Received 17 July 2022, accepted 1 August 2022, date of publication 4 August 2022, date of current version 9 August 2022.
Digital Object Identifier 10.1109/ACCESS.2022.3196342

RESEARCH ARTICLE

Capturing a Dubins Car With a Differential Drive Robot

UBALDO RUIZ
Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, Baja California 22860, Mexico
Consejo Nacional de Ciencia y Tecnología, Ciudad de México 03840, Mexico
e-mail: uruiz@cicese.mx

This work was supported in part by Catedras-CONACYT under Project 1850, and in part by CONACYT under Grant A1-S-21934.

ABSTRACT In this paper, we study the problem of capturing a Dubins Car with a Differential Drive Robot in minimum time. The two vehicles are represented like unitary discs moving in the plane without obstacles. Both agents have the same maximum speed and a bounded turning ratio. We frame the problem as a pursuit-evasion game. The Differential Drive Robot plays as a pursuer and aims to capture the Dubins Car as soon as possible. The Dubins Car, on the contrary, takes the evader's role and tries to avoid capture. Using differential game theory, we compute the players' time-optimal motion strategies to accomplish their tasks and provide analytical expressions describing them. In particular, we reveal four singular surfaces in this game. Two evader's dispersal surfaces (EDS) where the evader can choose between two controls and the pursuer must react accordingly, leading to trajectories with the same cost. One pursuer's dispersal surface (PDS) where the evader must select its control based on the pursuer's choice. And a transition surface (TS), where the DDR switches its controls. Some examples of the players' time-optimal motion strategies are shown in numerical simulations.

INDEX TERMS Differential games, optimal control, pursuit-evasion, robotics.

I. INTRODUCTION

This paper studies a pursuit-evasion problem [1]–[3] between two antagonistic agents. In the literature, one can identify different formulations of pursuit-evasion games [4]–[8]. The main differences are usually the players' goals, motion constraints, and sensing capabilities. There are different practical applications of pursuit-evasion problems. Some examples are 1) a robotic system whose task is to capture or keep surveillance of an independent malicious agent, 2) a convoy of autonomous vehicles following a leader, 3) a group of robotic routers moving in an environment to maintain connectivity between a mobile user and a base station, and 4) a robotic system helping a person as she moves in an environment, for example, carrying luggage in an airport.

In particular, this work addresses the problem of capturing a Dubins Car (DC) with a Differential Drive Robot (DDR) in minimum time. Both players move in a plane without obstacles and are modeled like unitary discs. The DDR plays the role of the pursuer, and the DC the role of the evader. The DDR's goal is to capture the DC as soon as possible. On the contrary, the DC wants to delay capture as much as possible.

In this work, we model the above problem as a zero-sum game and compute the players' time-optimal motion strategies [3]. Additionally, we characterize regions of the playing space containing initial configurations that lead to capture. To the best of our knowledge, this is the first work addressing a pursuit-evasion problem between a DDR and DC and presents an analysis based on differential game theory [1], [3].

One of the main contributions of this work is that it considers two of the most popular non-holonomic mobile vehicles in robotics [9], a DDR and a DC. The time-optimal motion primitives of a DDR and a DC moving in a plane without obstacles were obtained in [10], and [11], respectively. In both cases, the motion primitives are computed using optimal control theory [3], [12]. However, different from our problem in which we consider two players moving simultaneously and interacting between them, the analysis made in [10], [11] is focused on a single type of vehicle having the goal of

The associate editor coordinating the review of this manuscript and approving it for publication was Liang Hu.

reaching particular configurations in the plane. Considering two non-holonomic vehicles in a pursuit-evasion game makes the problem harder to model and analyze compared to existing works in pursuit-evasion games [13]–[16]. The game of capture involving two DCs as players was studied in [17] and the corresponding version involving two DDRs in [18]. However, different from those previous works, in our game, the evader is a DC, and the pursuer is DDR, each with a different kinematic model. That change has a consequence that the time-optimal motion strategies of our players and those in [17], [18] are distinct, motivating our study of the proposed problem.

We compute the solution of our pursuit-evasion problem following the approach developed by Isaacs [1], Lewin [19]. Isaacs' methodology is based on partitioning the playing space into regions, where the value function is differentiable. Generally, the more difficult part of the process is finding the boundaries of those regions, known as *singular surfaces*, since there is no knowledge ahead of time if the optimal trajectories have singular portions or not. Usually, one assumes the existence of a singular surface and tributary trajectories joining it only when the regular backward construction of candidate trajectories does not cover the entire playing space [19].

To characterize a singular surface and the corresponding outcome, one player must often base his choice of controls based on the knowledge of his opponent's control selection. If a strategy is computed using this information is called a non-admissible strategy in the context of differential game theory. In contrast, an admissible strategy does not require additional information on the players' controls and is based only on knowledge of the system's state. We characterize the singular surfaces appearing in the solution and the trajectories filling the regions defined by them in our problem. The trajectories inside a region correspond to admissible strategies for the players.

For some specific game models, using the methodology developed by Isaacs allows computing closed-form solutions [1], [13]–[16]. In our work, we succeed in finding mathematical equations describing the players' time-optimal motion strategies.

II. RELATED WORK

Pursuit-evasion problems can be classified into three main variants. In the first one, the pursuers have the task of finding an evader in an environment [20]–[22]. In the second variant, the goal consists in maintaining the visibility of a moving evader [23]–[25]. Finally, in the third variant, like in this work, the pursuer has the task of capturing a moving evader, i.e., moving a to contact configuration or closer than a given distance to the pursuer [1], [14], [18].

The problem studied in this work belongs to a class of problems known as differential games [1]–[3], [19]. In the literature, we can find many works that have studied differential games in the past [1], [4], [13]–[16], [18]. Probably, the most widely recognized of them is the Homicidal Chauffeur problem [13]. In that problem, a car wants to run over a pedestrian as soon as possible. The pedestrian, on the other side, wants to avoid it. The car is quicker than the pedestrian but has a turning ratio constraint. The pedestrian is more agile than the car since he can change its motion direction instantaneously. The players are located in a parking lot without obstacles. The solution to this problem consists of finding the players' motion strategies to accomplish their goals and deciding the winner of the game based on the system's initial configurations.

To the best of our knowledge, the most related problems to this work in the differential games' literature are the following: [14], [15], [17], [18]. In this section, we present a comparison between the current work and [14], [15], [17], [18].

The problem of capturing an Omnidirectional Agent (OA) with a DDR was addressed in [14]. In that work, similar to the Homicidal Chauffeur problem, the OA is slower than the DDR but can instantaneously change its motion direction. The goal of the DDR is to capture the OA as soon as possible. The OA wants to delay it. The authors computed the players' time-optimal motion strategies. Also, based on the players' initial configurations, they can decide the winner of the game, i.e., if the DDR captures the OA in finite time or not. In [15], the authors studied the analogous problem in which the players switch roles, i.e., the OA's goal is to capture the DDR in minimum time. The DDR wants to avoid it.

In this work, similar to those two previous works, we also consider a DDR. However, the evader in our case is a Dubins car, which has non-holonomic constraints, that change has as a consequence that the players' motion strategies and the nature of the solution differ from those in [14], [15]. Another fundamental difference is that our game requires a representation in a higher-dimensional space which makes the solution and its analysis harder to compute.

As in this paper, two related works considering non-holonomic players are [17] and [18]. [17] studies a game of capture in minimum time where the pursuer and the evader are two identical Dubins cars. The authors found the time-optimal motion strategies for both players. More recently, [26] presents a comprehensive solution to that problem, and [27] provides feedback-based solutions for particular cases. Note that the pursuer has different kinematic capabilities than in our case, where the pursuer is a DDR. That implies that the players' motion strategies found in [17], [26], [27] differ from those in our work. [18] also analyzes a pursuit-evasion problem between two identical DDRs. As in our work, the pursuer is a DDR and wants to capture the evader in minimum time; however, the evader is also a DDR, which, as was mentioned earlier, has a different kinematic model than a Dubins car. Thus, the players' time-optimal motion strategies and trajectories also differ from our work.

Finally, it is important to remark that to the best of our knowledge, the setting addressed in our work, a pursuit-evasion game of capture in minimum time between a Dubins Car and a Differential Drive Robot, has not been reported before in the literature. Thus, our solution is entirely novel, and we cannot make a direct comparison with other previous works addressing the same problem. A significant feature of our solution is that it was constructed using tools from differential game theory, a widely used methodology having well-established mathematical foundations.

A. CONTRIBUTIONS

The main contributions of our work are the following.

- We reveal four singular surfaces in this game, two evader's dispersal surfaces (EDS), one pursuer's dispersal surface, and a transition surface where the DDR switches its controls.
- We characterize regions of the playing space containing initial configurations that lead to capture, and we indicate the players' time-optimal motion strategies at each one.
- We provide some numerical simulations of the players' time-optimal motion strategies in a local reference frame attached to the DDR and in a global frame.

III. PROBLEM DEFINITION

A Differential Drive Robot (DDR) and a Dubins Car (DC) move in a plane without obstacles. The DDR (pursuer) tries to capture the DC (evader) as soon as possible. On the other hand, the DC tries to avoid capture. We model both players like unitary discs, and we consider that they have the same maximum speed of $V^{\max}$; thus, the game's outcome is determined by the specific non-holonomic constraints of the players. The game is over when the players' distance is smaller than a value l_c. Any effects due to dynamic constraints (e.g., acceleration bounds) are neglected, and only a purely kinematic problem is considered. We find regions of the playing space (initial configurations) where the DDR captures the DC when both players apply their time-optimal motion strategies.

A. REALISTIC SPACE REPRESENTATION

Similar to previous works [14], our problem can be modeled in the Euclidean plane (see Fig. 1a). Let (x_p, y_p, θ_p) denote the DDR's pose and (x_e, y_e, θ_e) represent the DC's pose. The state of the system can be expressed as $(x_p, y_p, \theta_p, x_e, y_e, \theta_e) \in \mathbb{R}^2 \times S^1 \times \mathbb{R}^2 \times S^1$. That representation is called the *realistic space*. The time evolution of the system in the realistic space is given by the following equations of motion

$$\begin{aligned} \dot{x}_p &= \left(\frac{u_1+u_2}{2}\right)\cos\theta_p, \quad \dot{y}_p = \left(\frac{u_1+u_2}{2}\right)\sin\theta_p, \\ \dot{\theta}_p &= \left(\frac{u_2-u_1}{2b}\right) \\ \dot{x}_e &= V^{\max}\cos\theta_e, \quad \dot{y}_e = V^{\max}\sin\theta_e, \\ \dot{\theta}_e &= \frac{V^{\max}}{r_e}v \end{aligned} \tag{1}$$

where $v \in [-1, 1]$ is the control of the DC, and r_e is the DC's maximum turning radius. $u_1, u_2 \in [-V^{\max}, V^{\max}]$ are the DDR's controls, and they correspond to the left and right wheel's velocities, respectively. b denotes the distance between the DDR's center and the wheels' location. We assume both wheels have an unitary radius, thus, the rotational and traslational speeds are equivalent. The previous equations can be expressed in the form $\dot{\mathbf{x}} = f(\mathbf{x}, u, v)$ where $u = (u_1, u_2) \in [-V^{\max}, V^{\max}] \times [-V^{\max}, V^{\max}]$ and $v \in [-1, 1]$. In this representation, all angles are measured in counter-clockwise sense from the x-axis.

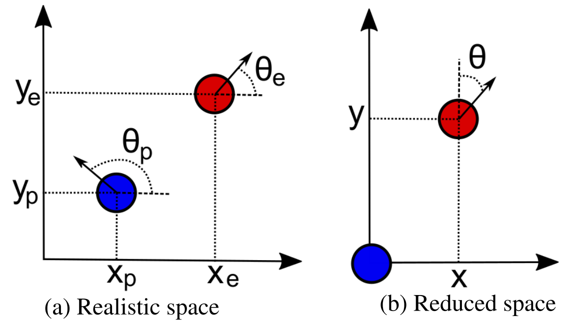


FIGURE 1. The DDR (pursuer) is represented by the blue disc and the DC (evader) by the red disc.

B. REDUCED SPACE REPRESENTATION

As customary in differential game theory, the problem analysis can be simplified using a coordinate system mounted on the pursuer. The reference frame is fixed to the DDR's body (see Fig. 1b). The system's state is described by $\mathbf{x} = (x, y, \theta)$, which corresponds to the DC's pose relative to the DDR's body. That representation is called the *reduced space*. All orientations are measured with respect to the positive y-axis in a clockwise sense. Using the following coordinate transformation between the reduced and realistic space

$$\begin{aligned} x &= (x_e - x_p)\sin\theta_p - (y_e - y_p)\cos\theta_p \\ y &= (x_e - x_p)\cos\theta_p + (y_e - y_p)\sin\theta_p \\ \theta &= \theta_p - \theta_e \end{aligned} \tag{2}$$

we obtain the motion equations in the DDR-fixed coordinate system

$$\begin{aligned} \dot{x} &= \left(\frac{u_2-u_1}{2b}\right)y + V^{\max}\sin\theta \\ \dot{y} &= -\left(\frac{u_2-u_1}{2b}\right)x - \left(\frac{u_1+u_2}{2}\right) + V^{\max}\cos\theta \\ \dot{\theta} &= \left(\frac{u_2-u_1}{2b}\right) - \frac{V^{\max}}{r_e}v \end{aligned} \tag{3}$$

where again $v \in [-1, 1]$ is the DC's control. For the DDR, we have that $u_1, u_2 \in [-V^{\max}, V^{\max}]$. We can express the previous equations in the form $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{v})$, where $\mathbf{x} = (x, y)$, $\mathbf{u} = (u_1, u_2)$ and $\mathbf{v} = (v)$.

IV. TERMINAL CONDITIONS

Following Isaacs' methodology [1], we compute the set of configurations where the DDR guarantees termination

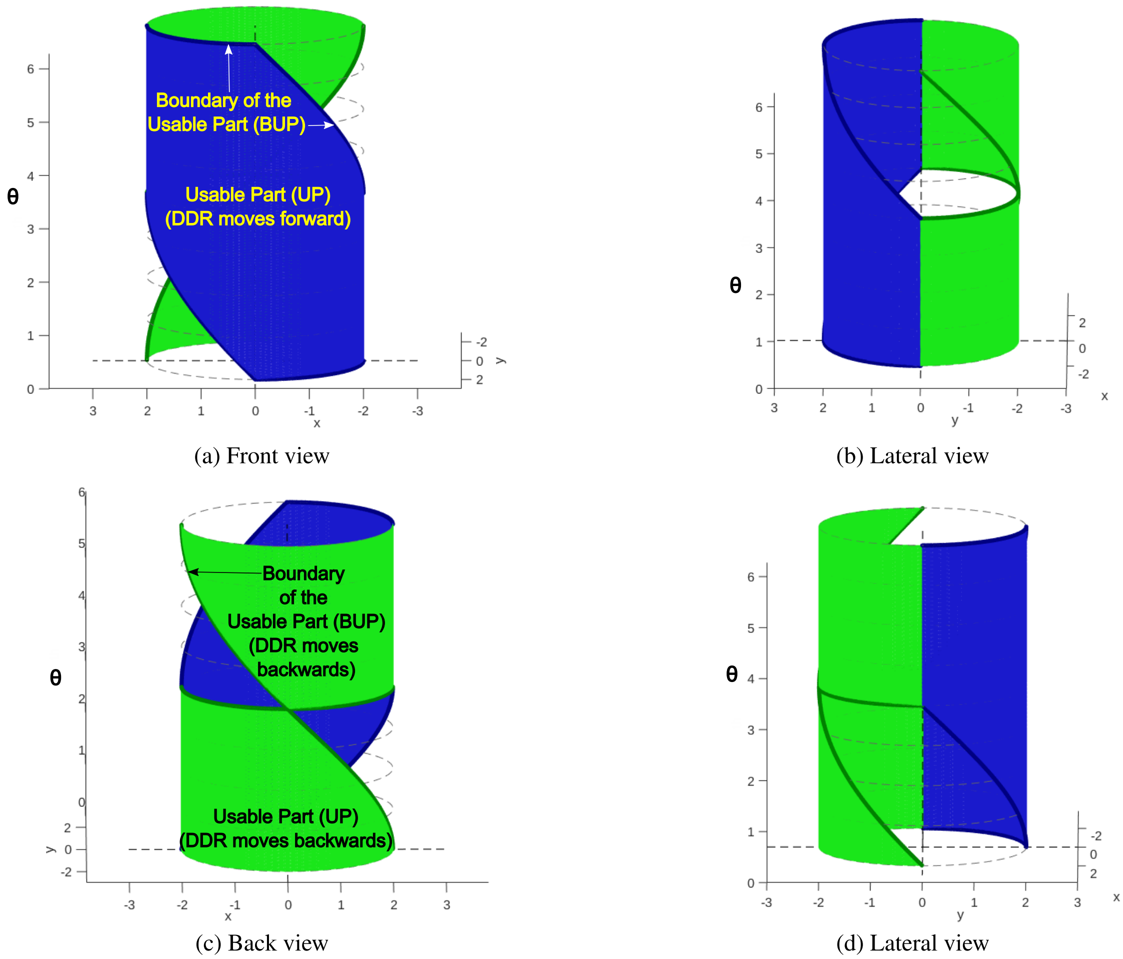


FIGURE 2. Representation of the Usable Part (UP) and its boundary (BUP). Four views have been included to illustrate the general structure. The regions in blue correspond to configurations where the DDR ends the game (captures the DC) moving forward. The dark blue curves indicate its boundaries. Similarly, the green regions correspond to configurations where the capture occurs while the DDR moves backward. Analogous, the dark green curves show its boundaries. Note that some regions of the surface of the cylinder are not colored. Despite having a distance l_c between both players, capture in those regions is impossible for the DDR since the DC could use a control to increase or at least keep a constant distance to the DDR.

(capture) regardless of the choice of controls of the DC. This set is known as the usable part (UP). For this problem, the DDR captures the DC when the distance between both players is smaller than the capture distance l_c despite any DC's opposition. In the reduced space, the terminal surface ζ can be represented as a cylinder of radius l_c centered at the origin with height 2π. We can parametrize ζ by two angles ϕ and ψ. ϕ is the angle between the DC's position and the DDR's heading, and ψ is the angle between the headings of both players. Let l denotes the distance between the DDR and the DC. In the reduced space, the DDR guarantees capture when $l = l_c$ and $\dot{l} < 0$ the UP is given by

$$x = l_c \sin\phi, \quad y = l_c \cos\phi, \quad \theta = \psi, \quad l^2 = x^2 + y^2, \tag{4}$$

We have that the game ends when

$$\min_{u_1,u_2} \max_{v} \dot{l} < 0 \tag{5}$$

Computing the time derivative for l and substituting (3) into the resulting expression, we have that

$$\begin{aligned}\dot{l} &= V^{\max}(\sin\phi\sin\psi + \cos\phi\cos\psi) - \left(\frac{u_1+u_2}{2}\right)\cos\phi \\ &= V^{\max}\cos(\phi-\psi) - \left(\frac{u_1+u_2}{2}\right)\cos\phi \end{aligned} \tag{6}$$

Applying the optimal controls for both players at the UP, we obtain

$$\min_{u_1,u_2} \max_{v} \dot{l} = V^{\max}\cos(\phi-\psi) - V^{\max}|\cos\phi| \tag{7}$$

thus, from (5)

$$\mathrm{UP} = \{\phi, \psi \mid |\cos\phi| > \cos(\phi-\psi)\} \tag{8}$$

The boundary of the usable part (BUP) is defined by

$$\min_{u_1,u_2} \max_{v} \dot{l} = 0 \tag{9}$$

thus,

$$\mathrm{BUP} = \{\phi, \psi \mid |\cos\phi| = \cos(\phi - \psi)\} \tag{10}$$

From (7), we can observe that the DDR performs a translation at maximum speed when it captures the DC. The DDR moves forward when $\cos\phi > 0$ and it moves backward when $\cos\phi < 0$. Also, note that both the UP and BUP do not depend on the DC's control v. Fig. 2 shows a graphical representation of the UP and its boundary in our problem.

V. OVERVIEW OF THE SOLUTION

The goal of this section is to provide a summary of the problem's solution that helps guide the reader during the identification and computation of the players' motion strategies later in the paper.

For this game, we found a characterization of the reduced space containing initial configurations that lead to capture into two regions, each corresponding to particular time-optimal motion strategies for the players. Those regions are bounded by two types of singular surfaces: transition and dispersal surfaces. In particular, we found four singular surfaces: two evader's dispersal surfaces (EDS), one pursuer's dispersal surface (PDS), and a transition surface (TS) where the DDR switches its controls. Fig. 3 shows the characterization of the reduced space.

On a dispersal surface (DS), one of the players can choose between two controls that end in two trajectories with the same cost. According to [19], this player dominates the singular surface. The second player must react according to the choice of the first one to avoid favoring that player. To do that, it must implement a non-admissible strategy. In Fig. 3b, Fig. 3d, Fig. 3e and Fig. 3f we can observe examples of the three dispersal surfaces we have found in this work. The EDS in Fig. 3b is formed by some primary trajectories intersecting the y-axis in the x, y-plane. On that dispersal surface, the evader has two choices for its optimal control; however, the associated trajectories to each control have the same cost. The upper trajectories correspond to a forward translation for the DDR when it captures the DC and the lower ones to a backward translation. Fig. 3d shows a PDS; in this case, the DC position is perpendicular to the DDR's heading. Note that for those configurations, the DDR is forced to rotate and align its heading with the DC's position to achieve capture. For the DDR, both trajectories have the same cost, i.e., either forcing the DC to rotate clockwise or counterclockwise. The evader must choose its control according to the DDR's decision. Finally, a third EDS surface is shown in Fig. 3e and Fig. 3f; similar to the first surface, this EDS is formed by the intersection of the primary trajectories. In this case, the intersections occur close to the yellow curve.

Fig. 3b and Fig. 3d also show that some of the trajectories associated with the primary solution do not reach a DS, defining a TS (the boundary between red and yellow trajectories). The DDR switches its controls on the TS and starts rotating in place.

In the following paragraphs, we define the properties of each region found in this work.

- **Region I** corresponds to the primary solution of the game. The trajectories in this region (red curves) reach the UP and correspond to a DDR's translation in place at maximum speed in the realistic space. The DC, on the other side, translates and rotates at maximum speed.
- **Region II** contains the trajectories (yellow curves) leaving the transition surface in retro-time. In all cases, the trajectories correspond to a DDR's rotation in place at maximum speed in the realistic space.

VI. MOTION STRATEGIES

In this section, we compute the time-optimal motion strategies of the players. To do that, we perform a retro-time integration of the motion equations starting at the ending configurations.

A. OPTIMAL CONTROLS

To integrate the motion equations, we first need to find the players' optimal controls used to accomplish their goals. Following the approach in [1], we construct the Hamiltonian of the system given by

$$H(\mathbf{x}, \lambda, \mathbf{u}, \mathbf{v}) = \lambda^T \cdot f(\mathbf{x}, \mathbf{u}, \mathbf{v}) + L(\mathbf{x}, \mathbf{u}, \mathbf{v}) \tag{11}$$

where λ^T are the costate variables and $L(\mathbf{x}, \mathbf{u}, \mathbf{v})$ is the cost function. For problems of minimum time, as the one studied in this work, $L(\mathbf{x}, \mathbf{u}, \mathbf{v}) = 1$. Thus, we have that

$$\begin{aligned} &H(\mathbf{x}, \lambda, \mathbf{u}, \mathbf{v}) \\ &= \lambda_x \left(\frac{u_2 - u_1}{2b}\right) y + \lambda_x V^{\max} \sin\theta \\ &\quad - \lambda_y \left(\frac{u_2 - u_1}{2b}\right) x - \lambda_y \left(\frac{u_1 + u_2}{2}\right) + \lambda_y V^{\max} \cos\theta \\ &\quad + \lambda_\theta \left(\frac{u_2 - u_1}{2b}\right) - \lambda_\theta \left(\frac{V^{\max}}{r_e}\right) v + 1 \end{aligned} \tag{12}$$

where $\lambda^T = (\lambda_x, \lambda_y, \lambda_\theta)$. For problems of minimum time [1]

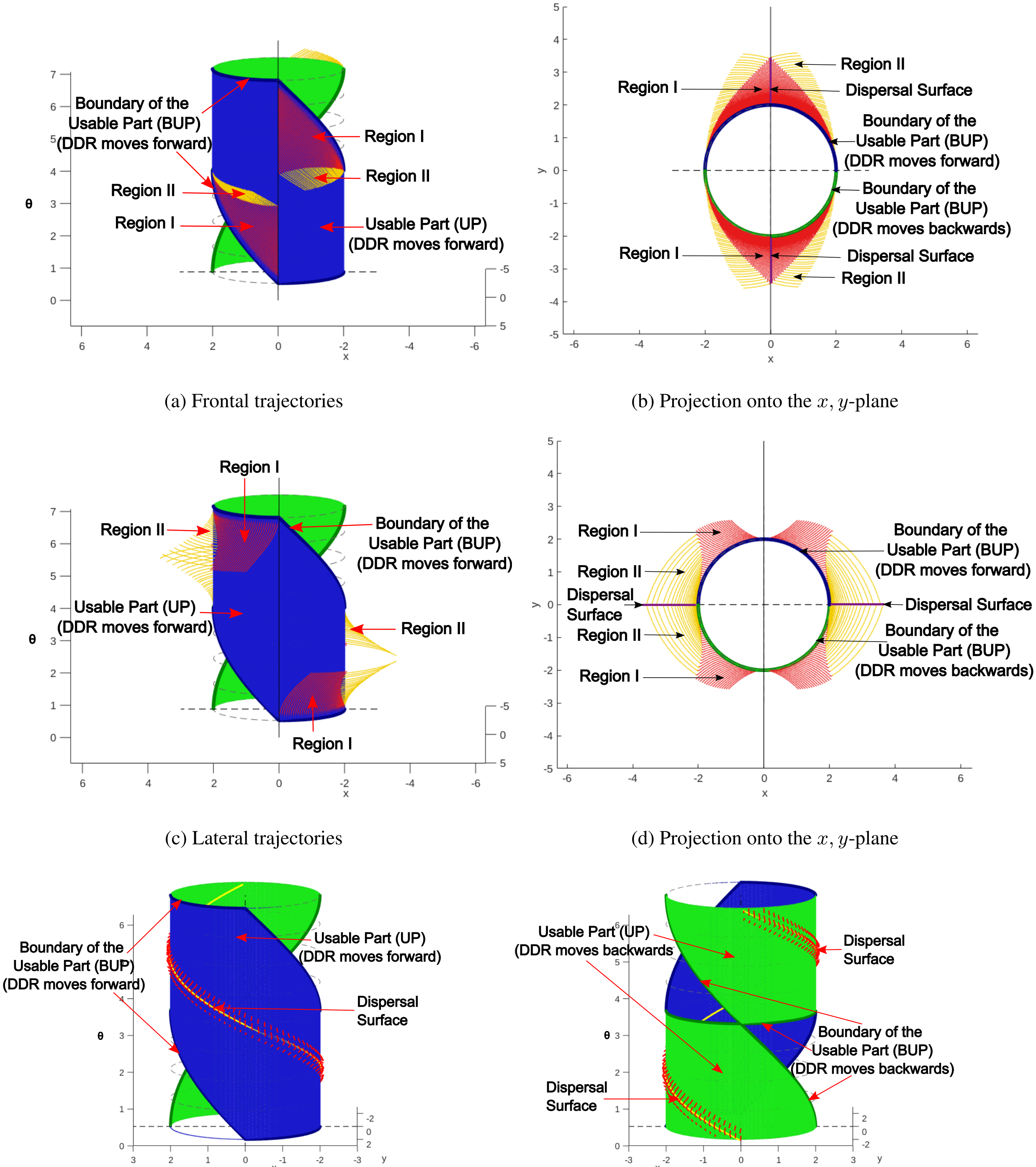
$$\begin{aligned} &\min_{\mathbf{u}} \max_{\mathbf{v}} H(\mathbf{x}, \lambda, \mathbf{u}, \mathbf{v}) = 0 \\ &\mathbf{u}^* = \arg\min_{\mathbf{u}} H(\mathbf{x}, \lambda, \mathbf{u}, \mathbf{v}) \\ &\mathbf{v}^* = \arg\max_{\mathbf{v}} H(\mathbf{x}, \lambda, \mathbf{u}, \mathbf{v}) \end{aligned} \tag{13}$$

where $\mathbf{u}^*$ and $\mathbf{v}^*$ denote the players' optimal controls. From (12) and (13), we have that the DDR's controls are given by

$$\begin{aligned} u_1^* &= -\mathrm{sgn}\left(\frac{-\lambda_x y}{2b} + \frac{\lambda_y x}{2b} - \frac{\lambda_y}{2} - \frac{\lambda_\theta}{2b}\right) V^{\max} \\ u_2^* &= -\mathrm{sgn}\left(\frac{\lambda_x y}{2b} - \frac{\lambda_y x}{2b} - \frac{\lambda_y}{2} + \frac{\lambda_\theta}{2b}\right) V^{\max} \end{aligned} \tag{14}$$

and the DC's control is given by

$$v^* = \mathrm{sgn}\left(-\lambda_\theta\right) \tag{15}$$



(a) Frontal trajectories

(b) Projection onto the x, y-plane

(c) Lateral trajectories

(d) Projection onto the x, y-plane

(e) Dispersal surface - Front view

(f) Dispersal surface - Back view

FIGURE 3. Overview of the solution in the reduced space. The figures in the first row show an example of front and back trajectories appearing in this game and a projection onto the x, y-plane. Similarly, the second row shows an example of the lateral trajectories appearing in the game and the corresponding projection onto the x, y-plane. In the third row, we can observe an Evader's Dispersal Surface (EDS) example.

B. COSTATE EQUATIONS

The players' optimal controls, given by (14) and (15), depend on the values of λ^T as time elapses. Those values can be computed using the costate equations, which are found by taking the Hamiltonian's partial derivatives with respect to the state variables. If t_f is the time of termination of the game, we define the retro-time as $\tau = t_f - t$. In this work, we denote the *retro-time derivative* of a variable x as $\mathring{x}$. The costate equation in its retro-time form is

$$\mathring{\lambda} = \frac{\partial}{\partial \mathbf{x}} H(\mathbf{x}, \lambda, \mathbf{u}^*, \mathbf{v}^*) \tag{16}$$

For our problem, we have that

$$\begin{aligned} \mathring{\lambda}_x &= -\lambda_y \left(\frac{u_2^* - u_1^*}{2b}\right), \quad \mathring{\lambda}_y = \lambda_x \left(\frac{u_2^* - u_1^*}{2b}\right) \\ \mathring{\lambda}_\theta &= V^{\max} \left(\lambda_x \cos\theta - \lambda_y \sin\theta\right) \end{aligned} \tag{17}$$

Note that the costate equations do not explicitly depend on the DC's control for this game.

C. PRIMARY SOLUTION

In this section, we compute the players' retro-time trajectories near the end of the game. First, we need to find the initial conditions of the costate and motion equations in the reduced space. From (4), we have that at the end of game, $x_f = l_c \sin\phi$, $y_f = l_c \cos\phi$, and $\theta = \psi$. Additionally,

$$\begin{aligned} &\frac{\partial x}{\partial \phi} = l_c \cos\phi, \quad \frac{\partial y}{\partial \phi} = -l_c \sin\phi, \quad \frac{\partial \theta}{\partial \phi} = 0 \\ &\frac{\partial x}{\partial \psi} = 0, \quad \frac{\partial y}{\partial \psi} = 0, \quad \frac{\partial \theta}{\partial \psi} = 1 \end{aligned} \tag{18}$$

Since $\lambda(\mathrm{x}) = 0$ on the UP, then λ_ϕ and λ_ψ are given by

$$\begin{aligned} \lambda_\phi &= \frac{\partial \lambda}{\partial \phi} = \frac{\partial \lambda}{\partial x}\frac{\partial x}{\partial \phi} + \frac{\partial \lambda}{\partial y}\frac{\partial y}{\partial \phi} + \frac{\partial \lambda}{\partial \theta}\frac{\partial \theta}{\partial \phi} \\ &= \lambda_x \cos\phi - \lambda_y \sin\phi = 0 \\ \lambda_\psi &= \frac{\partial \lambda}{\partial \psi} = \frac{\partial \lambda}{\partial x}\frac{\partial x}{\partial \psi} + \frac{\partial \lambda}{\partial y}\frac{\partial y}{\partial \psi} + \frac{\partial \lambda}{\partial \theta}\frac{\partial \theta}{\partial \psi} = \lambda_\theta = 0 \end{aligned} \tag{19}$$

From (19), we obtain that

$$\lambda_x \cos\phi = \lambda_y \sin\phi, \quad \lambda_\theta = 0 \tag{20}$$

Therefore, on the UP

$$\lambda_x = \sin\phi, \quad \lambda_y = \cos\phi, \quad \lambda_\theta = 0 \tag{21}$$

From the analysis in Section IV, we know that near the end of the game, the DDR performs a translation at maximum speed. Therefore (17) takes the form

$$\mathring{\lambda}_x = 0, \quad \mathring{\lambda}_y = 0, \quad \mathring{\lambda}_\theta = V^{\max}(\lambda_x \cos\theta - \lambda_y \sin\theta) \tag{22}$$

The solution of the previous equations is given by

$$\begin{aligned} \lambda_x &= \sin\phi, \quad \lambda_y = \cos\phi \\ \lambda_\theta &= \frac{r_e}{v^*}\left(-\cos(\phi - \psi) + \cos(\phi - \psi - \frac{V^{\max}}{r_e} v^* \tau)\right) \end{aligned} \tag{23}$$

Recall that r_e and v^* are the DC's maximum turning radius and the DC's control. ϕ is the angle between the DC's position and the DDR's heading, and the ψ is the angle between the headings of both players, both at the game end. That solution will be valid at the UP and if the DDR's controls do not change. Later, we compute the retro-time instant when the players switch their controls numerically.

Now we proceed to integrate the motion equations in the reduced space. The retro-time version of (3) is given by

$$\begin{aligned} \mathring{x} &= -\left(\frac{u_2 - u_1}{2b}\right) y - V^{\max} \sin\theta \\ \mathring{y} &= \left(\frac{u_2 - u_1}{2b}\right) x + \left(\frac{u_1 + u_2}{2}\right) - V^{\max} \cos\theta \\ \mathring{\theta} &= -\left(\frac{u_2 - u_1}{2b}\right) + \frac{V^{\max}}{r_e} v \end{aligned} \tag{24}$$

Solving (24) with the initial conditions $x_f = l_c \sin\phi$, $y_f = l_c \cos\phi$ and $\theta = \psi$, and the optimal controls u_1^*, u_2^* and v^*, we get

$$\begin{aligned} x &= l_c \sin\phi + \frac{r_e}{v^*}\left(-\cos\psi + \cos(\psi + \frac{V^{\max}}{r_e} v^* \tau)\right) \\ y &= l_c \cos\phi + \frac{r_e}{v^*}\left(\sin\psi - \sin(\psi + \frac{V^{\max}}{r_e} v^* \tau)\right) \\ &\quad + \left(\frac{u_1^* + u_2^*}{2}\right)\tau \\ \theta &= \psi + \frac{V^{\max}}{r_e} v^* \tau \end{aligned} \tag{25}$$

Recall that the values of u_1^*, u_2^* are given by (14) and v^* is given by (15). Note that (25) provides the players' trajectories in the reduced space, thus, to find the corresponding trajectories in the realistic space, we need to apply a coordinate transformation. The trajectories given by (25) are known as the *primary solution*.

D. TRANSITION SURFACE

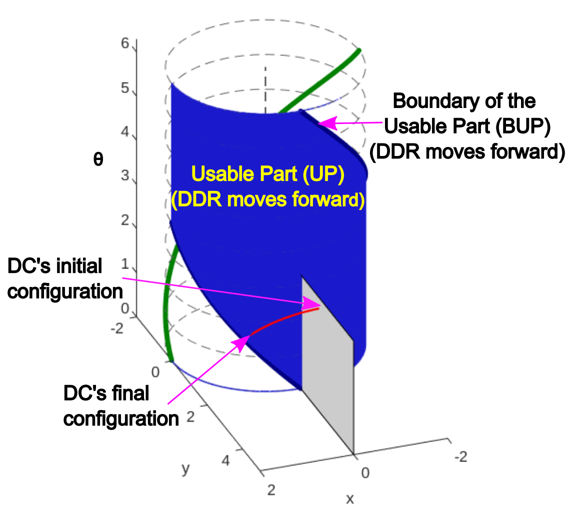
After following some of the primary trajectories in this game, the DDR switches its controls, starting to rotate in place at maximum speed. Unfortunately, given that (25) contains transcendental functions, we cannot find an analytic function to compute the retro-time instant τ_s when that occurs, and we are forced to apply numerical analysis to find its value.

Once τ_s is computed and the new optimal controls u_1^* and u_2^* are obtained using (14), we need to perform a new integration of the costate and motion equations. Considering that the DDR's controls correspond to a rotation in place at maximum speed, the solution of (17) is

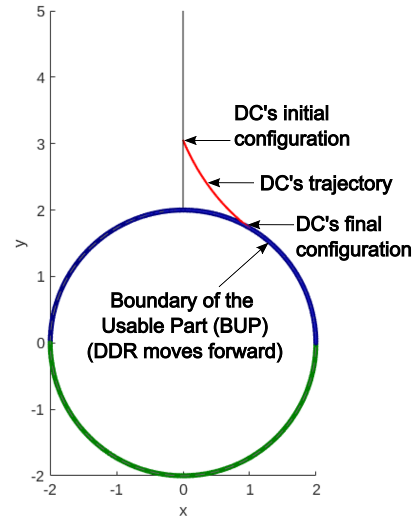
$$\begin{aligned} \lambda_x &= \sin\left(\phi - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right) \\ \lambda_y &= \cos\left(\phi - \left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right) \\ \lambda_\theta &= \frac{r_e}{v^*}\Big(-\cos(\phi - \theta_s) \\ &\quad + \cos(\phi - \theta_s - \frac{V^{\max}}{r_e} v^*(\tau - \tau_s))\Big) + \lambda_{\theta_s} \end{aligned} \tag{26}$$

where λ_{θ_s} is computed substituting τ_s into the third expression in (23), and θ_s is the value of θ at time τ_s obtained from the third expression in (25).

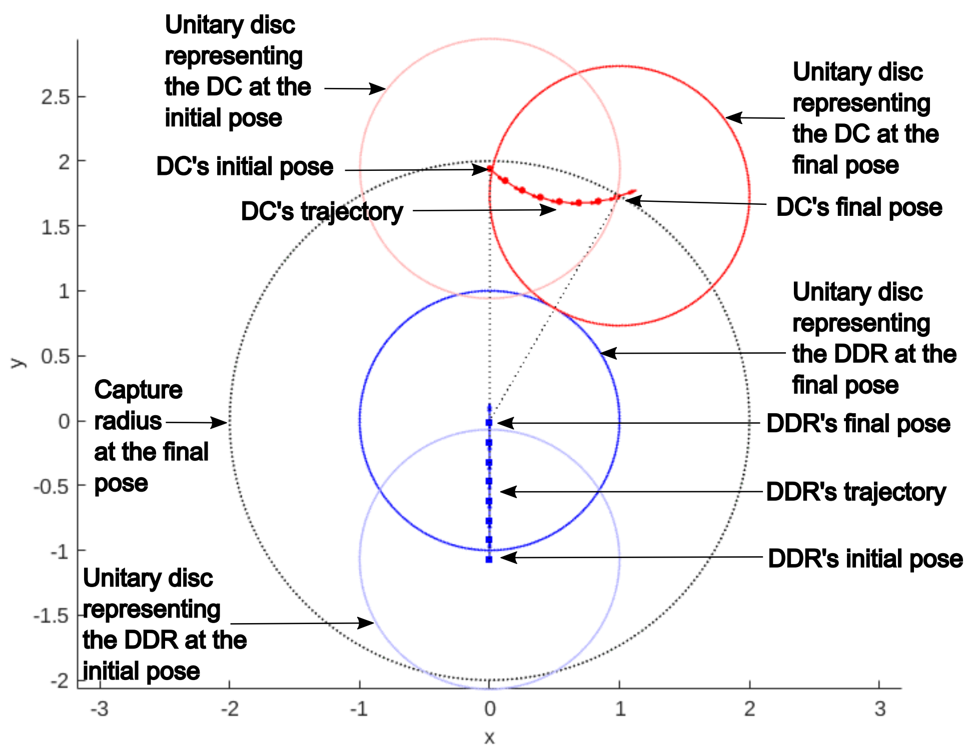
Integrating (24) with x_s, y_s and θ_s (the values of x, y and θ at τ_s), and considering that the DDR rotates in place at



(a) The red curve shows the trajectory followed by the DC in the reduced space.



(b) Projection onto the x, y-plane of the trajectory (red curve) followed by the DC in the reduced space.



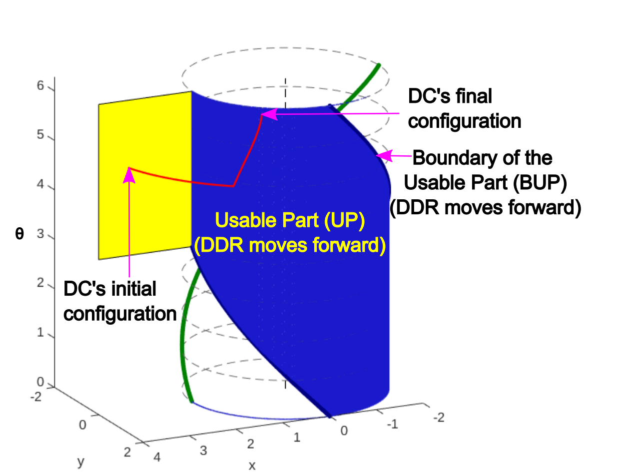
(c) Trajectories of the players in the realistic space. The blue curve indicates the trajectory followed by the DDR, and the red one corresponds to the trajectory followed by the DC. The arrows show the motion directions of the players.

FIGURE 4. Results of the first simulation. The evader starts at the EDS in the reduced space.

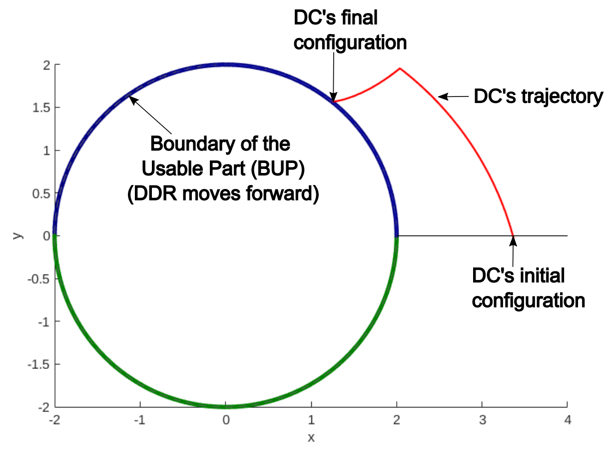
maximum speed, we obtain

$$x = \left(x_s - \frac{r_e}{v^*}\cos\theta_s\right)\cos\left(\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right)$$
$$- \left(y_s + \frac{r_e}{v^*}\sin\theta_s\right)\sin\left(\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right)$$
$$+ \frac{r_e}{v^*}\cos\left(\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s) - \frac{V^{\max}}{r_e}v^*(\tau - \tau_s) - \theta_s\right)$$

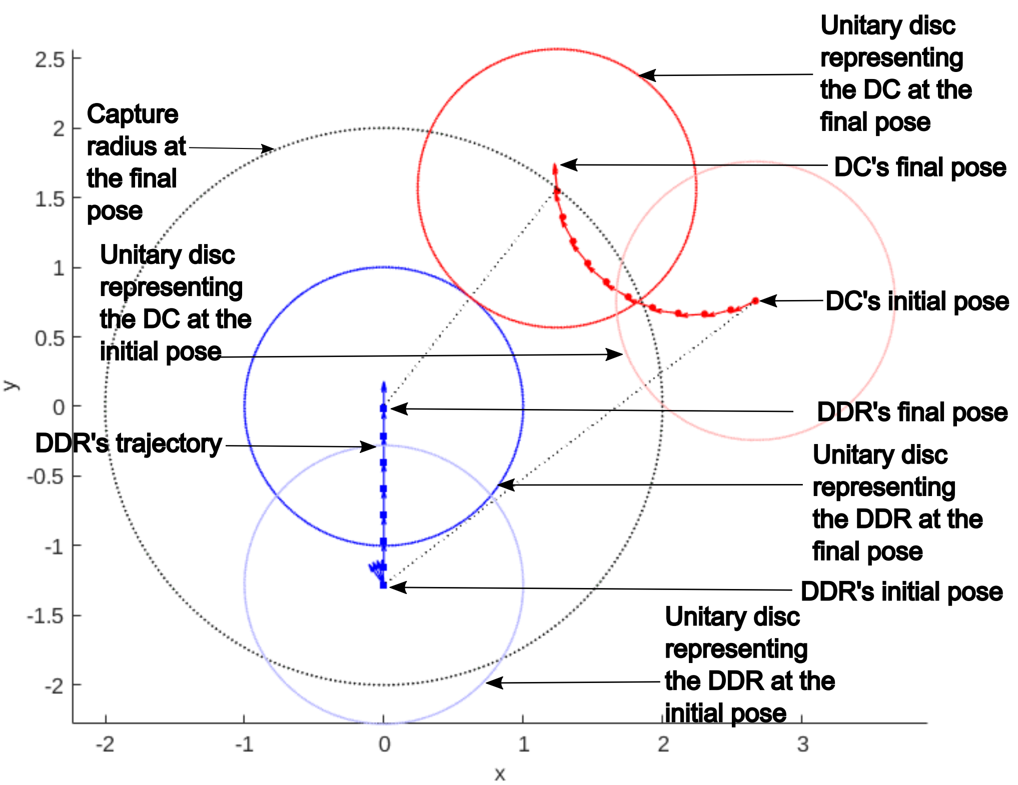
$$y = \left(x_s - \frac{r_e}{v^*}\cos\theta_s\right)\sin\left(\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right)$$
$$+ \left(y_s + \frac{r_e}{v^*}\sin\theta_s\right)\cos\left(\left(\frac{u_2^* - u_1^*}{2b}\right)(\tau - \tau_s)\right)$$



(a) The red curve shows the trajectory followed by the DC in the reduced space.



(b) Projection onto the x, y-plane of the trajectory (red curve) followed by the DC in the reduced space.



(c) Trajectories of the players in the realistic space. The blue curve indicates the trajectory followed by the DDR, and the red one corresponds to the trajectory followed by the DC. The arrows show the motion directions of the players.

FIGURE 5. **Results of the second simulation. The evader starts at the PDS in the reduced space.**

$$+\frac{r_e}{v^*}\sin\left(\left(\frac{u_2^*-u_1^*}{2b}\right)(\tau-\tau_s)-\frac{V^{\max}}{r_e}v^*(\tau-\tau_s)-\theta_s\right)$$

$$\theta=\theta_s+\frac{V^{\max}}{r_e}v^*(\tau-\tau_s)-\left(\frac{u_2^*-u_1^*}{2b}\right)(\tau-\tau_s) \tag{27}$$

The previous retro-time trajectories are valid until the system reaches configurations where $\theta = k\frac{\pi}{2}$ where $k \in \mathbf{Z}^+$ or the PDS dispersal surface.

VII. DECISION PROBLEM

Solving a pursuit-evasion game also involves finding the initial conditions that make capture possible for the DDR or escape for the DC. From [1], we have that the barrier separates the set of starting configurations into two sets. One with those that result in capture and another with those that result in escape. The approach we have used to compute the time-optimal motion strategies and their corresponding trajectories is also applied in constructing the barrier. The answer to the capture-escape question relies on whether or not the barrier divides the playing space into two parts.

To find the trajectories associated with the barrier, in this game, we need to perform the backward integration of the adjoint (17) and motion equations (24) considering as initial conditions the configurations belonging to the BUP, described by (10). Unfortunately, performing a detailed analysis of the barrier trajectories in the reduced space and validating if they define closed regions proved very difficult. We have to deal with discontinuities in the angles involved in the reduced model, and the solution entangles transcendental functions that do not allow us to compute the intersections between trajectories analytically; thus, we could not succeed in such a task.

However, we observed that capture is only possible for initial configurations near the UP in this game. It is not hard to realize that given that both players have the same speed, the DDR cannot reduce the distance to the DC once they have reached an alignment condition.

VIII. SIMULATIONS

This section presents two simulations of the players' motion strategies. In the first one, the evader starts at the EDS shown in Fig. 3b. In the second case, the evader begins at the PDS shown in Fig. 3d.

The parameters for the first simulation were $V^{\max} = 1\,m/s$, $l_c = 2\,m$, $b = 1\,m$, $r_e = 1\,m$, $\phi = 0.5240\,rad$, and $\psi = 1.2479\,rad$. The trajectory followed by the DC in the reduced space is shown in Fig. 4a. In Fig. 4b, the same trajectory is shown projected onto the x, y-plane. The DC is initially located at Region I and travels a trajectory departing from the EDS. The corresponding trajectories of the players in the realistic space are shown in Fig. 4c. In that figure, the DDR is represented like the blue circle, and the DC is shown like the red circle. We can observe that the DDR translates at maximum speed towards the DC (blue dashed line). The DC, on the contrary, tries to escape by leaving the front region of the DDR. Note that at the initial configuration, the DC (see Fig. (4b)) may have selected a control associated with the symmetric trajectory lying on the adjacent quadrant of the x, y-plane; however, both trajectories have the same outcome.

In the second simulation, the parameters were $V^{\max} = 1\,m/s$, $l_c = 2\,m$, $b = 1\,m$, $r_e = 1\,m$, $\phi = 0.6708\,rad$, and $\psi = 6.1832\,rad$. The trajectory followed by the DC in the reduced space is shown in Fig. 5a. Fig. 5b shows the trajectory's projection onto the x, y-plane. In this case, the DC is located at Region II initially, and it travels a trajectory starting at the PDS. After some time, it reaches Region I, and it follows the corresponding trajectory until the DC reaches the UP. The trajectories of the players in the realistic space are shown in Fig. 5c. In that figure, we can observe that the DDR initially rotates in place, trying to align its heading with the DC's position. After some time, the DDR translates at maximum speed towards the DC. The DC, on the contrary, tries to escape by leaving the front region of the DDR. Note that for this case, at the initial configuration, the DC needs to select the appropriate rotation direction to avoid benefitting the DDR. As in the first simulation, the DDR may have selected a control associated with the symmetric trajectory lying on the adjacent quadrant of the x, y-plane; however, both trajectories have the same outcome as long as the DC selects the correct rotation direction.

IX. CONCLUSION

In this work, we studied the problem of capturing a DC with a DDR in minimum time. We found the time-optimal motion strategies of the players to accomplish their tasks. We exhibited the existence of four singular surfaces in this game: two evader's dispersal surfaces (EDS), one pursuer's dispersal surface (PDS), and a transition surface (TS) where the DDR switches controls. We characterize the playing space into two regions where the players' optimal strategies are well-established. We also presented simulation examples of the players' trajectories at each region in the reduced space and the corresponding ones in the realistic space.

Unfortunately, performing a detailed analysis of the barrier trajectories (decision problem) in the reduced space and validating if they define closed regions proved very difficult. However, we observed that capture is only possible for initial configurations near the UP. One can quickly realize that since both players have the same speed, the DDR cannot reduce the distance to the DC once they have reached an alignment condition.

For future work, we are interested in constructing motion strategies for several DDRs that cooperate to capture one or several DCs. Another interesting problem is computing the time-optimal strategies when the players reverse roles, i.e., the DC plays as a pursuer, and the DDR plays an evader. From that analysis and the results of the current work, one can define the better roles for the players, i.e., if it is more convenient for each player to play as a pursuer or as an evader. An additional problem is considering the case when the players have visibility constraints, such as a bounded range or field of view. Finally, considering environments with obstacles is a very challenging and promising research avenue. In that case, one has to abandon the idea of obtaining an analytical solution characterizing the players' motion strategies in the playing space and employ heuristic methods that are sufficient to reach an approximation, like the ones in [28], [29].

REFERENCES

[1] R. Isaacs, *Differential Games*. New York, NY, USA: Wiley, 1965.
[2] A. Friedman, *Differential Games*. New York, NY, USA: Wiley, 1971.
[3] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory* (SIAM Series in Classics in Applied Mathematics), 2nd ed. Philadelphia, PA, USA: Academic, 1999.
[4] N. Karnad and V. Isler, "Lion and man game in the presence of a circular obstacle," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, St. Louis, MO, USA, Oct. 2009, pp. 5045–5050.
[5] R. Murrieta-Cid, U. Ruiz, J. L. Marroquin, J.-P. Laumond, and S. Hutchinson, "Tracking an omnidirectional evader with a differential drive robot," *Auto. Robots*, vol. 31, no. 4, pp. 345–366, Nov. 2011.

[6] W. Li, "A dynamics perspective of pursuit-evasion: Capturing and escaping when the pursuer runs faster than the agile evader," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 451–457, Jan. 2017.
[7] R. Zou and S. Bhattacharya, "On optimal pursuit trajectories for visibility-based target-tracking game," *IEEE Trans. Robot.*, vol. 35, no. 2, pp. 449–465, Apr. 2019.
[8] R. Yan, X. Duan, Z. Shi, Y. Zhong, and F. Bullo, "Matching-based capture strategies for 3D heterogeneous multiplayer reach-avoid differential games," *Automatica*, vol. 140, Jun. 2022, Art. no. 110207.
[9] J. P. Laumond, *Robot Motion Planning and Control*. Berlin, Germany: Springer, 1998.
[10] 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. 199–217, Mar. 2002.
[11] 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, May 1996.
[12] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The Mathematical Theory of Optimal Processes*. New York, NY, USA: Wiley, 1962.
[13] A. W. Merz, "The homicidal chauffeur—A differential game," Ph.D. dissertation, Dept. Aeronaut. Astronaut., Stanford Univ., Stanford, CA, USA, 1971.
[14] U. Ruiz, R. Murrieta-Cid, and J. L. Marroquin, "Time-optimal motion strategies for capturing an omnidirectional evader using a differential drive robot," *IEEE Trans. Robot.*, vol. 29, no. 5, pp. 1180–1196, Oct. 2013.
[15] U. Ruiz and R. Murrieta-Cid, "A differential pursuit/evasion game of capture between an omnidirectional agent and a differential drive robot, and their winning roles," *Int. J. Control*, vol. 89, no. 11, pp. 2169–2184, Feb. 2016.
[16] U. Ruiz, "A game of surveillance between an omnidirectional agent and a differential drive robot," *Int. J. Control*, vol. 95, no. 6, pp. 1694–1706, Jan. 2021.
[17] A. W. Merz, "The game of two identical cars," *J. Optim. Theory Appl.*, vol. 9, no. 5, pp. 324–342, May 1972.
[18] L. Bravo, U. Ruiz, and R. Murrieta-Cid, "A pursuit–evasion game between two identical differential drive robots," *J. Franklin Inst.*, vol. 357, no. 10, pp. 5773–5808, Jul. 2020.
[19] J. Lewin, *Differential Games: Theory and Methods for Solving Game Problems With Singular Surfaces*, London, U.K.: Springer, 2012.
[20] L. J. Guibas, J.-C. Latombe, S. M. LaValle, D. Lin, and R. Motwani, "A visibility-based pursuit-evasion problem," *Int. J. Comput. Geometry Appl.*, vol. 9, nos. 4–5, pp. 471–493, 1999.
[21] B. P. Gerkey, S. Thrun, and G. Gordon, "Visibility-based pursuit-evasion with limited field of view," *Int. J. Robot. Res.*, vol. 25, no. 4, pp. 299–315, Apr. 2006.
[22] B. Tovar and S. M. LaValle, "Visibility-based pursuit—Evasion with bounded speed," *Int. J. Robot. Res.*, vol. 27, nos. 11–12, pp. 1350–1360, Nov. 2008.
[23] I. Becerra, R. Murrieta-Cid, R. Monroy, S. Hutchinson, and J.-P. Laumond, "Maintaining strong mutual visibility of an evader moving over the reduced visibility graph," *Auto. Robots*, vol. 40, no. 2, pp. 395–423, Feb. 2016.
[24] 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, Mar. 2007.
[25] S. Bhattacharya and S. Hutchinson, "A cell decomposition approach to visibility-based pursuit evasion among obstacles," *Int. J. Robot. Res.*, vol. 30, no. 14, pp. 1709–1727, Sep. 2011.
[26] R. Bera, V. R. Makkapati, and M. Kothari, "A comprehensive differential game theoretic solution to a game of two cars," *J. Optim. Theory Appl.*, vol. 174, no. 3, pp. 818–836, Jul. 2017.
[27] A. Chaudhari and D. Chakraborty, "A time-optimal feedback control for a particular case of the game of two cars," *IEEE Trans. Autom. Control*, vol. 67, no. 4, pp. 1806–1821, Apr. 2022.
[28] E. Lozano, I. Becerra, U. Ruiz, L. Bravo, and R. Murrieta-Cid, "A visibility-based pursuit-evasion game between two nonholonomic robots in environments with obstacles," *Auton. Robot.*, vol. 46, no. 2, pp. 349–371, 2022.
[29] E. Lozano, U. Ruiz, I. Becerra, and R. Murrieta-Cid, "Surveillance and collision-free tracking of an aggressive evader with an actuated sensor pursuer," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6854–6861, Jul. 2022.

UBALDO RUIZ received the Ph.D. degree in computer science from the Centro de Investigación en Matemáticas (CIMAT), Guanajuato, Mexico, in 2013. From 2013 to 2014, he was a Postdoctoral Fellow with the Department of Computer Science, University of Minnesota. Since 2014, he has been a CONACYT Research Fellow working at the Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE), Baja California, Mexico. His research interests include robotics, differential games, optimal control, and motion planning.