
AUT Journal of Mathematics and Computing

AUT J. Math. Comput., 1(2) (2020) 153-163
DOI: 10.22060/ajmc.2019.16949.1025

# Heuristic artificial bee colony algorithm for solving the Homicidal Chauffeur differential game

Zahra Yazdaniyan[a], Mostafa Shamsi*[a], Maria do Rosário de Pinho [b], Zahra Foroozandeh [b]

[a]*Department of Applied Mathematics, Faculty of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran*
[b]*Department of Electrical and Computer Engineering, SYSTEC, Faculdade de Engenharia, Universidade do Porto, 4200-465, Porto, Portugal*

**ABSTRACT:** In this paper, we consider the Homicidal Chauffeur (HC) problem as an interesting and practical differential game. At first, we introduce a bilevel optimal control problem (BOCP) and prove that a saddle point solution for this game exists if and only if this BOCP has an optimal solution in which the optimal value of the objective function is equal to 1. Then, BOCP is discretized and converted to a nonlinear bilevel programming problem. Finally, an Artificial Bee Colony (ABC) algorithm is used for solving this problem, in which the lower-level problem will be considered as a constraint and solved by an NLP-solver. Finally, to demonstrate the effectiveness of the presented method, various cases of HC problem are solved and the simulation results are reported.

**Review History:**
Received:21 August 2019
Revised:12 October 2019
Accepted:15 October 2019
Available Online:01 September 2020

**Keywords:**
Differential game
Saddle point solution
Artificial bee colony
Bilevel optimal control

## 1. Introduction

The differential games are a type of dynamic games in which the game state is described by a system of differential equations [17, 2]. The Pursuit-Evasion (PE) Games are an important and interesting class of differential games, which model a conflict between two competitive players, namely *pursuer* and *evader*.

One of the most widely studied PE games is the Homicidal Chauffeur(HC) game [17]. The HC problem, as one of the first motivated problems in the field of dynamic games, has both historical and mathematical interests [22]. In the HC problem, a car (pursuer) and a pedestrian (evader) are located in an infinite planar parking. The car is faster with a finite minimum-turn radius and the pedestrian is slower but more agile. Capture occurs when the distance between the car and pedestrian becomes equal to a given *capture radius*. The car attempts to capture or run down the pedestrian in minimum time and the pedestrian wishes to maximize the capture time or avoid capturing.

There are also problems that have similar dynamic equations with HC problem, but in these problems, the objective of the players differ. Among them, surveillance-evasion problem [19], suicidal pedestrian and collision avoidance differential game [10, 11] are the most well known.

The difficulty of HC lies in the fact that solution of the problem is not unique and the solution depends very strongly upon the speeds of the players and maneuverability of the car [20].

The HC problem is introduced by Isaacs [17]. Since then, several numerical methods for solving this problem have been developed. For instance, we can refer to indirect methods [18], semi-direct methods [14, 23, 18], heuristic methods [15], HJB methods [12], and others [18, 21, 20].

*Corresponding author.
*E-mail addresses: z_yazdaniyan64@aut.ac.ir, m_shamsi@aut.ac.ir, mrpinho@fe.up.pt, zahra@fe.up.pt*

In the indirect methods, by using the necessary optimality conditions, the PE problem is reduced to a system of Differential Algebraic Equations (DAEs). Then, by eliminating the control functions, the obtained DAE is converted into a Two-Point Boundary Value Problem (TBVP), which can be solved by a BVP-solver. However, the numerical solution of TPBVPs arising from PE games is difficult due to at least three reasons: 1) the BVP-solvers need to an initial guess and generally the BVP-solvers are very sensitive to the initial guess when they are applied to the TPBVPs arising from PE games. Accordingly, providing a suitable initial guess is a major difficulty. 2) When the controls are bounded, as in HC problem, discontinuities appear in the resulting TPBVP. As a consequence, the discretization of the resulting TPBVP leads to a discontinuous or non-smooth system of algebraic equations, which causes the failure of solvers to find the correct solution. 3) There exist solutions for the mentioned BVP which are not the game solution.

It seems that the direct methods, which do not use the necessary conditions and are widely applied to optimal control problems, can be applied to PE problems, too. However, the direct discretization of PE games leads to a nontraditional optimization problem, which cannot be solved straightforwardly by the well-known optimization solvers. In addition, the accuracy is not satisfactory [9].

Due to the aforementioned difficulties occurred in the solution of PE problems by the indirect and direct methods, the researchers developed the so-called *semi-direct methods* [14, 15, 18]. The semi-direct methods are less sensitive to the initial guess in comparison with the indirect methods; However, in the practical problems, the sensitivity to initial guess is problematic such that the genetic algorithm is used to provide a suitable initial guess [15, 25]. It is worth noting that under some suitable conditions, the equivalence between the associated optimal control problem and original PE game is guaranteed [14, 26].

Due to technical challenges in the solution of PE games, practical methods often rely on heuristic methods. In these methods, a heuristic method (mostly Genetic Algorithm) is used for finding an initial guess. However, in most problems, the provided initial guesses are not a good approximation of the game solution. The main reason lies in the fact that these heuristics deal with the TPBVP, which arise from the indirect methods [14, 15, 16, 25, 24].

The goal of this paper is to provide a direct numerical method based on heuristic algorithms for solving HC problem, which leads to a solution that satisfies the necessary optimal condition but does not have the disadvantageous of the indirect methods.

In this regard, firstly, an equivalent BOCP to HC problem will be proposed. Then, utilizing the Euler method, this BOCP is discretized to a nonlinear bilevel optimization problem. In this bilevel problem, the lower-level problem is considered as a constraint and is solved by an NLP-solver. Moreover, the upper-level problem will be solved by heuristic *Artificial Bee Colony (ABC) Algorithm*.

The paper is organized as follows. Section 2 provides the mathematical model and necessary optimality condition of HC problem. Section 3 provides some necessary backgrounds for BOCPs. In Section 4, we present our new method for solving HC problem. The proposed method is applied to nine cases of HC in Section 5. Conclusions are then presented in Section 6.

## 2. Problem Statement

The mathematical model of the HC problem is as follows [17]

$$\text{HC Problem}: \begin{cases} \min\limits_{u_{\mathrm{p}}(t)} \max\limits_{u_{\mathrm{e}}(t)} \; J = t_{\mathrm{f}} & \text{(1a)} \\ \dot{\mathbf{x}}_{\mathrm{p}}(t) = \begin{bmatrix} \dot{x}_p(t) \\ \dot{y}_p(t) \\ \dot{\varphi}_{\mathrm{p}}(t) \end{bmatrix} = \begin{bmatrix} v_{\mathrm{p}} \cos(\varphi_{\mathrm{p}}(t)) \\ v_{\mathrm{p}} \sin(\varphi_{\mathrm{p}}(t)) \\ u_{\mathrm{p}}(t) \end{bmatrix}, \quad \mathbf{x}_{\mathrm{p}}(0) = \begin{bmatrix} x_{\mathrm{p}0} \\ y_{\mathrm{p}0} \\ \varphi_{\mathrm{p}0} \end{bmatrix} & \text{(1b)} \\ \dot{\mathbf{x}}_{\mathrm{e}}(t) = \begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_{\mathrm{e}}(t) \end{bmatrix} = \begin{bmatrix} v_{\mathrm{e}} \cos(u_{\mathrm{e}}(t)) \\ v_{\mathrm{e}} \sin(u_{\mathrm{e}}(t)) \end{bmatrix}, \quad \mathbf{x}_{\mathrm{e}}(0) = \begin{bmatrix} x_{\mathrm{e}0} \\ y_{\mathrm{e}0} \end{bmatrix} & \text{(1c)} \\ -1 \leq u_{\mathrm{p}}(t) \leq +1, & \text{(1d)} \\ (x_{\mathrm{p}}(t_{\mathrm{f}}) - x_{\mathrm{e}}(t_{\mathrm{f}}))^2 + (y_{\mathrm{p}}(t_{\mathrm{f}}) - y_{\mathrm{e}}(t_{\mathrm{f}}))^2 - d^2 = 0. & \text{(1e)} \end{cases}$$

In this problem, $v_{\mathrm{p}}$ and $v_{\mathrm{e}}$ are the speeds of car and pedestrian, respectively, and $v_{\mathrm{p}} > v_{\mathrm{e}}$. Note that, in problem (1), the final time $t_{\mathrm{f}}$ is the first time that capture occurs, in other words,

$$t_{\mathrm{f}} = \inf\{t \in \mathbb{R}^+ : (x_{\mathrm{p}}(t) - x_{\mathrm{e}}(t))^2 + (y_{\mathrm{p}}(t) - y_{\mathrm{e}}(t))^2 - d^2 = 0.\} \tag{2}$$

It is common for a terminal surface of a game to be divided into two regions: the *Usable Part* and the *Nonuseable Part*, which are separated by what is known in the literature as the *Boundary of the Usable Part*. The usable part is the subset of the terminal surface on which the pursuer can enforce capture, namely, penetration of the terminal surfaces [17]. The usable part of the terminal surface for the game at hand can be specified by

$$\frac{v_\mathrm{e}}{v_\mathrm{p}} < \cos(s), \quad 0 \leq s \leq \pi/2, \tag{3}$$

where $s = u_\mathrm{e}(t_\mathrm{f}) - \varphi_\mathrm{p}(t_\mathrm{f})$.

### 2.1. Necessary optimality conditions for saddle-point solution

To state the necessary conditions for saddle point solution of the problem (1), the following Hamiltonian function is considered

$$\mathcal{H}\left(u_\mathrm{p}, \mathbf{x}_\mathrm{p}, \boldsymbol{\lambda}_\mathrm{p}, u_\mathrm{e}, \mathbf{x}_\mathrm{e}, \boldsymbol{\lambda}_\mathrm{e}, t\right) := \boldsymbol{\lambda}_\mathrm{p}^T(t) \begin{bmatrix} \dot{x}_p(t) \\ \dot{y}_p(t) \\ \dot{\varphi}_p(t) \end{bmatrix} + \boldsymbol{\lambda}_\mathrm{e}^T(t) \begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_e(t) \end{bmatrix}, \tag{4}$$

where, $\boldsymbol{\lambda}_\mathrm{p}^T(t) \in \mathbb{R}^3$ and $\boldsymbol{\lambda}_\mathrm{e}^T(t) \in \mathbb{R}^2$ are the co-state variables conjugate to the state equations (1b) and (1c), respectively. Also, the function of terminal conditions is considered as follows,

$$\ell\left(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f}\right) := t_\mathrm{f} + \nu\left((x_\mathrm{e}(t_\mathrm{f}) - x_\mathrm{p}(t_\mathrm{f}))^2 + (y_\mathrm{e}(t_\mathrm{f}) - y_\mathrm{p}(t_\mathrm{f}))^2 - d^2\right), \tag{5}$$

where, $\boldsymbol{\nu} \in \mathbb{R}$ is the adjoint variable conjugate to the terminal constraint (1e).

According to [7], [2], if $\left(u_\mathrm{p}^*, u_\mathrm{e}^*, \mathbf{x}_\mathrm{p}^*, \mathbf{x}_\mathrm{e}^*, t_\mathrm{f}^*\right)$ be the solution of the HC problem (1), then there exist $\boldsymbol{\lambda}_\mathrm{p}^*(t) \in \mathbb{R}^3$ and $\boldsymbol{\lambda}_\mathrm{e}^*(t) \in \mathbb{R}^2$, such that the following co-state equations are satisfied

$$\dot{\boldsymbol{\lambda}}_\mathrm{p}^*(t) = \begin{bmatrix} \lambda_{x_\mathrm{p}}^*(t) \\ \lambda_{y_\mathrm{p}}^*(t) \\ \lambda_{\varphi_\mathrm{p}}^*(t) \end{bmatrix} = -\begin{bmatrix} \frac{\partial \mathcal{H}(\cdot)}{\partial x_\mathrm{p}(t)} \\ \frac{\partial \mathcal{H}(\cdot)}{\partial y_\mathrm{p}(t)} \\ \frac{\partial \mathcal{H}(\cdot)}{\partial \varphi_\mathrm{p}(t)} \end{bmatrix} = v_\mathrm{p}\begin{bmatrix} 0 \\ 0 \\ \lambda_{x_\mathrm{p}}^*(t)\sin\left(\varphi_\mathrm{p}^*(t)\right) - \lambda_{y_\mathrm{p}}^*(t)\cos\left(\varphi_\mathrm{p}^*(t)\right) \end{bmatrix}, \tag{6a}$$

$$\dot{\boldsymbol{\lambda}}_\mathrm{e}^*(t) = \begin{bmatrix} \lambda_{x_\mathrm{e}}^*(t) \\ \lambda_{y_\mathrm{e}}^*(t) \end{bmatrix} = -\begin{bmatrix} \frac{\partial \mathcal{H}(\cdot)}{\partial x_\mathrm{e}(t)} \\ \frac{\partial \mathcal{H}(\cdot)}{\partial y_\mathrm{e}(t)} \end{bmatrix} = \mathbf{0}, \tag{6b}$$

where $(\cdot) = (u_\mathrm{p}, \mathbf{x}_\mathrm{p}, \boldsymbol{\lambda}_\mathrm{p}, u_\mathrm{e}, \mathbf{x}_\mathrm{e}, \boldsymbol{\lambda}_\mathrm{e}, t)$.

Moreover, the co-states $\boldsymbol{\lambda}_\mathrm{p}^*(t)$ and $\boldsymbol{\lambda}_\mathrm{e}^*(t)$ satisfy the following terminal conditions

$$\boldsymbol{\lambda}_\mathrm{p}^*(t_\mathrm{f}^*) = \begin{bmatrix} \frac{\partial \ell(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f})}{\partial x_\mathrm{p}(t)} \\ \frac{\partial \ell(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f})}{\partial y_\mathrm{p}(t)} \\ \frac{\partial \ell(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f})}{\partial \varphi_\mathrm{p}(t)} \end{bmatrix} = -2\nu \begin{bmatrix} x_{ep}(t_\mathrm{f}) \\ y_{ep}(t_\mathrm{f}) \\ 0 \end{bmatrix}, \tag{6c}$$

$$\boldsymbol{\lambda}_\mathrm{e}^*(t_\mathrm{f}^*) = \begin{bmatrix} \frac{\partial \ell(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f})}{\partial x_\mathrm{e}(t)} \\ \frac{\partial \ell(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f})}{\partial y_\mathrm{e}(t)} \end{bmatrix} = 2\nu \begin{bmatrix} x_{ep}(t_\mathrm{f}) \\ y_{ep}(t_\mathrm{f}) \end{bmatrix}, \tag{6d}$$

such that

$$x_{ep}(t) := x_\mathrm{e}(t) - x_\mathrm{p}(t),$$
$$y_{ep}(t) := y_\mathrm{e}(t) - y_\mathrm{p}(t).$$

Transversality condition for this problem is

$$\mathcal{H}\left(u_\mathrm{p}, \mathbf{x}_\mathrm{p}, \boldsymbol{\lambda}_\mathrm{p}, u_\mathrm{e}, \mathbf{x}_\mathrm{e}, \boldsymbol{\lambda}_\mathrm{e}, t\right)|_{t=t_\mathrm{f}} = -\frac{\partial \ell}{\partial t_\mathrm{f}}\left(\mathbf{x}_\mathrm{p}(t_\mathrm{f}), \mathbf{x}_\mathrm{e}(t_\mathrm{f}), t_\mathrm{f}\right). \tag{6e}$$

By the equation (6e), we have

$$2\nu = \frac{-1}{v_\mathrm{p}\left[x_{ep}(t_\mathrm{f})\cos\left(\varphi_\mathrm{p}(t_\mathrm{f})\right) + y_{ep}(t_\mathrm{f})\sin\left(\varphi_\mathrm{p}(t_\mathrm{f})\right)\right] - v_\mathrm{e}\left[x_{ep}(t_\mathrm{f})\cos\left(u_e(t_\mathrm{f})\right) + y_{ep}(t_\mathrm{f})\sin\left(u_e(t_\mathrm{f})\right)\right]} \tag{6f}$$

Moreover, $(u_\mathrm{p}^*, u_\mathrm{e}^*)$ satisfies the following conditions

$$u_\mathrm{p}^*(t) = \underset{|u_\mathrm{p}| \leq 1}{\operatorname{argmin}} \mathcal{H} \Rightarrow u_p^*(t) = \begin{cases} 1 & \lambda_{\varphi_\mathrm{p}}^*(t) < 0 \\ \in [-1, 1] & \lambda_{\varphi_\mathrm{p}}^*(t) = 0 \\ -1 & \lambda_{\varphi_\mathrm{p}}^*(t) > 0 \end{cases} \tag{6g}$$

$$u_\mathrm{e}^*(t) = \underset{u_\mathrm{e}}{\operatorname{argmax}} \mathcal{H} \Rightarrow -\lambda_{x_\mathrm{e}}^*(t) v_\mathrm{e}\sin\left(u_e^*(t)\right) + \lambda_{y_\mathrm{e}}^*(t) v_\mathrm{e}\cos\left(u_e^*(t)\right) = 0. \tag{6h}$$

By equation (6b) and (6h), one may see easily that the control $u_\mathrm{e}^*(t)$ is constant in the whole time $[0, t_\mathrm{f}{}^*]$.

## 3. Background on Bilevel Programming Problems

In this section, we provide a brief introduction to bilevel programming.

**Definition 3.1.** *For an upper-level (leader) objective function $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and a lower-level (follower) objective function $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, a bilevel problem is expressed by*

$$\begin{aligned} &\min_{x\in X} \quad F(x,y) \\ &\qquad G_k(x,y) \le 0, \quad k = 1,\dots,K \\ &\qquad \min_{y\in Y} \quad f(x,y) \\ &\qquad\qquad g_j(x,y), \quad j = 1,\dots,J \le 0 \end{aligned} \tag{7}$$

*where $G_k : X \times Y \to \mathbb{R}, k = 1,\dots,K$ denote the upper-level constraints, and $g_j : X \times Y \to \mathbb{R}, j = 1,\dots,J$ represent the lower-level constraints, respectively. Equality constraints may also exist that have been avoided for brevity [3].*

The bilevel programming problem has a natural interpretation as a noncooperative game between two players. Player A (the "leader") chooses her decision x first, and afterward player B (the "follower") observes x and responds with a decision y. Both the objective function and the feasible region of the follower may depend on the leader's decision. Likewise, the leader has to satisfy a constraint that depends on the follower's decision.

**Definition 3.2.** *For a given (fixed) vector $\bar{x} \in X$, the lower-level feasible set is defined by*

$$\Omega(\bar{x}) = \{y \in Y : g(\bar{x}, y) \le 0\},$$

*while the lower-level reaction set is*

$$R(\bar{x}) = \{y \in Y : y \in \operatorname{argmin}(f(\bar{x}, \hat{y}), \hat{y} \in \Omega(\bar{x}))\}.$$

There are two modeling approaches to bilevel programming: optimistic and pessimistic. In the first one, it is assumed that, whenever the reaction set $R(x)$ is not a singleton, the leader is allowed to select the element in $\Omega(x)$ that suits him best, while in the second one, the cooperation of the leader and follower is not allowed [3].

BOCPs are a special class of optimization problems combining Bilevel Programming Programs and optimal control theory. A general formulation for BOCPs includes an optimal control problem in both levels, commonly linked with the dynamical system [3, 8, 13].

## 4. The Presented Method for solving HC

In this section, a new numerical method is presented to solve HC (1). In the presented method, at first the HC problem will be associated with an equivalent BOCP and then, by discretization of this BOCP, an ordinary bilevel problem will be obtained. Finally, this bilevel problem will be solved by a heuristic ABC algorithm.

### *4.1. Associating a BOCPs with the HC problem*

Let's consider the following BOCP

$$\max_{u_e, t_e \ge 0} \quad J_1 = \frac{1}{1 + (t_e - t_p)^2} \tag{8a}$$

$$\text{s.t.} \quad \dot{x}_e(t) = v_e \cos(u_e(t)), \quad x_e(0) = x_{e0}, \tag{8b}$$

$$\dot{y}_e(t) = v_e \sin(u_e(t)), \quad y_e(0) = y_{e0}, \tag{8c}$$

$$\sin(u_e(t)) = \frac{y_e(t_e) - y_p(t_p)}{d}, \tag{8d}$$

$$\cos(u_e(t)) = \frac{x_e(t_e) - x_p(t_p)}{d}, \tag{8e}$$

$$\min_{u_p \in [-1,1], t_p \ge 0} \quad J_2 = t_p \tag{8f}$$

$$\text{s.t.} \quad \dot{x}_p(t) = v_p \cos(\varphi_p(t)), \quad x_p(0) = x_{p0}, \tag{8g}$$

$$\dot{y}_p(t) = v_p \sin(\varphi_p(t)), \quad y_p(0) = y_{p0}, \tag{8h}$$

$$\dot{\varphi}_p(t) = u_p(t), \quad \varphi_p(0) = \varphi_{p0}, \tag{8i}$$

$$\big(x_p(t_p) - x_e(t_e)\big)^2 + \big(y_p(t_p) - y_e(t_e)\big)^2 - d^2 = 0, \tag{8j}$$

In BOCP (8), it is understood that, for a given $(u_{\mathrm{e}}(t), t_{\mathrm{e}})$, we first calculate $x_{\mathrm{e}}(t_{\mathrm{e}})$ and $y_{\mathrm{e}}(t_{\mathrm{e}})$ through (8b) and (8c) respectively. Then, we choose $(u_{\mathrm{p}}(t), t_{\mathrm{p}})$ to minimize $J_2$ subject to the lower-level problem (8f)-(8j). Now, if equations (8d) and (8e) hold, then $(u_{\mathrm{e}}(t), t_{\mathrm{e}})$ is feasible for the upper-level problem (8). We should maximize $J_1$ over all the feasible points $(u_{\mathrm{e}}(t), t_{\mathrm{e}})$. In problem (8), the pursuer is the lower-level decision-maker, which observes the strategy of the evader and then optimizes its strategy by solving the lower-level problem (8f)-(8j).

Let $u_{\mathrm{e}} = u_{\mathrm{e}}^*$ with final time $t_{\mathrm{e}} = t_{\mathrm{f}}^*$ be a solution of problem (8), where $u_{\mathrm{p}} = u_{\mathrm{p}}^*$ with final time $t_{\mathrm{p}} = t_{\mathrm{f}}^*$ is the optimal solution of the lower-level problem (8f)-(8j). In this case, we have $J_1 = 1$. For such case, we have the following theorem

**Theorem 4.1.** *HC problem* (1)*, has an optimal solution, iff the BOCP* (8) *has an optimal solution, and the optimal value of the upper-level objective function is 1.*

**Proof.** Suppose that $(u_{\mathrm{e}}(t), t_{\mathrm{e}})$ is the optimal solution of (8), and the value of upper-level objective function equals 1. We will show that this solution satisfies in (6). Note that, in this case, we have $t_{\mathrm{e}} = t_{\mathrm{p}}$. This common value for the final time will be shown by $t_{\mathrm{f}}$.

For this solution, $x_{\mathrm{e}}(t_f)$ and $y_{\mathrm{e}}(t_f)$ are obtained by equation (8b) and (8c). Then these two variables are given to the lower-level problem (8f)-(8j) as known parameters. The optimal solution of the lower-level problem is *the best response of pursuer to evader.*

Let $u_{\mathrm{p}}(t)$ and $t_{\mathrm{f}}$ be the optimal solution of the lower-level problem, according to Bryson and Ho [7], there exist $\gamma_{x_{\mathrm{p}}}(t)$, $\gamma_{y_{\mathrm{p}}}(t)$ and $\gamma_{\varphi_{\mathrm{p}}}(t)$ such that the following first order optimality conditions are satisfied

$$\text{Costate equation:} \begin{cases} \dot{\gamma}_{x_{\mathrm{p}}}(t) = -\frac{\partial \mathcal{H}_{\mathrm{p}}}{\partial x_{\mathrm{p}}} = 0, \\ \dot{\gamma}_{y_{\mathrm{p}}}(t) = -\frac{\partial \mathcal{H}_{\mathrm{p}}}{\partial y_{\mathrm{p}}} = 0, \\ \dot{\gamma}_{\varphi_{\mathrm{p}}}(t) = -\frac{\partial \mathcal{H}_{\mathrm{p}}}{\partial \varphi_{\mathrm{p}}} = -\gamma_{x_{\mathrm{p}}}(t) v_{\mathrm{p}} \sin(\varphi_{\mathrm{p}}(t)) + \gamma_{y_{\mathrm{p}}}(t) v_{\mathrm{p}} \cos(\varphi_{\mathrm{p}}(t)), \end{cases} \tag{9a}$$

$$\text{Control equation: } u_p(t) = \begin{cases} 1 & \text{if} \quad \gamma_{\varphi_{\mathrm{p}}}(t) < 0 \\ 0 & \text{if} \quad \gamma_{\varphi_{\mathrm{p}}}(t) = 0 \\ -1 & \text{if} \quad \gamma_{\varphi_{\mathrm{p}}}(t) > 0 \end{cases} \tag{9b}$$

$$\text{Terminal condition:} \begin{cases} \gamma_{x_{\mathrm{p}}}(t_{\mathrm{f}}) = \frac{\partial \ell_{\mathrm{p}}}{\partial x_{\mathrm{p}}} = -2\alpha x_{ep}(t_{\mathrm{f}}) \\ \gamma_{y_{\mathrm{p}}}(t_{\mathrm{f}}) = \frac{\partial \ell_{\mathrm{p}}}{\partial y_{\mathrm{p}}} = -2\alpha y_{ep}(t_{\mathrm{f}}) \\ \gamma_{\varphi_{\mathrm{p}}}(t_{\mathrm{f}}) = \frac{\partial \ell_{\mathrm{p}}}{\partial \varphi_{\mathrm{p}}} = 0 \end{cases} \tag{9c}$$

$$\text{Transversality condition: } \mathcal{H}_{\mathrm{p}}\left(u_{\mathrm{p}}, x_{\mathrm{p}}, y_{\mathrm{p}}, \gamma_{x_{\mathrm{p}}}, \gamma_{y_{\mathrm{p}}}, \gamma_{\varphi_{\mathrm{p}}}, t\right)|_{t=t_{\mathrm{f}}} = -\frac{\partial \ell_{\mathrm{p}}}{\partial t_{\mathrm{p}}} \tag{9d}$$

where

$$\mathcal{H}_{\mathrm{p}}\left(u_{\mathrm{p}}, x_{\mathrm{p}}, y_{\mathrm{p}}, \gamma_{x_{\mathrm{p}}}, \gamma_{y_{\mathrm{p}}}, \gamma_{\varphi_{\mathrm{p}}}, t\right) := \gamma_{x_{\mathrm{p}}}(t) v_{\mathrm{p}} \cos(\varphi_{\mathrm{p}}(t)) + \gamma_{y_{\mathrm{p}}}(t) v_{\mathrm{p}} \sin(\varphi_{\mathrm{p}}(t))) + \gamma_{\varphi_{\mathrm{p}}}(t) u_p(t),$$

and

$$\ell_{\mathrm{p}}\left(x_{\mathrm{p}}(t_{\mathrm{p}}), y_{\mathrm{p}}(t_{\mathrm{p}}), t_{\mathrm{p}}; x_{\mathrm{e}}(t_{\mathrm{e}}), y_{\mathrm{e}}(t_{\mathrm{e}})\right) := t_{\mathrm{p}} + \alpha\left[\left(x_{\mathrm{p}}(t_{\mathrm{p}}) - x_{\mathrm{e}}(t_{\mathrm{e}})\right)^2 + \left(y_{\mathrm{p}}(t_{\mathrm{p}}) - y_{\mathrm{e}}(t_{\mathrm{e}})\right)^2 - d^2\right].$$

are respectively the Hamiltonian function and the function of terminal condition for the lower-level problem (8f)-(8j).

Define

$$\dot{\gamma}_{x_{\mathrm{e}}}(t) = 0, \qquad \gamma_{x_{\mathrm{e}}}(t_{\mathrm{f}}) = 2\alpha x_{ep}(t_{\mathrm{f}}), \tag{9e}$$

$$\dot{\gamma}_{y_{\mathrm{e}}}(t) = 0, \qquad \gamma_5(t_f) = 2\alpha y_{ep}(t_{\mathrm{f}}). \tag{9f}$$

Now using equation (8d) and (8e), we have

$$\gamma_{y_{\mathrm{e}}}(t) \cos(u_e(t)) - \gamma_{x_{\mathrm{e}}}(t) \sin(u_e(t)) = 0. \tag{9g}$$

Let

$$2\acute{\nu} = \frac{1}{v_{\mathrm{p}}(x_{ep}(t_{\mathrm{f}}) \cos\varphi_{\mathrm{p}}(t_{\mathrm{f}}) + y_{ep}(t_{\mathrm{f}}) \sin\varphi_{\mathrm{p}}(t_{\mathrm{f}})) - v_{\mathrm{e}}(x_{ep}(t_{\mathrm{f}}) \cos(u_e(t_{\mathrm{f}})) + y_{ep}(t_{\mathrm{f}}) \sin(u_e(t_{\mathrm{f}})))}. \tag{10}$$

By (3), we have

$$\frac{\alpha}{\acute{\nu}} = 1 - \frac{v_{\mathrm{e}}}{v_{\mathrm{p}}} \frac{d}{d\cos(\varphi_{\mathrm{p}}(t_{\mathrm{f}}) - u_e(t_{\mathrm{f}}))} > 0. \tag{11}$$

Multiply variables $\gamma_{x_{\mathrm{p}}}, \gamma_{y_{\mathrm{p}}}, \gamma_{\varphi_{\mathrm{p}}}, \gamma_{x_{\mathrm{e}}}$ and $\gamma_{y_{\mathrm{e}}}$ by $\frac{\acute{\nu}}{\alpha}$, equation (9) will be converted to necessary optimality condition (6).

Conversely, suppose that $(u_p(t), u_e(t), t_f)$ is the optimal solution of game, i.e., satisfies necessary optimality conditions (6). We will show that this is a feasible solution for bilevel (8) and clearly for this solution, the value of upper-level objective function equals 1. It is obvious that equation (8b), (8c) and (8g)-(8j) hold. By equation (6b), (6d) and (6h) we have,

$$\tan u_{\mathrm{e}}(t) = \frac{\lambda_{y_{\mathrm{e}}}(t)}{\lambda_{x_{\mathrm{e}}}(t)} = \frac{y_{ep}(t_{\mathrm{f}})}{x_{ep}(t_{\mathrm{f}})}, \tag{12}$$

and then,

$$\sin(u_{\mathrm{e}}(t)) = \frac{y_{ep}(t_{\mathrm{f}})}{d}, \tag{13a}$$

$$\cos(u_{\mathrm{e}}(t)) = \frac{x_{ep}(t_{\mathrm{f}})}{d}. \tag{13b}$$

So equation (8d) and (8e) hold. It is concluded from equation (12) that, pursuer's final position lies on the evader's path. In other words, evader's optimal path overlaps with the vector $(x_{ep}(t_{\mathrm{f}}), y_{ep}(t_{\mathrm{f}}))$. It is because that evader is maximizer and pursuer is minimizer. In addition, by the definition of saddle point solution in differential games [6], this solution is the optimal solution for the lower-level problem (8f)-(8j).

□

**Remark 1.** *In the lack of equations* (8d)-(8e)*, the bilevel* (8) *convert to an optimistic model, which does not correspond to the HC problem.*

**Remark 2.** *Note that there exist* $(u_{\mathrm{e}}(t), u_{\mathrm{p}}(t), t_f)$ *that satisfies the necessary optimality condition* (6)*, but equation* (2) *does not hold for them and hence these kinds of triples are not an optimal solution for HC problem* (1)*. The bilevel problem* (8)*, discard these solutions because they are not satisfied the equations* (8d) *and* (8e)*.*

### *4.2. Discretization of the obtained BOCP*

In this paper, to discretize the BOCP (8), we use the direct Euler method [5, 4], which is known as a simple and robust method for solving optimal control problems. We should mention that any other method can be used for discretizing BOCP (8).

Let $n$ be a positive integer number and $h_{\mathrm{s}} := t_{\mathrm{s}}/n,\ \mathrm{s} = \mathrm{p}, \mathrm{e}$. Define $\tau_{\mathrm{s},i} := ih_{\mathrm{s}},\ i = 0, \dots, n,\ \mathrm{s} = \mathrm{p}, \mathrm{e}$ and for the sake of simplicity in notations, set

$$a_{\mathrm{s},i} := x_{\mathrm{s}}(\tau_{\mathrm{s},i}), \quad b_{\mathrm{s},i} := y_{\mathrm{s}}(\tau_{\mathrm{s},i}), \quad c_i := \varphi_{\mathrm{p}}(\tau_{\mathrm{s},i}), \quad d_{\mathrm{s},i} := u_{\mathrm{s}}(\tau_{\mathrm{s},i}), \qquad i = 0, \dots, n, \quad \mathrm{s} = \mathrm{p}, \mathrm{e}.$$

In the direct Euler method, the differential equations are approximated by the Euler method with step length $h$. In summary, by applying the direct Euler method, the BOCP (8) is transcribed to the following bilevel programming problem:

$$\max_{d_{\mathrm{e},i},\ t_{\mathrm{e}} \geq 0} J_1 = \frac{1}{1 + (t_{\mathrm{e}} - t_{\mathrm{p}})^2} \tag{14a}$$

$$\text{s.t.} \quad a_{\mathrm{e},i+1} = a_{\mathrm{e},i} + \frac{t_{\mathrm{e}}}{n} v_{\mathrm{e}} \cos(d_{\mathrm{e},i}), \ i = 0, \dots, n-1, \quad a_{\mathrm{e},0} = x_{\mathrm{e}0}, \tag{14b}$$

$$b_{\mathrm{e},i+1} = b_{\mathrm{e},i} + \frac{t_{\mathrm{e}}}{n} v_{\mathrm{e}} \sin(d_{\mathrm{e},i}), \ i = 0, \dots, n-1, \quad b_{\mathrm{e},0} = y_{\mathrm{e}0}, \tag{14c}$$

$$\sin(d_{\mathrm{e},n}) = \frac{b_{\mathrm{e},n} - b_{\mathrm{p},n}}{d}, \tag{14d}$$

$$\cos(d_{\mathrm{e},n}) = \frac{a_{\mathrm{e},n} - a_{\mathrm{p},n}}{d}, \tag{14e}$$

$$\min_{|d_{\mathrm{p},i}| \leq 1, i=0,\dots,n,\ t_{\mathrm{p}} \geq 0} J_2 = t_{\mathrm{p}} \tag{14f}$$

$$\text{s.t.} \quad a_{\mathrm{p},i+1} = a_{\mathrm{p},i} + \frac{t_{\mathrm{p}}}{n} v_{\mathrm{p}} \cos(c_i), \ i = 0, \dots, n-1, \ a_{\mathrm{p},0} = x_{\mathrm{p}0}, \tag{14g}$$

$$b_{\mathrm{p},i+1} = b_{\mathrm{p},i} + \frac{t_{\mathrm{p}}}{n} v_{\mathrm{p}} \sin(c_i), \ i = 0, \dots, n-1, \ b_{\mathrm{p},0} = y_{\mathrm{p}0}, \tag{14h}$$

$$c_{i+1} = c_i + \frac{t_{\mathrm{p}}}{n} d_{\mathrm{p},i}, \qquad c_0 = \varphi_{\mathrm{p}0}, \tag{14i}$$

$$\left(a_{\mathrm{p},n} - a_{\mathrm{e},n}\right)^2 + \left(b_{\mathrm{p},n} - b_{\mathrm{e},n}\right)^2 - d^2 = 0. \tag{14j}$$

We note that in the above BOCP, $t_{\mathrm{s}},\ a_{\mathrm{s},i},\ b_{\mathrm{s},i},\ c_i,\ d_{\mathrm{s},i},\ \ \mathrm{s} = \mathrm{p}, \mathrm{e},\ i = 0, \dots, n$ are decision variables.

### 4.3. ABC Approach

There are various solution approaches for bilevel programming problems [8]. These methods need some requirements such as regularity and convexity to guarantee the convergence. In addition, all of these methods require an initial guess for solving these problems and may be really sensitive to it. Also, the obtained solution is not always the global one. A good solution method for solving optimization problems are meta heuristics, which are problem independent and need no initial guess.

In this paper, an artificial bee colony algorithm is used for solving the upper-level problem, while the lower-level problem is solved by an NLP-solver. Now, in what follows, the details of ABC algorithm will be described.

Here, each solution is represented by a pair of $X_i = (u_\mathrm{e}(t), t_\mathrm{e})$. In the ABC algorithm, the swarm is divided into two equal groups: a group of employed bees and a group of onlooker bees [1].

The initial population consists of SN solutions (food sources), as follows:

$$X_{i,d} = lb_d + \psi(ub_d - lb_d), \qquad \forall i \in \{1, \dots, SN\}, \quad \forall d \in \{1, 2\}, \tag{15}$$

where $\psi$ is a random number within $[0, 1]$ and $lb_d$, $ub_d$ are the lower and upper bounds for the $d$th element of $X_i$, respectively. For all of the solution, $u_\mathrm{e}(t)$ is the angle of evader's motion and changes in $[0, 2\pi)$ and $t_\mathrm{e}$ related to evader's motion time and it is greater than zero. A more precise lower bound for this variable can be founded by solving the minimum time optimal control problem with dynamics in HC problem (1). This problem ignores the role of evader and lets the pursuer to capture him as soon as possible. In this paper, $5 \times lb_2$ is considered as the upper bound for this variable.

First, for each solution $X = (u_\mathrm{e}(t), t_\mathrm{e})$, $x_\mathrm{e}(t_\mathrm{e})$ and $y_\mathrm{e}(t_\mathrm{e})$ will be calculated through equation (8b) and (8c). Next, the best response of pursuer to the final position of evader by solving the lower-level problem (8f)-(8j) will be determined. Suppose that $(u_\mathrm{p}(t), t_\mathrm{p})$ is the optimal solution of this problem. Then, by pursuer's dynamic (8g) and (8h), we can calculate pursuer's final position $x_\mathrm{p}(t_\mathrm{p})$ and $y_\mathrm{p}(t_\mathrm{p})$. By theorem 4.1, the *fitness* function can be defined as,

$$\mathrm{fitness}\,(X) = \frac{1}{1 + |t_\mathrm{e} - t_\mathrm{p}| + |\sin(u_\mathrm{e}(t_\mathrm{e})) - \frac{y_\mathrm{e}(t_\mathrm{e}) - y_\mathrm{p}(t_\mathrm{p})}{d}| + |\cos(u_\mathrm{e}(t_\mathrm{e})) - \frac{x_\mathrm{e}(t_\mathrm{e}) - x_\mathrm{p}(t_\mathrm{p})}{d}|}. \tag{16}$$

In the employed bees' phase, the neighborhood of each solution will be exploited, that is for each solution $X_i$, a new solution $Y_i$ will be generated as follows:

$$Y_{i,d} = X_{i,d} + \rho_{i,d}(X_{i,d} - X_{j,d}), \tag{17}$$

where $X_j$ is a randomly selected candidate solution $(i \neq j)$, $d$ is a random dimension index selected from the set $\{1, 2\}$, and $\rho_{i,d}$ is a random number within $[-1, 1]$. Once the new candidate solution $Y_i$ is generated, a greedy selection is used, and the better solution between $X_i$ and $Y_i$ will be considered as the $i$th solution.

After completing the employed phase, the probability value for each solution $X_i$, will be calculated as follows:

$$P_i = \frac{\mathrm{fitness}\,(X_i)}{\sum_{i=1}^{SN} \mathrm{fitness}\,(X_i)}. \tag{18}$$

By equation (18), for a better solution, the higher probability will be obtained.

In the onlooker bees' phase, an onlooker selects a solution with probability $P_i$. In other words, this probabilistic selection is a roulette wheel mechanism. The neighborhood of this solution will be exploited again as in the employed bees' phase.

If a solution cannot be improved over a predefined number, called *limit of cycles*, then it will be abandoned. Assume that the abandoned solution is $X_i$, then the scout bee explores a new one to be replaced with $X_i$ as in equation (15).

This procedure is repeated until at least one of the following stopping criteria met:

1. The number of cycles is less than a user-defined parameter `maxCycle`.
2. By equation (16), it is clear that the best amount of fitness is 1. So, an acceptable tolerance $\epsilon$ for the fitness can be considered as a stopping criterion.

## 5. Numerical Results

To illustrate and assess the presented method, we have implemented it on a 3.5 GHz PC using Matlab. Moreover, the Interior-Point Optimization Solver IPOPT [27] is used for solving the NLP (14f)-(14j).

Table 1: Nine cases (versions) of HC problem. The values of parameters, together with the exact final time and control functions.

| Case No. | Parameters of the HC problem | | | | | | Exact Solution | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\varphi_{\mathrm{p}0}$ | $x_{\mathrm{e}0}$ | $y_{\mathrm{e}0}$ | $v_{\mathrm{p}}$ | $v_{\mathrm{e}}$ | $d$ | Game time $t_{\mathrm{f}}^{*}$ | $u_{\mathrm{p}}(t)$ | $u_{\mathrm{e}}(t)\vert_{t\in[0,t_{\mathrm{f}}^{*}]}$ |
| 1 | 0.0 | 4.0 | 4.0 | 3 | 1.0 | 1.0 | 2.50392193161987 | $+\chi_{[0\,,\,1.05980557949785]}$ | 1.05980557949785 |
| 2 | 0.0 | -7.0 | 0.0 | 3 | 1.0 | 1.0 | 8.92706433923994 | $+\chi_{[0\,,\,3.95137622615996]}$ | -2.33180908101963 |
| 3 | 0.0 | 2.5 | 7.5 | 3 | 1.0 | 1.0 | 4.12039939580511 | $+\chi_{[0\,,\,1.68583288631328]}$ | 1.69076396264651 |
| 4 | 0.0 | -1.0 | 3.0 | 3 | 1.0 | 1.0 | 9.74025320137715 | $-\chi_{[0\,,\,5.06300126020837]}$ | 1.22018404697122 |
| 5 | $\pi/2$ | 3.0 | 0.0 | 1 | 0.1 | 0.8 | 3.36271767773564 | $-\chi_{[0\,,\,2.09439510239319]}$ | -0.52359877559830 |
| 6 | $\pi/2$ | 2.0 | 0.0 | 1 | 0.1 | 0.8 | 2.59420467933156 | $-1$ | -1.29710233966578 |
| 7 | $\pi/2$ | 1.0 | 0.0 | 1 | 0.1 | 0.8 | 5.07653935214838 | $2\chi_{[0\,,\,0.409889237583752]}-1$ | 2.78895316068613 |
| 8 | $\pi/2$ | 1.0 | -3.0 | 1 | 0.1 | 0.8 | 6.12206298643345 | $-\chi_{[0\,,\,3.48142956304391]}$ | -1.91063323624902 |
| 9 | $\pi/2$ | 0.1 | -1.0 | 1 | 0.1 | 0.8 | 5.34709886709410 | $-\chi_{[0\,,\,4.71238898038469]}$ | 3.14159265358979 |

Without loss of generality, the initial position of the car is $(0, 0)$ and the initial position of the pedestrian is $(x_{\mathrm{e}0}, y_{\mathrm{e}0})$. In Table 1, nine cases (versions) of HC problem are considered. These cases are treated in [14, 23, 18] and are different in the car and pedestrian's speed, initial position of pedestrian and capture radius. The parameters of the HC problem and the exact control functions and final time are also given in Table 1. In this table, $\chi$ is the characteristic function.

The parameters of the ABC algorithm are reported in Table 2. For each case, the method is executed ten times

Table 2: ABC parameters for all cases of the HC problem.

| Case No. | SN | limit | $\epsilon$ | `maxCycle` |
|---|---|---|---|---|
| I | 5 | 20 | 1e-5 | 500 |
| II | 10 | 40 | 0.05 | 1000 |
| III | 5 | 20 | 1e-5 | 500 |
| IV | 10 | 40 | 0.05 | 1000 |
| V | 5 | 20 | 1e-5 | 500 |
| VI | 5 | 20 | 1e-5 | 500 |
| VII | 10 | 40 | 1e-2 | 1000 |
| VIII | 10 | 40 | 0.05 | 1000 |
| IX | 5 | 20 | 0.05 | 500 |

with $n = 100$. The maximum, minimum and average of the obtained best fitnesses, CPU time and number of cycles among these ten times, are reported in Table 3. As we see in Tables 2 and 3, for case II and IV the `maxCycle` parameter set to 1000 and this stopping criterion is met for all ten executions, However, for remained cases, the second stopping criterion is met. The best and worst executions for each case are also shown in Figure 1.

To assess the accuracy of the presented method, the following quantities are defined

$$E_{t_{\mathrm{e}}} := |t_f^* - t_{\mathrm{e}}|,\ E_{u_{\mathrm{e}}} := \max_{i=1,\ldots,n} |u_{\mathrm{e}}^*(\tau_{\mathrm{e},i}) - u_{\mathrm{e}}(\tau_{\mathrm{e},i})|\,. \tag{19}$$

The above quantities are reported in Table 4. We see that, by the presented method, satisfactory accuracy within the heuristic method is obtained.

Table 3: Summaries of the result of ABC algorithm on the HC problem

| Case No. | Fitness | | | CPU Time(s) | | | Cycle | | |
|---|---|---|---|---|---|---|---|---|---|
| | max | avg | min | max | avg | min | max | avg | min |
| I | 0.999995 | 0.999992 | 0.999990 | 3070 | 1086 | 376 | 146 | 76 | 46 |
| II | 0.928090 | 0.8690927 | 0.782780 | 112119 | 90803 | 77952 | 1000 | 1000 | 1000 |
| III | 0.999997 | 0.999993 | 0.999990 | 2571 | 958 | 367 | 151 | 77 | 44 |
| IV | 0.817387 | 0.7770183 | 0.742328 | 79414 | 60910 | 48199 | 1000 | 1000 | 1000 |
| V | 0.999999 | 0.999995 | 0.999991 | 2501 | 922 | 257 | 157 | 85 | 51 |
| VI | 0.999999 | 0.9999953 | 0.999991 | 3065 | 1934 | 1008 | 230 | 118 | 67 |
| VII | 0.994790 | 0.990830 | 0.984307 | 40974 | 28382 | 369 | 1000 | 704 | 16 |
| VIII | 0.987737 | 0.967191 | 0.952411 | 344 | 196 | 89 | 16 | 11 | 6 |
| IX | 0.985811 | 0.967772 | 0.955665 | 3471 | 1578 | 154 | 488 | 187 | 17 |

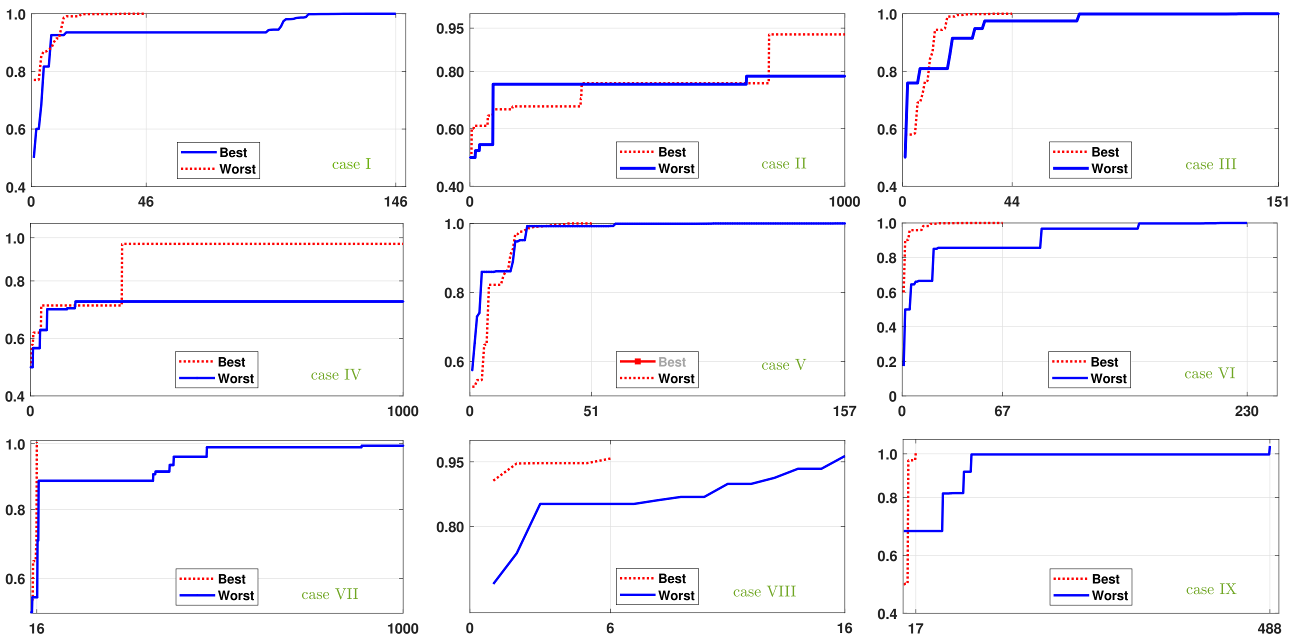


Figure 1: The worst and best execution of ABC algorithm for all cases.

Table 4: Error of the obtained final time and control of the evader.

| Case | $E_{u_e}$ | $E_{t_e}$ |
|---|---|---|
| I | 9.56e-06 | 6.62e-05 |
| II | 8.8e-03 | 1.09e-01 |
| III | 4.9e-03 | 2.12e-04 |
| IV | 9.60e-02 | 4.2e-02 |
| V | 1.77e-06 | 9.03e-05 |
| VI | 2.02e-04 | 1.00e-06 |
| VII | 1.20e-03 | 3.2e-03 |
| VIII | 4.8e-03 | 8.4e-03 |
| IX | 2.00e-03 | 1.43e-02 |

For depicting Remark 2, we can mention case V. The exact solution for this case is shown in figure 2(right). In this case, the triple $(u_{\mathrm{p}}(t), u_{\mathrm{e}}(t), t_f)$, that is shown in the left figure, satisfies the necessary optimal condition, but this triple is not even feasible for the bilevel (8).

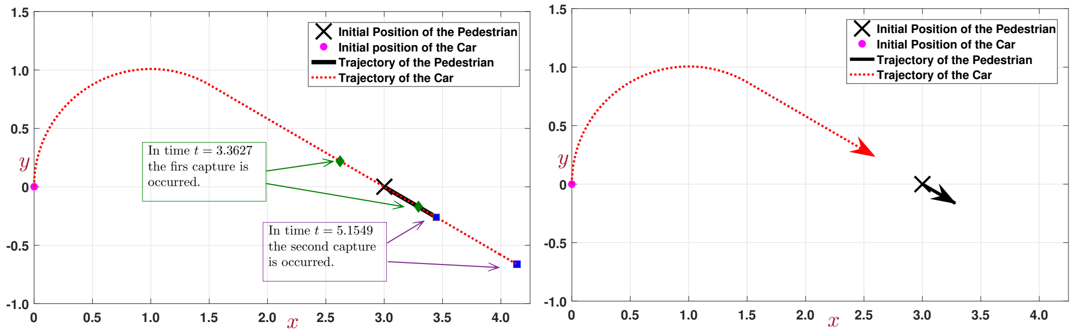


Figure 2: (The HC problem in the case V.) Left: the triple $(u_p, u_e, t_f)$ which is not the game solution. Right: the game solution.

## 6. conclusion

In this paper, the HC problem is reformulated as a BOCP, then by utilizing Euler discretization, this problem is converted to a parametric bilevel problem. For solving this bilevel problem, the ABC based method is used, while the lower-level problem is considered as a constraint and is solved by an NLP solver for each member of population. Nine cases of this game are examined and good solutions for all cases are found. By this paper, we showed that the heuristic ABC algorithm can solve the HC problem directly and without relying on the necessary optimality conditions. We believe that the presented method can be extended to solve the differential game problems with sophisticated dynamics.

## References

[1] S. Aslan, H. Badem, D. Karaboga, Improved quick artificial bee colony (iqABC) algorithm for global optimization, Soft Computing, 2019.

[2] T. Başar, G. Olsder, Dynamic Noncooperative Game Theory, 2nd Edition, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1998.

[3] J. F. Bard, Practical bilevel optimization: algorithms and applications, volume 30. Springer Science & Business Media, 2013.

[4] J. Betts, Practical Methods for Optimal Control and Estimation Using Nonlinear Programming, Society for Industrial and Applied Mathematics, second edition, 2010.

[5] J. T. Betts, Survey of numerical methods for trajectory optimization, Journal of Guidance, Control, and Dynamics, 21(2) (1998) 193-207.

[6] A. E. Bryson, Applied optimal control: optimization, estimation and control, Routledge, 2018.

[7] A. E. Bryson, Y-C. Ho, Applied optimal control: optimization, estimation and control, Hemisphere, 1975.

[8] B. Colson, P. Marcotte, G. Savard, An overview of bilevel optimization, Annals of Operations Research, 153(1) (2007) 235-256.

[9] H. Ehtamo, T. Raivio, On applied nonlinear and bilevel programming or pursuit-evasion games, Journal of Optimization Theory and Applications, 108(1) (2001) 65-96.

[10] I. Exarchos, P. Tsiotras, M. Pachter, On the suicidal pedestrian differential game, *Dynamic Games and Applications*, 5(3) (2015) 297-317.

[11] I. Exarchos, P. Tsiotras, M. Pachter, UAV collision avoidance based on the solution of the suicidal pedestrian differential game, 2016.

[12] M. Falcone, Numerical methods for differential games based on partial differential equations, International Game Theory Review, 8(2) (2006) 231-272.

[13] L. Guo, J. J. Ye, Necessary optimality conditions for optimal control problems with equilibrium constraints, SIAM Journal on Control and Optimization, 54(5) (2016) 2710-2733.

[14] K. Horie, Collocation with nonlinear programming for two-sided flight path optimization, PhD thesis, University of Illinois at Urbana-Champaign, 2002.

[15] K. Horie, B. A. Conway, Genetic algorithm preprocessing for numerical solution of differential games problems, *Journal of Guidance, Control, and Dynamics*, 27(6) (2004) 1075-1078.

[16] K. Horie, B. A. Conway, Optimal fighter pursuit-evasion maneuvers found via two-sided optimization, Journal of Guidance, Control, and Dynamics, 29(1) (2006)105-112.

[17] R. Isaacs, Differential games. A mathematical theory with applications to warfare and pursuit, control and optimization, John Wiley & Sons, Inc., New York-London-Sydney, 1965.

[18] P. A. Johnson, Numerical Solution Methods for Differential Game Problems, PhD thesis, Massachusetts Institute of Technology, 2009.

[19] J. Lewin, J. V. Breakwell, The surveillance-evasion game of degree, Journal of Optimization Theory and Applications, 16(3-4) (1975) 339-353.

[20] V. S. Patsko, V. L. Turova, Families of semipermeable curves in differential games with the homicidal chauffeur dynamics, Automatica, 40(12) (2004) 2059-2068.

[21] V. S. Patsko, V. L. Turova, Numerical investigation of the value function for the homicidal chauffeur problem with a more agile pursuer, Annals of the International Society of Dynamic Games, 10 (2009) 231-258.

[22] V. S. Patsko, V. L. Turova, Homicidal chauffeur game: History and modern studies, Annals of the International Society of Dynamic Games, 11 (2011) 227-251.

[23] M. Pontani, Differential games treated by a gradient–restoration approach. In Variational Analysis and Aerospace Engineering, pages 379-396. Springer, 2009.

[24] M. Pontani, Numerical solution of orbital combat games involving missiles and spacecraft, Dynamic Games and Applications, 1(4) (2011) 534-557.

[25] M. Pontani, B. A. Conway, Numerical solution of the three-dimensional orbital pursuit-evasion game, Journal of Guidance, Control, and Dynamics, 32(2) (2009) 474-487.

[26] S. Sun, Q. Zhang, R. Loxton, B. Li, Numerical solution of a pursuit-evasion differential game involving two spacecraft in low earth orbit. Journal of Industrial and Management Optimization, 11(4) (2015) 1127-1147.

[27] A. Wächter, L. T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Program., 106(1, Ser. A) (2006) 25-57.

Please cite this article using:

Zahra Yazdaniyan, Mostafa Shamsi, Maria do Rosário de Pinho, Zahra Foroozandeh, Heuristic artificial bee colony algorithm for solving the Homicidal Chauffeur differential game, AUT J. Math. Comput., 1(2) (2020) 153-163
DOI: 10.22060/ajmc.2019.16949.1025