, 16 and 17, the trajectories in region II are bounded by the BS, the TS, the x-axis and the trajectory reaching the starting point of the US. Each point inside region II moves according to Eq. (43), it reach the TS at some particular point and to reach the UP it must follow the trajectory in region I reaching the same point in the TS. Some trajectories (dashed lines) in region II are shown in Fig. 6.

3) Region III: We denote as *region III* the set of points that reach the US following one of its tributary trajectories given by Eq. (43) in the reduced space corresponding to a rotation in place for the DDR and a straight line trajectory for the evader, both in the realistic space. From Lemmas 15, 16 and 17, the trajectories in region III are bounded by the US over the y-axis, the point over the TS touching the y-axis and the trajectory given by Eq. (43) reaching that point. Some trajectories in region III (bold dashed lines) are shown in Fig. 6.

Region III in the reduced space corresponds to configurations in the realistic space where the DDR rotates in place until *it aligns its heading* with the segment joining the DDR's position and the evader's position. Then the DDR moves following a straight line towards the evader until the capture condition is achieved. In this case, the evader has the option to change its motion direction at the point over the y-axis in the reduced space where the time-optimal trajectories in region I intersect and it can follow one of them. At this point, the time-optimal trajectories in region I are equivalent and they require the same amount of time to capture the evader. Region II in the reduced space corresponds to configurations in the realistic space where the DDR initially also rotates in place but *it is not necessary to align completely* the DDR's heading with the segment joining the positions of both players in order to capture the evader. In this case, the time-optimal trajectories for both players are unique. We have a bijection between trajectories in region II and region I.

B. Graph representation and playing space partition

Figure 5(a) shows a graph representation of the partition in the first quadrant. The nodes of the graph represent the regions described above, and the edges indicate the transitions between them. For all points in one region of the partition, a particular selection of the controls for both players is used, i.e., the regions are equivalence classes under a relation given by the controls. Note that the transition between regions is uniquely defined, i.e., from the current node in the graph, the system can only reach one node and therefore only a particular selection of the controls for both players can be made. Figure 5(a) shows that from any node in the graph, the system will be able to reach the terminal surface following the transitions given by the edges. Also, this figure shows that the sequence followed by the system and the value function are uniquely determined.



(a) Graph representation of the partition in the first quadrant. (excluding the DS, see text)

(b) An example where the transitions to reach the goal are not uniquely determined.

Fig. 5. Graph representations

It is well-known in optimal control theory that if the system has two possible locally optimal controls in a particular state, one cannot locally make a selection that guarantees global optimality of the solution. For example, Figure 5(b), shows a case, where the transitions between states are not uniquely determined and they have different costs. In this example, from state 1 the system can move to state 2 or 3 with the same cost. From these two states, however, the system moves to state 4 with different costs. It is important to note that from state 1, it is not possible to locally choose the set of transitions that will lead to the minimum cost to reach the state 4, and therefore the PMP, which is a local condition, in general cannot be used by itself to find the globally optimal solution.

In this work, these conditions are met in almost all points of the reduced space, as shown in Fig. 5(a). The only exception is the set of points that belongs to the dispersal surface (DS). In this surface, represented as bold lines in Fig. 6, the rotation-in-place trajectories in retro-time coming from the upper and bottom parts of the UP intersect. Over the DS both players have two choices for their controls. It is important to note that at the DS, the choice of the control of one player *must correspond* to the choice of the control of the other player. If one of the players selects the wrong control, the other player will benefit from that decision. In this problem, the DS corresponds to configurations where the pursuer's heading (orientation of the wheels) is perpendicular to the pursuer's location, and the DDR has the option to rotate either clockwise or counterclowise to catch the evader. If the DDR fails to initially choose the correct sense of rotation against the evader's decision then feedback will be necessary to correct the decision and capture the evader in suboptimal time. To avoid the selection problem, the instantaneous velocity vector of both players should be known, but in general (and in particular for this problem) it is assumed that this information is not available. Therefore, a solution will be to employ an instantaneous mixed strategy (IMS) [1], which means the randomizing of a player's decision in accordance with some probabilistic law until the system is no longer on the DS. The trajectories generated by the correct pair of controls will lead to the same optimal time-to-go. In this problem, the difference will be that at the end the capture will be attained moving forward or backward in straight line in the realistic space. It should be remarked that this particular situation only occurs when the wheels of the DDR are exactly perpendicular to the line that joins the DDR and the evader, a situation that in practice occurs with probability 0.

Another particular behavior occurs at point y_c (see Fig. 6), where the US meets region I, and the straight line system trajectories reaching the UP with an orientation $s \in [0, \tan^{-1}(\rho_v \rho_d)]$ reaching the y-axis. At this unique point, the system has the option to follow any of the straight line trajectories reaching the y-axis, which will lead to the same optimal capture time but the system will have a different position at the UP. In the realistic space, at the point y_c the evader has the option to select among different motion direction, but all of them correspond to the same optimal capture time.

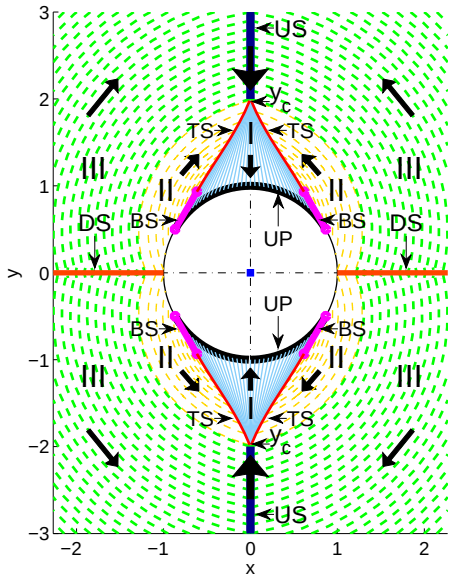


Fig. 6. Partition of the entire reduced space and the corresponding trajectories.

IX. Simulations

In this section, we present some simulation results of our pursuit-evasion game. We use m/sec as units for velocities, meters for distance and seconds for time. First we show the case described in Lemma 12, where the barriers intersect and define regions in the reduced space where the evader can indefinitely escape or is captured in finite time (refer to Fig. 7). The capture condition is only possible for the configurations inside the closed region defined by the barriers (shown in magenta) and the usable part (bold arc in black). Those configurations are the only ones that can reach the terminal surface. The system trajectories inside the closed region end in the y-axis and consist of straight lines in the reduced space (refer to Fig. 6), which corresponds to a straight line motion of the DDR and the evader in the realistic space. The parameters of this simulation were $V_p^{\max} = 1$, $V_e^{\max} = 0.787$, $b = 1$ and $l = 1$. In Fig. 7, we also present a trajectory followed by the system in the reduced space when the evader avoids capture. In this case, we assumed that the evader's position is directly

aligned with the pursuer's heading, i.e., the evader is in front of the pursuer. The initial configuration of the system is outside the closed region. The system first moves over the y-axis following the trajectory (T1). When the system hits the barrier it starts following this trajectory (T2) reaching tangentially the terminal surface. The distance between both players equals l over the target set, however, the pursuer will not be able to get closer from the evader than this value and capture cannot be attained (since in the reduced space, the system is pointing tangentially to the terminal surface). Over the target set, the system will start moving toward the y-axis following the arc trajectory (T3) where the pursuer is aligning is heading with the evader's position. Note that the complete trajectory of the system *is cyclic and the process can be repeated again implying that evader can always avoid capture*.

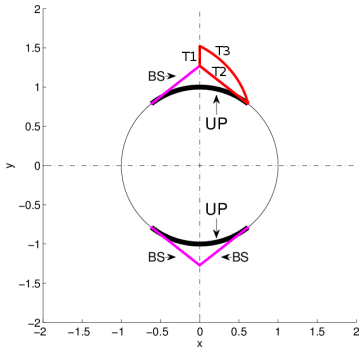


Fig. 7. A trajectory followed by the system in the reduced space when the evader avoids capture.

Figure 8 shows the trajectories followed by both players in the realistic space for the case described above. The pursuer's initial position is denoted by P_I and the evader's initial position by E_I. The arrows show the motion direction of the players. The pursuer starts moving directly towards the evader which moves away following a straight line. After a time interval, which corresponds to reaching the barrier in the reduced space, the evader at E_S switches its motion direction but the pursuer at P_S continues moving in the same direction. The circle represents the time instant when the distance between both players, at P_F and E_F, equals l. Note that at this point no matter what the pursuer does (e.g., continue moving in straight line, rotating in place, etc.) the distance between both players increases. If the pursuer starts rotating in place at P_F, which is the optimal strategy, the whole process is repeated implying that the evader can always avoid capture.

In the next simulation, we will show an example where the pursuer wins by capturing the evader. The parameters were $V_p^{\max} = 1$, $V_e^{\max} = 0.5$, $b = 1$ and $l = 1$. The sample step is $0.001s$ for the reduced space but the trajectories in the realistic space show the positions and motion directions of the players at every 200 iterations. We show the case when two system trajectories start at the same point over the x-axis (DS), at this point the evader has a relative orientation of $\frac{\pi}{2}$ with respect to the pursuer's heading (refer to Fig. 6). The DDR has two possible optimal controls to capture the evader:

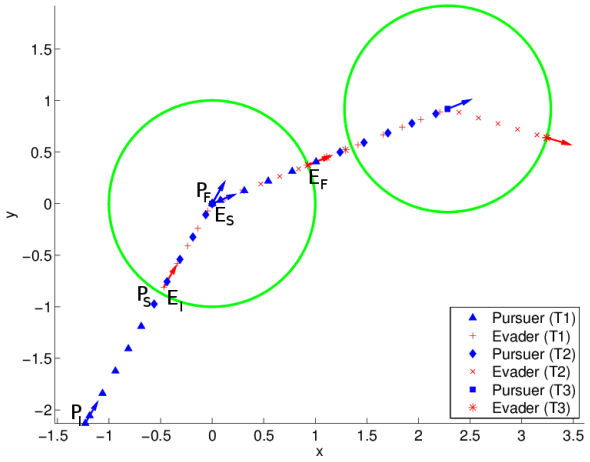


Fig. 8. The trajectories followed by both players in the realistic space when the evader avoids capture. We show the corresponding trajectories in the realistic space when the system is following the trajectories T1, T2 and T3 in the reduced space representation. The arrows show the motion direction of the players. The blue triangles (pursuer) and red plus signs (evader) correspond to T1. The blue diamonds (pursuer) and red crosses (evader) correspond to T2. Finally, the blue squares (pursuer) and red asterisks (evader) correspond to T3.

rotate clockwise or counterclockwise, both leading to the same optimal time-to-go. If the DDR rotates clockwise the trajectory ends in the upper UP and if it rotates counterclockwise the trajectory ends in the bottom UP. For this case, the trajectories pass over regions II and I in order to reach the UP.

Figure 9 shows in the realistic space the trajectories of the evader and the pursuer. These trajectories correspond in the reduced space to the trajectory ending at the upper UP (shown in Fig. 6). In Fig. 9, P_I and E_I are the initial positions of the pursuer and the evader, and P_F and E_F the positions where capture is attained.

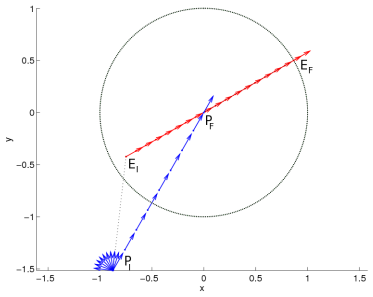


Fig. 9. The DDR captures the evader with a forward motion in the realistic space.

Finally, we use a local strategy (gradient-following) in which the pursuer tries to locally minimize the distance between both players, and the evader tries to maximize it. The strategy for the DDR is to find the controls that minimize the distance between both players at each time instant. The DDR assumes that after both players applied their controls, it can consider a fixed position for the evader; the DDR searches the controls that minimize the distance between the new position of the DDR (from the application of the control) and the fixed

position for the evader. The strategy for the evader is moving away from the DDR at maximal speed, in the same direction of the segment joining the position of the DDR and the evader, at each time instant. This strategy locally maximizes the distance between both players.

In Fig. 10, we can see the trajectories of both players when they apply the local strategies. The same maximal players' speeds and initial configuration, as in Fig. 9, were used.

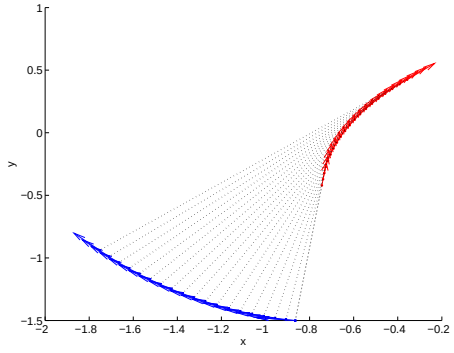


Fig. 10. Gradient following pursuit/evasion.

In this case, we can observe that when the DDR uses the local strategy it cannot capture the evader, as it is done when the DDR applies its optimal strategy. Note that, in this case both players have followed sub-optimal strategies.

We have included videos, available at the Multimedia Material of the paper, showing the simulation results presented in this section and additional ones.

X. Conclusions and Future Work

In this paper, we considered the problem of capturing an omnidirectional evader using a DDR in an obstacle free environment or, more precisely, the problem of getting closer from the evader than the capture distance l. Differently to the classical Homicidal Chauffeur problem [1], [16], in which the pursuer is a car-like vehicle, in this work, the pursuer is a Differential Drive Robot (DDR), i.e. the pursuer can rotate in place. The change in the mechanical model of the pursuer has as a distinctive consequence that both the pursuer motion primitives and the motion strategies of the players also change w.r.t. the Homicidal Chauffeur solution. In this work, we made the following contributions: We presented closed-form representations of the motion primitives and time-optimal strategies for each player. In the realistic space, the motion primitives for the pursuer are straight lines and rotations in place and for the evader are straight lines. The strategies for the players that we have found are in Nash Equilibrium. We proposed a partition of the playing space into mutually disjoint regions where the strategies of the players are well established. The boundaries of these regions are called *singular surfaces* [1], [16], [2], and they indicate a change in one the player's strategies. The partition is represented as a graph which exhibits properties that guarantee global optimality. We also analyzed the decision problem of the game and presented the conditions defining the winner of the game. As future work we will include acceleration bounds in the solution of this problem.

Appendix

This section describes the necessary and sufficient conditions for existence of saddle-point equilibrium strategies in pursuit-evasion games [2]. The sufficient condition is provided by an extension of the Hamilton-Jacobi-Bellman (HJB) equation [2] to a non-cooperative game with two players. This extension is called the Isaacs equation (Eq. (55)) [1], which can be rewritten in terms of the Hamiltonian (Eq. (63)). An analogous extension of the Pontryagin's maximum principle (PMP) [27] to a two-players non-cooperative game provides a necessary condition [2]. This extension of the PMP provides a constructive manner of computing saddle-point strategies.

A. Isaacs equation

Eq. (55) is known as the *Isaacs equation* [2]:

$$-\frac{\partial V(t,\mathbf{x}(t))}{\partial t} = \min_{u(t)\in\widehat{U}} \max_{v(t)\in\widehat{V}} [\frac{\partial V(t,\mathbf{x}(t))}{\partial x} \cdot f(t,\mathbf{x}(t),u(t),v(t)) + L(t,\mathbf{x}(t),u(t),v(t))] \tag{55}$$

where $\widehat{U}$ and $\widehat{V}$ are the sets of valid values for the controls. In this game $V(\mathbf{x}(t))$, $f(\mathbf{x}(t),u(t),v(t))$ and $L(\mathbf{x}(t),u(t),v(t))$ do not explicitly depend on time, therefore Eq. (55) takes the form

$$\min_{u(t)\in\widehat{U}} \max_{v(t)\in\widehat{V}} [\frac{\partial V(\mathbf{x}(t))}{\partial x} \cdot f(\mathbf{x}(t),u(t),v(t)) + 1] = 0 \tag{56}$$

Solving the HJB is a functional optimization problem. This equation provides a sufficient condition for saddle-point strategies, which is stated in the following theorem from [2]:

Theorem 4: If 1) a continuously differentiable function $V(\mathbf{x}(t))$ exists that satisfies the Isaacs equation (56), 2) $V(\mathbf{x}(t)) = 0$ on the boundary of the terminal surface ζ, 3) either $u^*(t) = \gamma_p^*(\mathbf{x}(t))$ or $v^*(t) = \gamma_e^*(\mathbf{x}(t))$, as derived from Eq. (56), generates trajectories that terminate in finite time (whatever γ_e, respectively γ_p, is), then $V(\mathbf{x}(t))$ is the value of the game and $(\gamma_p^*(\mathbf{x}(t)), \gamma_e^*(\mathbf{x}(t)))$ constitutes a saddle point. The reader can consult a sketch of the proof for this theorem in [2] (p. 427). The assumption of interchangeability of the $\min$ and $\max$ operations in the Isaacs equation is referred as the *Isaacs condition*. The Isaacs condition holds if both $L(\mathbf{x}(t),u(t),v(t))$ and $f(\mathbf{x}(t),u(t),v(t))$ are separable in $u(t)$ and $v(t)$, i.e., they can be written as

$$\begin{aligned} L(\mathbf{x}(t),u(t),v(t)) &= L_1(\mathbf{x}(t),u(t)) + L_2(\mathbf{x}(t),v(t)) \\ f(\mathbf{x}(t),u(t),v(t)) &= f_1(\mathbf{x}(t),u(t)) + f_2(\mathbf{x}(t),v(t)) \end{aligned} \tag{57}$$

B. Pontryagin's principle

$V(\mathbf{x}(t))$ is not known at the beginning of the game therefore Eq. (56) cannot directly be used in the derivation of saddle-point strategies. A possibility is to use Theorem 5, given below, which provides a set of necessary conditions for an *open-loop representation* of the solution.

Theorem 5 (PMP): Suppose that the pair $\{\gamma_p^*, \gamma_e^*\}$ provides a saddle-point solution in closed-loop strategies, with $\mathbf{x}^*(t)$ denoting the corresponding state trajectory. Furthermore, let its open-loop representation $\{u^*(t) = \gamma_p(\mathbf{x}^*(t)), v^*(t) = \gamma_e(\mathbf{x}^*(t))\}$ also provide a saddle-point solution (in open-loop polices). Then there exists a costate function $p(\cdot) : [0, t_f] \rightarrow R^n$ such that the following relations are satisfied:

$$\dot{\mathbf{x}}^*(t) = f(\mathbf{x}^*(t), u^*(t), v^*(t)),\ \mathbf{x}^*(0) = \mathbf{x}(t_s) \tag{58}$$

$$\begin{aligned} H(p(t), \mathbf{x}^*(t), u^*(t), v(t)) &\leq H(p(t), \mathbf{x}^*(t), u^*(t), v^*(t)) \\ &\leq H(p(t), \mathbf{x}^*(t), u(t), v^*(t)) \end{aligned} \tag{59}$$

$$\dot{p}^T(t) = -\frac{\partial}{\partial x} H(p(t), \mathbf{x}^*(t), u^*(t), v^*(t)) \tag{60}$$

$$p^T(t_f) = \frac{\partial}{\partial x} G(\mathbf{x}^*(t_f)) \text{ along } \zeta(\mathbf{x}^*(t)) = 0 \tag{61}$$

where

$$H(p(t), \mathbf{x}(t), u(t), v(t)) = p^T(t) \cdot f(\mathbf{x}(t), u(t), v(t)) + 1 \tag{62}$$

and T denotes the transpose operator.

Again, a sketch of the proof for this theorem is in [2] (pp. 428-429). Equation (60) is known as the *adjoint equation*, and Eq. (62) as the *Hamiltonian function*. Using Eq. (62) with $p(t) = \nabla V(\mathbf{x}(t))$ for the case of vector-valued functions, and assuming that the Hamiltonian is separable in $u(t)$ and $v(t)$ (refer to Lemma 2), we can rewrite Eq. (56) as

$$\begin{aligned} \min_{u(t)\in\widehat{U}} \max_{v(t)\in\widehat{V}} H(\mathbf{x}(t), \nabla V(\mathbf{x}(t)), u(t), v(t)) &= 0 \\ u^*(t) = \arg\min_{u(t)\in\widehat{U}} H(\mathbf{x}(t), \nabla V(\mathbf{x}(t)), u(t), v(t)) \\ v^*(t) = \arg\max_{v(t)\in\widehat{V}} H(\mathbf{x}(t), \nabla V(\mathbf{x}(t)), u(t), v(t)) \end{aligned} \tag{63}$$

where $u^*(t)$ and $v^*(t)$ are the optimal controls. The vector $\nabla V(\mathbf{x}(t))$ can be interpreted as the Lagrange multipliers used in constrained optimization or optimal control theory.

The maximum principle, in particular Eqs. (59) and (60), can be considered as a specialization of the HJB equation which corresponds to the application of the optimal actions $u^*(t)$ and $v^*(t)$. This causes the $\min\max$ to disappear, but along with it the global properties of the HJB equation also vanish. The PMP expresses conditions along the optimal trajectory, as opposed to the value of the game $V(\mathbf{x}(t))$ over the whole state space. Therefore, it can at best assure local optimality in the space of possible trajectories [19]. In the PMP methodology, the optimal controls for the players are functions of $p(t) = \nabla V(\mathbf{x}(t))$, it is important to note that at moment $u^*(t)$ and $v^*(t)$ are chosen the relation with the state $\mathbf{x}(t)$ is lost. That is the reason we denote $p(t)$ and not $p(\mathbf{x}(t))$. Later, the optimal motion trajectories of the players are constructed using $u^*(t)$ and $v^*(t)$. Therefore, the resulting optimal trajectories are not directly related with the state. However, it is possible to find this relation using the synthesis as it was described on the paper.

REFERENCES

[1] R. Isaacs. *Differential Games*. Wiley, New York, 1965.

[2] T. Başar and G. Olsder, *Dynamic Noncooperative Game Theory, 2nd Ed.* SIAM Series in Classics in Applied Mathematics, Philadelphia, 1999.

[3] L. Guibas, J.-C. Latombe,S.M. LaValle, D. Lin, R. Motwani, Visibility-based pursuit-evasion in a polygonal environment, *Int. J. Comput. Geom. Appl.*, 9(5):471–494, 1999.

[4] V. Isler, S. Kannan and S. Khanna, Randomized Pursuit-Evasion in a Polygonal Environment. *IEEE Trans. Robot.*, 5(21):864–875, 2005.

[5] B. Tovar and S. M. LaValle: Visibility-based Pursuit - Evasion with Bounded Speed. *I. J. Robotic Res*, 27(11-12): 1350-1360, 2008.

[6] I. Suzuki and M. Yamashita Searching for a mobile intruder in a polygonal region. *SIAM J. Comput*, 21(5):863-888, 1992.

[7] R. Vidal, O. Shakernia, H.Jin, D. Hyunchul and S. Sastry, Probabilistic Pursuit-Evasion Games: Theory, Implementation, and Experimental Evaluation, *IEEE Trans. Robot. Autom.*, 18(5):662-669, October, 2002.

[8] J. Hespanha, M. Prandini, and S. Sastry, Probabilistic Pursuit-Evasion Games: A one-step Nash approach, *In Proc. IEEE Conf. Decision, Control*, 2000.

[9] T. H. Chung, On Probabilistic Search Decisions under Searcher Motion Constraints, *In Proc. Int. WAFR* , 2008

[10] T. Chung, G. Hollinger and V. Isler, Search and pursuit-evasion in mobile robotics: A survey, Auton Robot, vol 31, No 4, pp 299-316, 2011.

[11] S.M. LaValle, H.H. González-Baños, C. Becker and J.-C. Latombe, Motion Strategies for Maintaining Visibility of a Moving Target *In Proc. IEEE Int. Conf. Robot. Autom.*, 1997.

[12] H.H. González, C.-Y. Lee and J.-C. Latombe, Real-Time Combinatorial Tracking of a Target Moving Unpredictably Among Obstacles, *In Proc IEEE Int. Conf. Robot. Autom.*, 2002.

[13] B. Jung and G. Sukhatme, Tracking targets using multiple robots: the effect of environment occlusion. In *Auton Robot*, vol. 12 pp. 191-205, 2002.

[14] S. Bhattacharya and S. Hutchinson, On the existence of nash equilibrium for a two player pursuit-evasion game with visibility constraints. *Int. J Robot Res*, 29(7): 831-839, Jun. 2010.

[15] R. Murrieta-Cid, T. Muppirala, A. Sarmiento, S. Bhattacharya and S. Hutchinson. Surveillance Strategies for a Pursuer with Finite Sensor Range. *Int J Robot Res*, Vol 26, No 3 pp. 233-253, March 2007.

[16] Merz A.W. The homicidal chauffeur – a differential game. *PhD. Thesis*, Stanford University, 1971.

[17] R. Murrieta-Cid, U. Ruiz, J.L. Marroquin, J.P. Laumond and S. Hutchinson, Tracking an Omnidirectional Evader with a Differential Drive Robot, *Special Issue on Search and Pursuit/Evasion, Auton Robot*, 31(4): 345-366, 2011.

[18] U. Ruiz and R. Murrieta-Cid, A Homicidal Differential Drive Robot, *In Proc. IEEE Int. Conf. Robot. Autom*, 2012.

[19] S.M. LaValle, *Planning Algorithms*, Cambridge University Press, 2006

[20] D.J. Balkcom and M.T. Mason, Time Optimal Trajectories for Bounded Velocity Differential Drive Vehicles, *Int J Robot Res Vol* 21, No 3, pp 219-232, 2002.

[21] P. Soueres and J.P. Laumond, Shortest paths synthesis for a car-like robot. *IEEE Trans. Autom. Control*, Vol. 41, No. 5, pp. 672-688, 1996.

[22] H. Wang, Y. Chen and P. Soueres. A geometric algorithm to compute time-optimal trajectories for a bidirectional steered robot. *IEEE Transactions on Robotics*, Vol. 25, No. 2, 399-413, 2009.

[23] R. Murrieta-Cid, B. Tovar and S. Hutchinson, A Sampling-Based Motion Planning Approach to Maintain Visibility of Unpredictable Targets, *Auton Robot*, Vol. 19. No 3 pages 285-300, December 2005.

[24] A. Efrat, H. H. Gonzalez, S. G. Kobourov and L. Palaniappan, Optimal Motion Strategies to Track and Capture a Predictable Target, *In Proc. IEEE Int. Conf. Robot. Autom.*, 2003.

[25] J. M. O'Kane, On the value of ignorance: Balancing tracking and privacy using a two-bit sensor. *In Proc. Int. WAFR*, 2008

[26] Tirthankar Bandyopadhyay, Marcelo H. Ang Jr. and David Hsu, Motion planning for 3-D target tracking among obstacles, *Int. Symp. on Robotics Research, 2007*

[27] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes.* JohnWiley, 1962.

[28] I. M. Mitchell, A. M. Bayen and C.J. Tomlin, A Time-Dependent Hamilton-Jacobi Formulation of Reachable Sets for Continuous Dynamics Games. *IEEE Trans. Autom. Control*, Vol. 50, No. 7, pp. 947-957, 2005.

[29] M. Gómez Plaza, T. Martínez-Marín, S. Sánchez Prieto and D. Meziat Luna. Integration of Cell-Mapping and Reinforcement-Learning Techniques for Motion Planning of Car-Like Robots. *IEEE Transactions on Instrumental and Measurement*, Vol. 58, No. 9, 2009

[30] M. Gómez, R.V. González, T. Martínez-Marín, D. Meziat and S. Sánchez, Optimal motion planning by reinforcement learning in autonomous mobile vehicles, *Robotica*, Vol. 30, pp. 158-170, 2012

[31] L. Parker. Algorithms for Multi-Robot Observation of Multiple Targets. In *Auton Robot*, vol. 12 pp. 231-255, 2002.

[32] D. Bhadauria and V. Isler, Capturing an Evader in a Polygonal Environment with Obstacles, *In Proc. Int. Joint Conf. Artif. Intell.*, 2011

[33] O. Tekdas, W. Yang and V. Isler, Robotic Routers: Algorithms and Implementation, *Int. J Robot Res, Vol* 29, No 1, pp 110-126, 2010.

[34] A.A. Melikyan, Generalized Characteristics of First Order PDEs: Applications in Optimal Control and Differential Games, Birkhauser, 2000