Numerical solution of a pursuit-evasion differential game involving two spacecraft in low earth orbit

Article *in* Journal of Industrial and Management Optimization · October 2015

DOI: 10.3934/jimo.2015.11.1127

Songtao Sun
Shanghai
2 PUBLICATIONS **10** CITATIONS
SEE PROFILE

Bin Li
Sichuan University
48 PUBLICATIONS **371** CITATIONS
SEE PROFILE

JOURNAL OF INDUSTRIAL AND
MANAGEMENT OPTIMIZATION
Volume **11**, Number **4**, October **2015**

doi:10.3934/jimo.2015.11.1127

pp. **1127–1147**

NUMERICAL SOLUTION OF A PURSUIT-EVASION DIFFERENTIAL GAME INVOLVING TWO SPACECRAFT IN LOW EARTH ORBIT

Songtao Sun and Qiuhua Zhang*

Department of Astronautical Science and Mechanics
Harbin Institute of Technology, Harbin, China

Ryan Loxton and Bin Li

Department of Mathematics and Statistics
Curtin University, Perth, Australia

(Communicated by Song Wang)

Abstract. This paper considers a spacecraft pursuit-evasion problem taking place in low earth orbit. The problem is formulated as a zero-sum differential game in which there are two players, a pursuing spacecraft that attempts to minimize a payoff, and an evading spacecraft that attempts to maximize the same payoff. We introduce two associated optimal control problems and show that a saddle point for the differential game exists if and only if the two optimal control problems have the same optimal value. Then, on the basis of this result, we propose two computational methods for determining a saddle point solution: a semi-direct control parameterization method (SDCP method), which is based on a piecewise-constant control approximation scheme, and a hybrid method, which combines the new SDCP method with the multiple shooting method. Simulation results show that the proposed SDCP and hybrid methods are superior to the semi-direct collocation nonlinear programming method (SDCNLP method), which is widely used to solve pursuit-evasion problems in the aerospace field.

1. **Introduction.**

Pursuit-evasion problems, in which opposing decision-makers interact within a complex dynamic environment, are challenging problems to solve. In this paper, we consider a pursuit-evasion problem involving two spacecraft acting under independent control: the pursuing spacecraft attempts to minimize a certain payoff function, while the evading spacecraft attempts to maximize the same function. This pursuit-evasion problem can be formulated as a zero-sum differential game, of the type first introduced by Issacs in 1965 [13].

Friedman [8] and Berkovitz [2, 3] have derived necessary optimality conditions and established the existence of open-loop saddle point trajectories for zero-sum differential games. However, for differential games in the aerospace field, such as the spacecraft pursuit-evasion problem considered in this paper, it is usually impossible to derive analytical solutions [26, 27]. Hence, numerical methods are indispensable.

2010 *Mathematics Subject Classification.* Primary: 91A25; Secondary: 49M37, 49N90.

Key words and phrases. Pursuit-evasion problem, differential game, optimal control, nonlinear programming, multiple shooting method.

* Corresponding author.

Two types of numerical methods have been proposed for solving spacecraft pursuit-evasion problems: the multiple shooting approach [4], which involves solving a two-point boundary value problem formed from the necessary optimality conditions of the corresponding differential game, and the semi-direct approach [11], which involves replacing the differential game with a corresponding optimal control problem. Multiple shooting methods give solutions of high accuracy [4], but their regions of convergence are usually very narrow [28]. Semi-direct methods, on the other hand, converge for a wide range of initial guesses, but they are usually not as accurate as multiple shooting methods, because the optimal control formulation of the differential game cannot be solved exactly. One of the most well-known semi-direct methods is the semi-direct collocation nonlinear programming method (SDCNLP method) proposed by Conway and his collaborators [11, 12, 23, 24]. This method involves solving the optimal control formulation of the differential game using a collocation method. Although the SDCNLP method often works well in practice, it has two disadvantages: (i) solution accuracy is usually far less than that of the multiple shooting method; and (ii) the optimal control problem being solved is not precisely equivalent to the original differential game.

In this paper, we propose two new computational methods — a semi-direct control parameterization method (SDCP method) and a hybrid method — to overcome the disadvantages of the SDCNLP method proposed by Conway et al. Our new SDCP method involves introducing two optimal control problems corresponding to the differential game, whereas the SDCNLP method relies on just one. We give a precise proof to show that solving these optimal control problems in succession is equivalent to solving the differential game. Similar results are not available for the SDCNLP method. Moreover, instead of using the collocation method to solve the optimal control formulations of the differential game (as suggested by Conway et al.), we use the control parameterization method [16, 18, 19, 30]. The advantage of control parameterization is that it results in a finite-dimensional approximate problem of much smaller dimension [15, 17]. After introducing the SDCP method, we present a hybrid method that combines the SDCP and multiple shooting methods to improve solution accuracy.

The remainder of this paper is organized as follows. In Section 2, we formulate the spacecraft pursuit-evasion problem under consideration as a zero-sum differential game. Then, in Section 3, we give conditions for solution existence for this differential game, and also review the necessary conditions for optimality. The SDCP and hybrid methods are introduced in Section 4. Finally, in Section 5, these new methods are compared with the existing SDCNLP method via a numerical example.

2. The pursuit-evasion problem.

Consider a pursuit-evasion problem involving two spacecraft (the players). Let P denote pursuing spacecraft and let E denote the evading spacecraft. Each spacecraft is governed by a dynamic system with its own control variables. The dynamic systems for players P and E (which are uncoupled) are given by

$$\dot{x}_P(t) = g_P(t, x_P(t)) + B_P(t)u_P(t), \tag{1a}$$

$$\dot{x}_E(t) = g_E(t, x_E(t)) + B_E(t)u_E(t), \tag{1b}$$

where $x_P(t)$ is the state of P, $x_E(t)$ is the state of E, $u_P(t)$ is the control of P, and $u_E(t)$ is the control of E. Here, $g_P(t, x_P(t))$ and $g_E(t, x_E(t))$ are continuously differentiable vector-valued functions, and $B_P(t)$ and $B_E(t)$ are continuous matrix-valued functions depending on t. Furthermore, the controls $u_P(t)$ and $u_E(t)$ belong to given admissible strategy sets U_P and U_E, respectively.

The initial conditions for system (1) are

$$x_P(t_0) = x_P^0, \tag{2a}$$

$$x_E(t_0) = x_E^0, \tag{2b}$$

where t_0 is a given initial time.

We consider the following Mayer objective functional:

$$J(u_P, u_E) = \Phi(x_P(t_f), x_E(t_f)), \tag{3}$$

where the terminal time t_f is fixed and Φ is a given continuously differentiable function. The spacecraft pursuit-evasion problem is defined as follows.

Problem (G). *Subject to Eqs.* (1) *and* (2)*, choose $u_P \in U_P$ to minimize the payoff* (3)*, and choose $u_E \in U_E$ to maximize the payoff* (3).

The payoff function (3) can often be viewed as a measure of spacecraft separation. Thus, in Problem (G), the pursuing spacecraft wants to minimize the separation, while the evading spacecraft wants to maximize the separation.

3. Preliminary results.

Problem (G) is a zero-sum differential game in which the two players have competing objectives: player P wants to minimize the payoff, while player E wants to maximize the same payoff. Because of these competing objectives, the concept of “solution” to Problem (G) needs to be carefully considered. In this section, we give existence conditions for a solution of Problem (G). The necessary optimality conditions for Problem (G) will also be reviewed.

3.1. Solution existence for Problem (G).

Throughout this paper, we assume that the following condition is satisfied.

Assumption 3.1. *The admissible strategy sets U_P and U_E are compact sets in some metric space, and J is a continuous function from $U_P \times U_E \to \mathbb{R}$.*

To solve Problem (G), we seek control strategies $u_P^* \in U_P$ and $u_E^* \in U_E$ such that

$$\max_{u_E \in U_E} J(u_P^*, u_E) = J(u_P^*, u_E^*) = \min_{u_P \in U_P} J(u_P, u_E^*). \tag{4}$$

Any pair (u_P^*, u_E^*) satisfying (4) is called a saddle point for Problem (G). Note that there could be multiple saddle points, and any such saddle point is a valid solution of Problem (G) [1]. Note also that if Eq. (4) holds, then for any pair of control strategies $(u_P, u_E) \in U_P \times U_E$,

$$J(u_P^*, u_E) \leq J(u_P^*, u_E^*) \leq J(u_P, u_E^*).$$

To proceed further, we need the following definitions.

Definition 3.2. For Problem (G), the set of best reply strategies for player P, given a fixed strategy $u_E \in U_E$ for player E, is defined as

$$S_P(u_E) = \Big\{ u'_P \in U_P : \, J(u'_P, u_E) = \min_{u_P} J(u_P, u_E) \Big\}.$$

Similarly, the set of best reply strategies for player E, given a fixed strategy $u_P \in U_P$ for player P, is defined as

$$S_E(u_P) = \Big\{ u'_E \in U_E : \, J(u_P, u'_E) = \max_{u_E} J(u_P, u_E) \Big\}.$$

Note that $S_P(u_E)$ contains the best reply strategies for P when E uses the fixed strategy u_E, and $S_E(u_P)$ contains the analogous strategies for E.

Definition 3.3. For any given $u_E \in U_E$, let $u'_P(u_E) \in S_P(u_E)$ be a corresponding best reply strategy for player P. Similarly, for any given $u_P \in U_P$, let $u'_E(u_P) \in S_E(u_P)$ be a corresponding best reply strategy for player E. Then the lower value of the zero-sum game is defined as

$$V^- = \max_{u_E} J(u'_P(u_E), u_E) = \max_{u_E} \min_{u_P} J(u_P, u_E).$$

Similarly, the upper value of the zero-sum game is defined as

$$V^+ = \min_{u_P} J(u_P, u'_E(u_P)) = \min_{u_P} \max_{u_E} J(u_P, u_E).$$

Definition 3.4. If V^+ and V^- are both finite, and $V^+ = V^- = V$, then V is called the value of the differential game.

Before we can characterize saddle point existence in Problem (G), we need the following lemmas.

Lemma 3.5. *The lower value of the zero-sum game is no greater than the upper value of the zero-sum game. Mathematically,*

$$V^- \leq V^+.$$

Proof. According to the definition of V^-, for each $\varepsilon > 0$, there exists a corresponding strategy $u_E^\varepsilon \in U_E$ such that

$$V^- - \varepsilon \leq J\left(u'_P(u_E^\varepsilon), u_E^\varepsilon\right), \tag{5}$$

where $u'_P(u_E^\varepsilon) \in S_P(u_E^\varepsilon)$ is a best reply strategy for P corresponding to u_E^ε. Then,

$$V^- - \varepsilon \leq J\left(u'_P(u_E^\varepsilon), u_E^\varepsilon\right) = \min_{u_P} J(u_P, u_E^\varepsilon) \leq \min_{u_P} \max_{u_E} J(u_P, u_E) = V^+.$$

Since $\varepsilon > 0$ was selected arbitrarily, we immediately deduce $V^- \leq V^+$. This completes the proof. □

Lemma 3.6. *Let $\left\{u_P^n\right\}$ be a sequence of control strategies for player P, and let $\left\{u'_E(u_P^n)\right\}$ be a corresponding sequence of best reply strategies for player E. Suppose that $(u_P^n, u'_E(u_P^n)) \to (\bar{u}_P, \bar{u}_E)$ as $n \to \infty$, where $(\bar{u}_P, \bar{u}_E) \in U_P \times U_E$. Then $\bar{u}_E$ is a best reply strategy for player E corresponding to $\bar{u}_P$.*

Proof. According to Definition 3.2, for each $n \geq 1$,

$$J(u_P^n, u'_E(u_P^n)) \geq J(u_P^n, u'_E(\bar{u}_P)).$$

Thus, since J is continuous and $(u_P^n, u'_E(u_P^n)) \to (\bar{u}_P, \bar{u}_E)$ as $n \to \infty$,

$$J(\bar{u}_P, \bar{u}_E) \geq J(\bar{u}_P, u'_E(\bar{u}_P)).$$

There are two cases to consider:

(i) $J(\bar{u}_P, \bar{u}_E) > J(\bar{u}_P, u'_E(\bar{u}_P))$; and
(ii) $J(\bar{u}_P, \bar{u}_E) = J(\bar{u}_P, u'_E(\bar{u}_P))$.

But case (i) is clearly impossible, as $u'_E(\bar{u}_P)$ is a best reply strategy for player E corresponding to $\bar{u}_P$. Thus, we must have

$$J(\bar{u}_P, \bar{u}_E) = J(\bar{u}_P, u'_E(\bar{u}_P)) = \max_{u_E} J(\bar{u}_P, u_E).$$

This shows that $\bar{u}_E \in S_E(\bar{u}_P)$, as required. □

We also have the following analogue to Lemma 3.6 for a sequence of control strategies for player E.

Lemma 3.7. *Let $\left\{u_E^n\right\}$ be a sequence of control strategies for player E, and let $\left\{u'_P(u_E^n)\right\}$ be a corresponding sequence of best reply strategies for player P. Suppose that $(u'_P(u_E^n), u_E^n) \to (\bar{u}_P, \bar{u}_E)$ as $n \to \infty$, where $(\bar{u}_P, \bar{u}_E) \in U_P \times U_E$. Then $\bar{u}_P$ is a best reply strategy for player P corresponding to $\bar{u}_E$.*

Proof. Similar to the proof of Lemma 3.6. □

We now present the following existence condition, which characterizes the solution of Problem (G).

Theorem 3.8. *Under Assumption 3.1, a saddle point for Problem (G) exists if and only if $V^- = V^+ = V$.*

Proof. Suppose that a saddle point (u_P^*, u_E^*) exists. Since we already know from Lemma 3.5 that $V^- \leq V^+$, it remains to show that $V^- \geq V^+$. We have

$$V^- = \max_{u_E} \min_{u_P} J(u_P, u_E) \geq \min_{u_P} J(u_P, u_E^*) = \max_{u_E} J(u_P^*, u_E) \geq \min_{u_P} \max_{u_E} J(u_P, u_E) \\ = V^+.$$

Conversely, suppose that $V^- = V^+ = V$. For each $\varepsilon > 0$, there exists a corresponding u_E^ε such that Eq. (5) holds. Similarly, there exists a corresponding u_P^ε such that

$$J\left(u_P^\varepsilon, u'_E(u_P^\varepsilon)\right) \leq V^+ + \varepsilon, \tag{6}$$

where $u'_E(u_P^\varepsilon)$ is a best reply strategy for the evader corresponding to u_P^ε.

Since U_P and U_E are compact (recall Assumption 3.1), we can choose a subsequence $\varepsilon_n \to 0$ such that the corresponding subsequence of strategy pairs $(u_P^{\varepsilon_n}, u_E^{\varepsilon_n})$, and the corresponding sequence of best reply strategies $(u'_P(u_E^{\varepsilon_n}), u'_E(u_P^{\varepsilon_n}))$, both have limit points. That is, there exists strategy pairs $(u_P^*, u_E^*) \in U_P \times U_E$ and $(\bar{u}_P^*, \bar{u}_E^*) \in U_P \times U_E$ such that, as $n \to \infty$,

$$(u_P^{\varepsilon_n}, u_E^{\varepsilon_n}) \to (u_P^*, u_E^*)\,, \quad (u'_P(u_E^{\varepsilon_n}), u'_E(u_P^{\varepsilon_n})) \to (\bar{u}_P^*, \bar{u}_E^*).$$

Therefore, from (5) and (6),

$$V^- - \varepsilon_n \leq J\left(u'_P(u_E^{\varepsilon_n}), u_E^{\varepsilon_n}\right) \leq J\left(u_P^{\varepsilon_n}, u_E^{\varepsilon_n}\right) \\ \leq J\left(u_P^{\varepsilon_n}, u'_E(u_P^{\varepsilon_n})\right) \leq V^+ + \varepsilon_n.$$

Since the payoff J is continuous on $U_P \times U_E$, taking the limit as $n \to \infty$ gives

$$V^- \leq J\left(\bar{u}_P^*, u_E^*\right) \leq J\left(u_P^*, u_E^*\right) \leq J\left(u_P^*, \bar{u}_E^*\right) \leq V^+.$$

According to Lemmas 3.6 and 3.7, $\bar{u}_P^*$ is a best reply strategy for P corresponding to u_E^*, and $\bar{u}_E^*$ is a best reply strategy for E corresponding to u_P^*. Thus,

$$V^- \leq \min_{u_P} J(u_P, u_E^*) = J\left(\bar{u}_P^*, u_E^*\right) \leq J\left(u_P^*, u_E^*\right) \\ \leq J\left(u_P^*, \bar{u}_E^*\right) = \max_{u_E} J(u_P^*, u_E) \leq V^+.$$

Since $V^- = V^+$, this shows that (u_P^*, u_E^*) is a saddle point for Problem (G). □

3.2. **Necessary optimality conditions for Problem (G).** In this section, we briefly review the necessary optimality conditions for Problem (G). For more details, we refer readers to [8].

Define the Hamiltonian H as

$$\begin{aligned} H(t, x_P, x_E, u_P, u_E, \lambda_P, \lambda_E) = \lambda_P^T g_P(t, x_P) + \lambda_P^T B_P(t) u_P \\ + \lambda_E^T g_E(t, x_E) + \lambda_E^T B_E(t) u_E, \end{aligned}$$

where λ_P and λ_E are the adjoint variables for P and E, respectively.

Any saddle point solution for Problem (G) must satisfy the following equations:

$$u_P^*(t) = \arg\min_{u_P \in U_P} H(t, x_P, x_E, u_P, u_E, \lambda_P, \lambda_E) = \arg\min_{u_P \in U_P} \lambda_P^T(t) B_P(t)\, u_P(t), \quad \text{(7a)}$$

$$u_E^*(t) = \arg\max_{u_E \in U_E} H(t, x_P, x_E, u_P, u_E, \lambda_P, \lambda_E) = \arg\max_{u_E \in U_E} \lambda_E^T(t) B_E(t)\, u_E(t), \quad \text{(7b)}$$

where the adjoint variables λ_P and λ_E satisfy the differential equations,

$$\dot{\lambda}_P(t) = -\left(\frac{\partial g_P(t, x_P(t))}{\partial x_P}\right)^T \lambda_P(t), \quad \text{(8a)}$$

$$\dot{\lambda}_E(t) = -\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \lambda_E(t), \quad \text{(8b)}$$

with the terminal conditions

$$\lambda_P(t_f) = \left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_P}\right)^T, \quad \text{(9a)}$$

$$\lambda_E(t_f) = \left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_E}\right)^T. \quad \text{(9b)}$$

Note that, in view of equation (7a), the optimal strategy for the pursuer depends solely on λ_P. Similarly, the optimal strategy for the evader depends solely on λ_E. If the optimal strategies for players P and E can be expressed as explicit functions of λ_P and λ_E, then these functions can be substituted into (1) and (8) to yield a two-point boundary value problem (TPBVP) consisting of (1), (2), (8) and (9). In principle, the multiple shooting method [28] can be applied to solve this TPBVP. However, good initial approximations of the optimal trajectories and adjoint variables are required. We refer readers to [28] for more details on the multiple shooting method. In this paper, we propose two new computational methods for solving Problem (G), the details of which are discussed in the next section.

4. **Numerical solution methods.**

In this section, we propose two numerical methods for solving Problem (G). In contrast with the traditional semi-direct methods in the literature (see, for example, the work by Horie [11]), our new methods involve solving two optimal control problems instead of one. The advantage of our new approach is that it is guaranteed to yield a saddle point solution of Problem (G) (assuming one exists), whereas the optimal control formulation in the traditional semi-direct approach is not precisely equivalent to the original differential game.

4.1. **Optimal control formulations.** We assume that the maximization condition (7b) can be used to express the optimal strategy for player E as a unique function of the adjoint variable $\lambda_E(t)$:

$$u_E^*(t) = \bar{u}_E^*(\lambda_E(t)). \tag{10}$$

Substituting (10) into (1), and appending the adjoint dynamics for λ_E, yields the following extended system:

$$\dot{x}_P(t) = g_P(t, x_P(t)) + B_P(t)u_P(t), \tag{11a}$$

$$\dot{x}_E(t) = g_E(t, x_E(t)) + B_E(t)\bar{u}_E^*(\lambda_E(t)), \tag{11b}$$

$$\dot{\lambda}_E(t) = -\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \lambda_E(t), \tag{11c}$$

with the initial conditions

$$x_P(t_0) = x_P^0, \tag{12a}$$

$$x_E(t_0) = x_E^0, \tag{12b}$$

and the terminal conditions

$$\lambda_E(t_f) = \left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_E}\right)^T. \tag{13}$$

Note that u_P completely determines the state trajectory for x_P via (11a) and (12a). By substituting the resulting value of $x_P(t_f)$ into (13), we obtain a corresponding two-point boundary value problem for x_E and λ_E consisting of (11b), (11c), (12b) and (13). We now define the following optimal control problem.

Problem (O$_P$). *Subject to the extended system* (11)-(13), *choose* $u_P \in U_P$ *to minimize the payoff* J.

In Problem (O$_P$), λ_E is viewed as an additional state variable subject to the terminal state constraint (13). The following result reveals the relationship between Problem (O$_P$) and Problem (G).

Proposition 4.1. *The optimal payoff value of Problem (O$_P$) is equal to the upper value of Problem (G).*

Proof. Suppose player P uses the fixed strategy $u_P \in U_P$. Then player E's goal is to solve the following optimal control problem to determine the corresponding best reply strategy:

$$\max_{u_E} \Phi(x_P(t_f), x_E(t_f)),$$

subject to

$$\begin{cases} \dot{x}_E(t) = g_E(t, x_E(t)) + B_E(t)u_E(t), \\ x_E(t_0) = x_E^0, \end{cases}$$

where x_P is the solution of (11a) and (12a) corresponding to the fixed strategy u_P. The optimal control for this problem satisfies,

$$u_E^*(t) = \arg\max_{u_E} \left\{ \lambda_E^T(t) B_E(t) u_E(t) \right\} = \bar{u}_E^*(\lambda_E(t)),$$

where the adjoint variable λ_E satisfies

$$\begin{cases} \dot{\lambda}_E(t) = -\left(\dfrac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \lambda_E(t), \\ \lambda_E(t_f) = \left(\dfrac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_E}\right)^T. \end{cases}$$

Since $\bar{u}_E^*$ is the unique control that satisfies these conditions, it must be the best reply strategy corresponding to u_P. Thus, the optimal value for Problem (O$_P$) is

$$\min_{u_P} J(u_P, \bar{u}_E^*) = \min_{u_P} \max_{u_E} J(u_P, u_E) = V^+,$$

as required. This completes the proof. □

We now introduce the analogous optimal control problem for player E. To do this, we assume that the minimization condition (7a) can be used to derive the optimal strategy function for player P as a unique function of the adjoint variable λ_P:

$$u_P^*(t) = \bar{u}_P^*(\lambda_P(t)). \tag{14}$$

Substituting (14) into (1) and appending the adjoint dynamics for λ_P yields the following extended system:

$$\dot{x}_P(t) = g_P(t, x_P(t)) + B_P(t)\bar{u}_P^*(\lambda_P(t)), \tag{15a}$$

$$\dot{x}_E(t) = g_E(t, x_E(t)) + B_E(t)u_E(t), \tag{15b}$$

$$\dot{\lambda}_P(t) = -\left(\frac{\partial g_P(t, x_P(t))}{\partial x_P}\right)^T \lambda_P(t), \tag{15c}$$

with the boundary conditions

$$x_P(t_0) = x_P^0, \tag{16a}$$

$$x_E(t_0) = x_E^0, \tag{16b}$$

$$\lambda_P(t_f) = \left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_P}\right)^T. \tag{16c}$$

The following optimal control problem is the analogue of Problem (O$_P$) for player E.

Problem (O$_E$). *Subject to the extended system* (15)-(16)*, choose $u_E \in U_E$ to maximize the payoff J.*

We have the following result, which links Problem (O$_E$) with Problem (G).

Proposition 4.2. *The optimal payoff value of Problem (O$_E$) is equal to the lower value of Problem (G).*

Proof. Similar to the proof of Proposition 4.1. □

Combining Propositions 4.1 and 4.2 with Theorem 3.8 yields the following result.

Proposition 4.3. *A saddle point for Problem (G) exists if and only if the optimal payoff values of Problems (O$_P$) and (O$_E$) are equal.*

We now present the main result for this section.

Theorem 4.4. *Let u_P^* and u_E^* be optimal controls for Problems (O$_P$) and (O$_E$), respectively. If the optimal payoff values of Problems (O$_P$) and (O$_E$) are equal, then (u_P^*, u_E^*) is a saddle point for Problem (G).*

Proof. Suppose that the optimal payoff values for Problems (O$_P$) and (O$_E$) are equal. Then it follows from Proposition 4.1, Proposition 4.2, and Theorem 3.8 that there exists a saddle point for Problem (G).

Let $\bar{u}_E^*$ be the best reply strategy for player E corresponding to u_P^*, and let $\bar{u}_P^*$ be the best reply strategy for player P corresponding to u_E^*. According to Propositions 4.1 and 4.2,

$$V^+ = \min_{u_P} J(u_P, u'_E(u_P)) = J(u_P^*, \bar{u}_E^*) = \max_{u_E} J(u_P^*, u_E),$$

and

$$V^- = \max_{u_E} J(u'_P(u_E), u_E) = J(\bar{u}_P^*, u_E^*) = \min_{u_P} J(u_P, u_E^*).$$

Since $V^+ = V^-$, we have

$$J(u_P^*, \bar{u}_E^*) = V^+ = V^- = J(\bar{u}_P^*, u_E^*). \tag{17}$$

Furthermore, since $\bar{u}_P^* \in S_P(u_E^*)$ and $\bar{u}_E^* \in S_E(u_P^*)$,

$$J(\bar{u}_P^*, u_E^*) = \min_{u_P} J(u_P, u_E^*) \leq J(u_P^*, u_E^*) \leq \max_{u_E} J(u_P^*, u_E) = J(u_P^*, \bar{u}_E^*). \tag{18}$$

Combining (17) and (18) yields

$$\min_{u_P} J(u_P, u_E^*) = J(u_P^*, u_E^*) = \max_{u_E} J(u_P^*, u_E).$$

This shows that (u_P^*, u_E^*) is a saddle point for Problem (G). □

On the basis of Theorem 4.4, Problem (G) can be solved using the following procedure:

1. Solve Problem (O$_P$) to obtain an optimal control u_P^*.
2. Solve Problem (O$_E$) to obtain an optimal control u_E^*.
3. Compare the optimal payoff values for Problems (O$_P$) and (O$_E$). If the optimal payoff values are equal, then (u_P^*, u_E^*) is a saddle point for Problem (G). Otherwise, no saddle point exists.

To implement this procedure successfully, we require a numerical method for solving Problems (O$_P$) and (O$_E$). This is the focus of the next section.

4.2. **Semi-direct control parameterization method.** Both Problems (O$_P$) and (O$_E$) are optimal control problems in canonical form. Since these problems are similar, we will restrict our attention to Problem (O$_P$) in this section.

To solve Problem (O$_P$) numerically, we will apply the control parameterization technique [19, 29]. This technique involves partitioning the time horizon $[\,t_0, t_f]$ into a set of subintervals $[\tau_{j-1}, \tau_j)$, where τ_j, $j = 1, \ldots, q$, are fixed knot points such that $t_0 = \tau_0 < \tau_1 < \cdots < \tau_q = t_f$. Then, the control function u_P is approximated by a piecewise-constant function as follows:

$$u_P(t) \approx \sum_{j=1}^{q} \sigma_j \chi_{[\,\tau_{j-1}, \tau_j)}(t), \tag{19}$$

where $\chi_{[\,\tau_{j-1}, \tau_j)}(t)$ is the characteristic function defined by

$$\chi_{[\,\tau_{j-1}, \tau_j)}(t) = \begin{cases} 1, & \text{if } t \in [\,\tau_{j-1}, \tau_j)\,, \\ 0, & \text{otherwise.} \end{cases}$$

Substituting (19) into (11) gives

$$\left.\begin{aligned} \dot{x}_P(t) &= g_P(t, x_P(t)) + B_P(t)\sigma_j \\ \dot{x}_E(t) &= g_E(t, x_E(t)) + B_E(t)\bar{u}_E^*(\lambda_E(t)) \\ \dot{\lambda}_E(t) &= -\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \lambda_E(t) \end{aligned}\right\} \quad t \in [\tau_{j-1}, \tau_j), \quad j = 1, \ldots, q. \tag{20}$$

Since the initial value of λ_E is unknown, we introduce a decision vector ζ such that $\lambda_E(t_0) = \zeta$. Let $\mathscr{E}$ denote the set consisting of all suitable vectors for ζ. Then the initial conditions for (20) are

$$x_P(t_0) = x_P^0, \tag{21a}$$

$$x_E(t_0) = x_E^0, \tag{21b}$$

$$\lambda_E(t_0) = \zeta. \tag{21c}$$

Let

$$\Psi(x_P, x_E, \lambda_E) = \lambda_E - \left(\frac{\partial \Phi(x_P, x_E)}{\partial x_E}\right)^T,$$

$$\sigma = \left[(\sigma_1)^T, (\sigma_2)^T, \ldots, (\sigma_p)^T\right]^T. \tag{22}$$

Then, the terminal condition (13) can be expressed as the following terminal state constraint:

$$G_q(\sigma, \zeta) = \left\|\Psi(x_P(t_f), x_E(t_f), \lambda_E(t_f))\right\|_2^2 = 0, \tag{23}$$

where $\|\cdot\|_2$ denotes the Euclidean norm.

Furthermore, let Ξ denote the set of all vectors in the form of (22) such that the corresponding approximate control defined by (19) belongs to U_P.

In view of the dynamic system (20)-(21), the payoff can be expressed as a function of σ and ζ:

$$J_q(\sigma, \zeta) = \Phi(x_P(t_f), x_E(t_f)). \tag{24}$$

We now define the following finite-dimensional approximation of Problem (O_P).

Problem (O_P^q). *Subject to the system given by (20)-(21), and the terminal state constraint (23), choose vectors $\sigma \in \Xi$ and $\zeta \in \mathscr{E}$ to minimize the payoff (24).*

Problem (O_P^q) can be solved as a nonlinear programming problem using standard constrained optimization methods, such as sequential quadratic programming (SQP). To do this, we need the gradient formulas for the payoff J_q and the constraint function G_q. These gradient formulas are given in the following theorems, which can be proved in a similar manner to Theorem 5.2.1 in [29].

Theorem 4.5. *The gradients of the cost function $J_q(\sigma, \zeta)$ with respect to σ and ζ are given by*

$$\frac{\partial J_q(\sigma, \zeta)}{\partial \sigma_j} = \int_{\tau_{j-1}}^{\tau_j} \eta_P^T(t) B_P(t) \mathrm{d}t, \quad j = 1, \ldots, q,$$

$$\frac{\partial J_q(\sigma, \zeta)}{\partial \zeta} = \eta_\lambda^T(t_0),$$

where the adjoint variable $\eta = [\eta_P^T, \eta_E^T, \eta_\lambda^T]^T$ *satisfies the differential equations*

$$\begin{aligned}
\dot{\eta}_P(t) &= -\left(\frac{\partial g_P(t, x_P(t))}{\partial x_P}\right)^T \eta_P(t),\\
\dot{\eta}_E(t) &= -\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \eta_E(t) + \left\{\frac{\partial}{\partial x_E}\left(\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \lambda_E(t)\right)\right\}^T \eta_\lambda(t),\\
\dot{\eta}_\lambda(t) &= -\left(\frac{\partial \bar{u}_E^*(\lambda_E(t))}{\partial \lambda_E}\right)^T B_E^T(t)\eta_E(t) + \left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)\eta_\lambda(t),
\end{aligned}$$

with the terminal conditions

$$\begin{aligned}
\eta_P(t_f) &= \left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_P}\right)^T,\\
\eta_E(t_f) &= \left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_E}\right)^T,\\
\eta_\lambda(t_f) &= 0.
\end{aligned}$$

Theorem 4.6. *The gradients of the constraint function* $G_q(\sigma, \zeta)$ *with respect to* σ *and* ζ *are given by*

$$\begin{aligned}
\frac{\partial G_q(\sigma, \zeta)}{\partial \sigma_j} &= \int_{\tau_{j-1}}^{\tau_j} \mu_P^T(t) B_P(t)\mathrm{d}t, \quad j = 1, \dots, q,\\
\frac{\partial G_q(\sigma, \zeta)}{\partial \zeta} &= \mu_\lambda^T(t_0),
\end{aligned}$$

where the adjoint variable $\mu = [\mu_P^T, \mu_E^T, \mu_\lambda^T]^T$ *satisfies the differential equations*

$$\begin{aligned}
\dot{\mu}_P(t) &= -\left(\frac{\partial g_P(t, x_P(t))}{\partial x_P}\right)^T \mu_P(t),\\
\dot{\mu}_E(t) &= -\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \mu_E(t) + \left\{\frac{\partial}{\partial x_E}\left(\left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)^T \lambda_E(t)\right)\right\}^T \mu_\lambda(t),\\
\dot{\mu}_\lambda(t) &= -\left(\frac{\partial \bar{u}_E^*(\lambda_E(t))}{\partial \lambda_E}\right)^T B_E^T(t)\mu_E(t) + \left(\frac{\partial g_E(t, x_E(t))}{\partial x_E}\right)\mu_\lambda(t),
\end{aligned}$$

with the terminal conditions

$$\begin{aligned}
\mu_P(t_f) &= -2\left[\frac{\partial}{\partial x_P}\left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_E}\right)^T\right]^T \Psi(x_P(t_f), x_E(t_f), \lambda_E(t_f)),\\
\mu_E(t_f) &= -2\left[\frac{\partial}{\partial x_E}\left(\frac{\partial \Phi(x_P(t_f), x_E(t_f))}{\partial x_E}\right)^T\right]^T \Psi(x_P(t_f), x_E(t_f), \lambda_E(t_f)),\\
\mu_\lambda(t_f) &= 2\Psi(x_P(t_f), x_E(t_f), \lambda_E(t_f)).
\end{aligned}$$

The gradient formulas in Theorems 4.5 and 4.6 can be incorporated into gradient-based optimization techniques such as SQP to solve Problem (O_P^q) numerically. When q is large, the optimal piecewise-constant control is a good approximation of the optimal control for Problem (O_P). In fact, it can be shown that, if $u_P^{q,*}$ is the optimal piecewise-constant control generated from the solution of Problem (O_P^q), and u_P^* is an optimal control for Problem (O_P), then

$$\lim_{q\to\infty} J(u_P^{q,*}) = J(u_P^*).$$

See [20, 21, 29] for proofs of this result under various conditions.

Gradient-based optimization methods can only guarantee local optimal solutions. Thus, the initial point used to start the optimization process is crucial. In the numerical simulations in Section 5.2, we use the multi-objective genetic algorithm (MOGA) in [7] to determine a good starting point.

4.3. **Hybrid method.**

As with any semi-direct method, the key limitation of the SDCP method is that the optimal control formulations of the differential game can, in general, only be solved numerically. Indeed, although the control parameterization method generates approximate solutions that converge to the true solution as the number of subintervals approaches infinity, the limiting solution is usually never achievable in practice. Accordingly, we now propose a hybrid method that combines the SDCP method with the multiple shooting method to improve solution accuracy. In this hybrid method, the SDCP method is first used as pre-processor to generate an approximate saddle point. This approximate saddle point is then used as the initial guess for the multiple shooting method. By using this hybrid strategy, the strong convergence properties of the semi-direct method and the high accuracy of the multiple shooting method can both be achieved. The hybrid algorithm is summarized below.

1. Solve Problem (O_P) numerically using the control parameterization method described in Section 4.2. Let u_P^* denote the approximate optimal control of Problem (O_P) and let x_E^* and λ_E^* denote the corresponding trajectories for player E and the adjoint variable for player E, respectively.
2. Solve Problem (O_E) numerically using the control parameterization method described in Section 4.2. Let u_E^* denote the approximate optimal control of Problem (O_E) and let x_P^* and λ_P^* denote the corresponding trajectories for player P and the adjoint variable for player P, respectively.
3. If the difference between the optimal payoff values of Problems (O_P) and (O_E) exceeds a given tolerance, then stop: no saddle point exists. Otherwise, take (u_P^*, u_E^*) as an approximate saddle point.
4. Starting with the initial trajectories x_P^*, x_E^*, λ_P^* and λ_E^*, use the multiple shooting method to solve the TPBVP described by Eqs. (1), (2), (8) and (9).

5. **Numerical example.**

In this section, we give a specific model for the spacecraft pursuit-evasion problem in low earth orbit. We then solve this problem using three methods: the new SDCP and hybrid methods presented in Section 4, and the existing SDCNLP method in reference [24].

5.1. **Mathematical model.**

We consider a pursuit-evasion scenario involving two spacecraft distributed in low earth orbit (altitude of between 160 km and 2000 km). The coordinate system for the two spacecraft is shown in Fig. 1. The motion of the ith spacecraft ($i = P$ for the pursuer and $i = E$ for the evader) is described by the

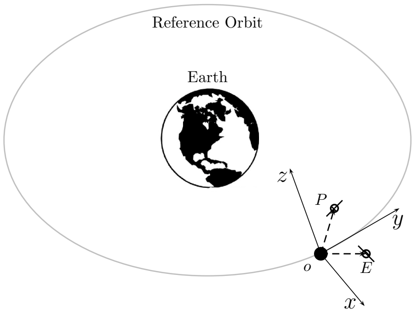


FIGURE 1. Coordinate system for the spacecraft pursuit-evasion problem.

following differential equations:

$$\ddot{s}_i^x(t) = \frac{2\mu}{r^3(t)} s_i^x(t) + 2\omega(t)\dot{s}_i^y(t) + \dot{\omega}(t)s_i^y(t) + \omega^2(t)s_i^x(t) + T_i\, u_i^x(t), \tag{25a}$$

$$\ddot{s}_i^y(t) = -\frac{\mu}{r^3(t)} s_i^y(t) - 2\omega(t)\dot{s}_i^x(t) - \dot{\omega}(t)s_i^x(t) + \omega^2(t)s_i^y(t) + T_i\, u_i^y(t), \tag{25b}$$

$$\ddot{s}_i^z(t) = -\omega^2(t)s_i^z(t) + T_i\, u_i^z(t), \tag{25c}$$

where s_i^x, s_i^y and s_i^z define the position of the ith spacecraft with respect to the three coordinate axes, and u_i^x, u_i^y and u_i^z are control variables representing the thrust directions along the three coordinate axes. Eqs. (25) are known as the Clohessy-Wiltshire equations [5]. In these equations, T_i is the thrust-to-mass ratio (constant) for the ith spacecraft, $\omega(t)$ is the angular velocity of the origin O, μ is the earth planetary constant, and $r(t)$ is the distance from the center of the earth to the origin of the coordinate system.

The control variables for player i are subject to the following constraints:

$$\sqrt{u_i^x(t)^2 + u_i^y(t)^2 + u_i^z(t)^2} \leq 1, \quad t \in [t_0, t_f].$$

Let v_i^x, v_i^y and v_i^z define the velocity of the ith spacecraft with respect to the three coordinate axes. Then, the differential equations (25) can be written as the following first-order dynamic system:

$$\dot{s}_i^x(t) = v_i^x(t), \tag{26a}$$

$$\dot{s}_i^y(t) = v_i^y(t), \tag{26b}$$

$$\dot{s}_i^z(t) = v_i^z(t), \tag{26c}$$

$$\dot{v}_i^x(t) = \frac{2\mu}{r^3(t)} s_i^x(t) + 2\omega(t)v_i^y(t) + \dot{\omega}(t)s_i^y(t) + \omega^2(t)s_i^x(t) + T_i\, u_i^x(t), \tag{26d}$$

$$\dot{v}_i^y(t) = -\frac{\mu}{r^3(t)} s_i^y(t) - 2\omega(t)v_i^x(t) - \dot{\omega}(t)s_i^x(t) + \omega^2(t)s_i^y(t) + T_i\, u_i^y(t), \tag{26e}$$

$$\dot{v}_i^z(t) = -\omega^2(t)s_i^z(t) + T_i\, u_i^z(t). \tag{26f}$$

The payoff function in this example measures the separation distance between the two spacecraft at the terminal time:

$$J = \frac{1}{2}\left[\left(s_P^x(t_f) - s_E^x(t_f)\right)^2 + \left(s_P^y(t_f) - s_E^y(t_f)\right)^2 + \left(s_P^z(t_f) - s_E^z(t_f)\right)^2\right]. \tag{27}$$

The spacecraft pursuit-evasion problem can be stated as follows: subject to the dynamic system (26) with $i = P$, the pursuing spacecraft wants to choose its control variables u_P^x, u_P^y and u_P^z to minimize the separation distance (27) at the terminal time, and subject to the dynamic system (26) with $i = E$, the evading spacecraft wants to choose its control variables u_E^x, u_E^y and u_E^z to maximize the separation distance (27) at the terminal time.

Using the minimization condition (7a), the optimal control functions for player P are obtained by solving the following optimization problem:

$$\min_{u_P^x, u_P^y, u_P^z} \lambda_P^x u_P^x + \lambda_P^y u_P^y + \lambda_P^z u_P^z$$

$$\text{subject to: } \sqrt{(u_P^x)^2 + (u_P^y)^2 + (u_P^z)^2} \leq 1,$$

where λ_P^x, λ_P^y and λ_P^z are the adjoint variables corresponding to v_P^x, v_P^y and v_P^z, respectively. Solving this optimization problem yields the optimal controls for player P in terms of the adjoint variables:

$$\begin{cases} u_P^x(t) = -\dfrac{\lambda_P^x(t)}{\sqrt{[\lambda_P^x(t)]^2 + [\lambda_P^y(t)]^2 + [\lambda_P^z(t)]^2}}, \\ u_P^y(t) = -\dfrac{\lambda_P^y(t)}{\sqrt{[\lambda_P^x(t)]^2 + [\lambda_P^y(t)]^2 + [\lambda_P^z(t)]^2}}, \\ u_P^z(t) = -\dfrac{\lambda_P^z(t)}{\sqrt{[\lambda_P^x(t)]^2 + [\lambda_P^y(t)]^2 + [\lambda_P^z(t)]^2}}. \end{cases}$$

Similarly, the optimal controls for player E are

$$\begin{cases} u_E^x(t) = \dfrac{\lambda_E^x(t)}{\sqrt{[\lambda_E^x(t)]^2 + [\lambda_E^y(t)]^2 + [\lambda_E^z(t)]^2}}, \\ u_E^y(t) = \dfrac{\lambda_E^y(t)}{\sqrt{[\lambda_E^x(t)]^2 + [\lambda_E^y(t)]^2 + [\lambda_E^z(t)]^2}}, \\ u_E^z(t) = \dfrac{\lambda_E^z(t)}{\sqrt{[\lambda_E^x(t)]^2 + [\lambda_E^y(t)]^2 + [\lambda_E^z(t)]^2}}. \end{cases}$$

Using the formulas given above for the optimal control functions for players P and E, we can readily obtain the optimal control problems corresponding to the differential game (i.e., Problems (O_P) and (O_E)).

5.2. **Numerical results.** According to Eq. (25), the in-plane motion (i.e., the x-y subsystem) and the out-of-plane motion (i.e., the z subsystem) are independent and can be considered separately. Thus, we assume that the pursuit-evasion problem occurs in a co-planar orbit and $r(t) = r_0$, where r_0 is the sum of the radius of the earth (6371 km) and the altitude of the origin O. We also assume that the angular velocity $\omega(t)$ is constant and equals $\sqrt{\mu/r_0^3}$, where $\mu = 3.986 \times 10^5$ km^3 is the earth planetary constant.

We consider eight versions of the pursuit-evasion problem. These versions differ in the initial values of the x-y state variables, the altitude of the origin, and the values of T_P and T_E. Each version of the problem has a fixed terminal time of $t_f = 500$ seconds.

In versions 1-4 of the pursuit-evasion problem, the altitude of the origin is 500 km; in versions 5-8, the altitude of the origin is 1000 km. The initial values of the x-y state variables for each version are given in Table 1. Moreover, Table 2 gives the

TABLE 1. Initial values of the x-y state variables

Version	s_P^x	s_P^y	v_P^x	v_P^y	s_E^x	s_E^y	v_E^x	v_E^y
1	0	0	0	0	9.958	24.01	−0.027	−0.0056
2	−1.046	−119.9	0.133	−0.001	9.906	36.02	−0.04	−0.006
3	−0.01	11.99	−0.013	0	19.74	60.13	−0.066	−0.011
4	−0.094	35.97	−0.039	−0.0001	19.83	48.11	−0.053	−0.011
5	0	0	0	0	9.988	12.88	−0.013	−0.005
6	−0.281	64.32	−0.064	−0.0002	9.955	25.76	-0.026	−0.005
7	20.00	0	0	−0.01	9.719	64.41	−0.0641	−0.006
8	19.99	12.89	−0.013	−0.0099	9.988	−12.88	0.0128	−0.0049

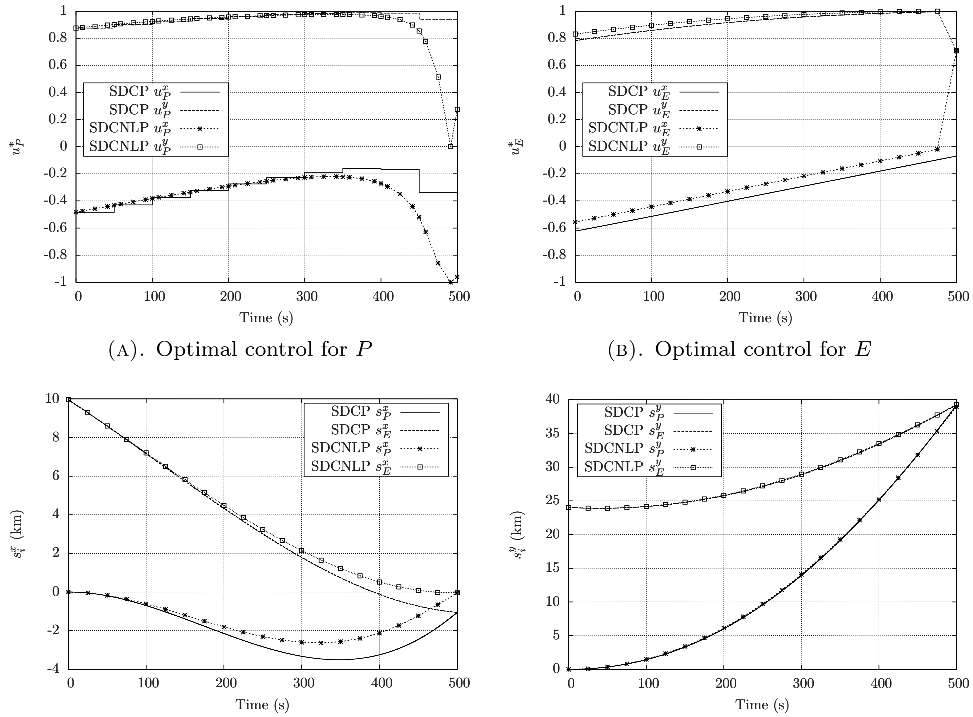
TABLE 2. Thrust-to-mass ratios for the two spacecraft

Version	T_P	T_E
1	$0.0330g$	$0.01g$
2	$0.1820g$	$0.01g$
3	$0.0550g$	$0.01g$
4	$0.0270g$	$0.01g$
5	$0.0215g$	$0.01g$
6	$0.0450g$	$0.01g$
7	$0.0860g$	$0.01g$
8	$0.0336g$	$0.01g$

values of T_P and T_E, where the constant $g = 9.8 \times 10^{-3}\text{km/s}^2$ is the base unit for the thrust-to-mass ratios.

Details of our numerical implementation are given as follows. For the SDCP method, we use the FORTRAN software package MISER 3.2 [14], which is based on the nonlinear programming code NLPQL by Schittkowski [25], to implement the control parameterization procedure with $q = 10$ subintervals. The multi-objective genetic algorithm code by Deb et al. [7] is used to determine a good starting point for MISER 3.2. For the hybrid method, we use the multiple shooting code BNDSCO [22], with the solution obtained by MISER 3.2 as the starting point. Finally, for the SDCNLP method, we use the Gauss-Lobatto quadrature collocation method and solve the resulting nonlinear programming problem using the FORTRAN code NPSOL [9].

The computer used for our numerical experiments is a server with two 2.27GHz Xeon CPUs and 24 GB memory. For both the SDCP and the SDCNLP methods, we set the constraint accuracy and the optimization tolerance to 10^{-9}. Fig. 2-5 show the optimal controls and optimal state trajectories for Problem (O_P) corresponding to versions 1 and 2 of the pursuit-evasion problem. Since the optimal controls and optimal state trajectories for the other versions are similar, they are not shown here.



(A). Optimal control for P

(B). Optimal control for E

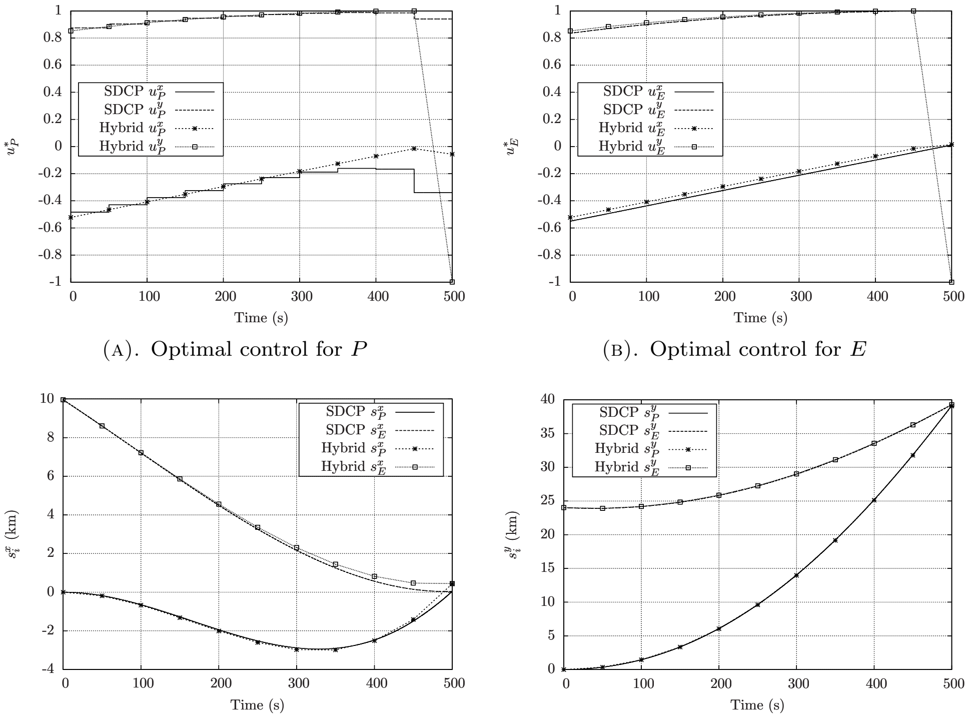
(C). Optimal trajectories for s_P^x and s_E^x

(D). Optimal trajectories for s_P^y and s_E^y

FIGURE 2. Optimal solution for version 1: optimal controls and state trajectories generated by the SDCP and SDCNLP methods.

In Table 3, we compare the accuracy of the SDCP, hybrid and SDCNLP methods. The errors for the SDCP and SDCNLP methods are measured by the violation of constraint (9); the error for the multiple shooting method is the sum of the constraint errors at each shooting point. The results in Table 3 indicate that, as expected, the hybrid method is considerably more accurate than both the SDCP and the SDCNLP methods.

In Tables 4 and 5, we compare the efficiency of the three methods in terms of CPU time. The results in Table 4 show that the SDCP method requires much less computation time than the SDCNLP method with the same level of precision. When the constraint tolerance is reduced to 10^{-16} to match the accuracy of the hybrid method, the SDCNLP method with ten subintervals (and five Gauss-Lobatto quadrature points per subinterval) does not converge. Moreover, we observed that even when the number of subintervals increases, the results from the SDCNLP method are still not as accurate as those from the hybrid method. This is due to finite precision arithmetic [6]; an accumulation of rounding errors begins to dominate solution accuracy as the number of collocation points increases [10]. Table 5 shows the corresponding CPU times for the hybrid method. The stars in the SDCNLP column indicate that when the constraint tolerance is 10^{-16}, the SDCNLP method did not converge for any of the eight versions of the pursuit-evasion problem.



(A). Optimal control for P

(B). Optimal control for E

(C). Optimal trajectories for s_P^x and s_E^x

(D). Optimal trajectories for s_P^y and s_E^y

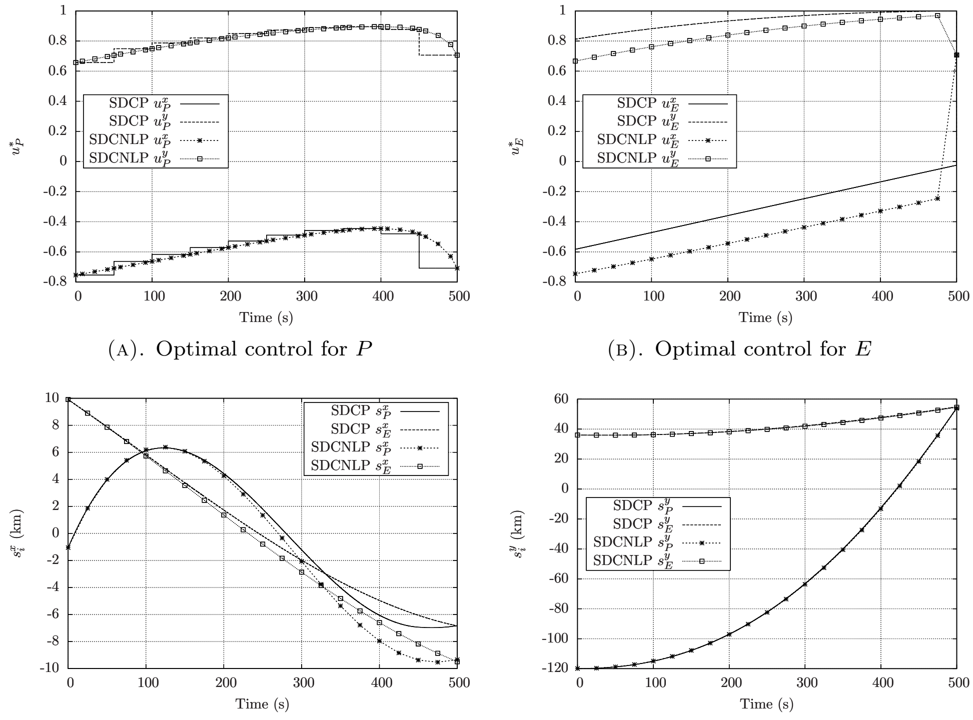
FIGURE 3. Optimal solution for version 1: optimal controls and state trajectories generated by the SDCP and hybrid methods.

TABLE 3. Constraint violations for the optimal solutions generated by the SDCP, hybrid and SDCNLP methods

Version	Constraint violation		
	SDCP	Hybrid method	SDCNLP
1	6.8025×10^{-8}	1.4142×10^{-16}	1.1032×10^{-10}
2	6.5385×10^{-9}	5.5099×10^{-15}	2.8806×10^{-9}
3	7.6607×10^{-10}	2.7568×10^{-16}	1.8813×10^{-10}
4	1.9229×10^{-10}	2.1679×10^{-17}	1.2515×10^{-9}
5	2.4050×10^{-10}	1.8708×10^{-16}	8.1093×10^{-10}
6	2.4050×10^{-10}	4.2426×10^{-15}	1.0611×10^{-10}
7	6.1088×10^{-10}	1.8708×10^{-16}	2.5512×10^{-10}
8	1.8344×10^{-10}	2.0736×10^{-15}	3.7322×10^{-10}

6. Conclusions.

This paper introduces two computational methods for generating numerical solutions to a class of spacecraft pursuit-evasion differential games. The first method, called the semi-direct control parameterization method (SDCP



(A). Optimal control for P

(B). Optimal control for E

(C). Optimal trajectories for s_P^x and s_E^x

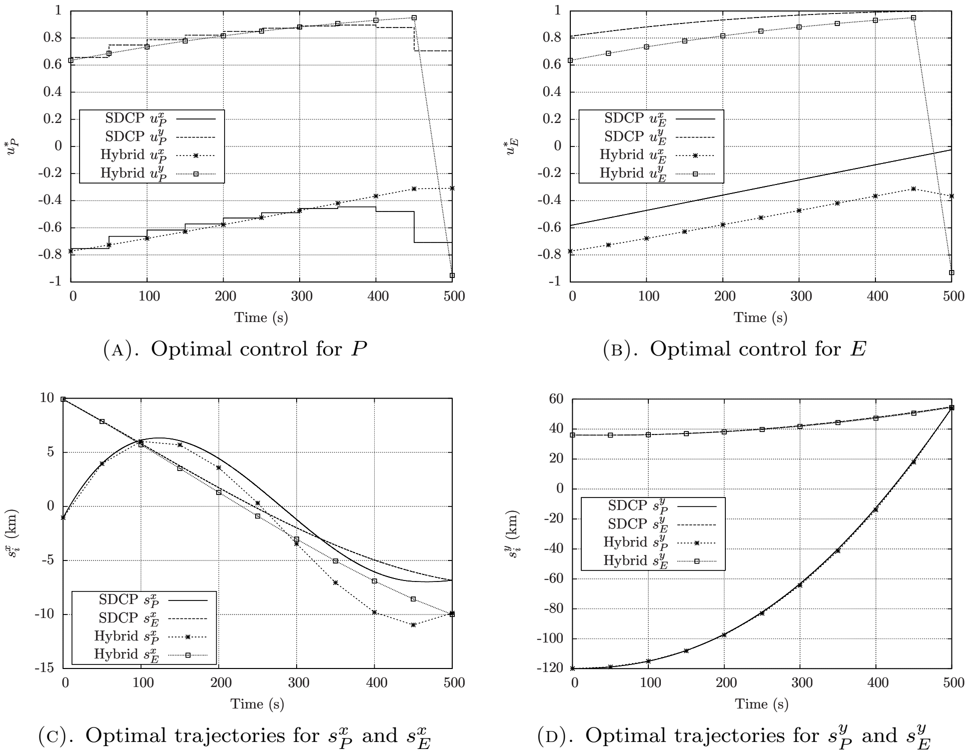
(D). Optimal trajectories for s_P^y and s_E^y

FIGURE 4. Optimal solution for version 2: optimal controls and state trajectories generated by the SDCP and SDCNLP methods.

TABLE 4. Computation times for the SDCNLP and SDCP methods

Version	Computation time (seconds)	
	SDCNLP	SDCP
1	2.424	0.358
2	1.887	0.124
3	4.161	0.328
4	2.443	0.299
5	5.327	1.627
6	2.243	0.427
7	5.319	0.459
8	1.885	0.453

method), is based on two optimal control formulations of the differential game. Our main result shows that by solving these optimal control problems, we can determine whether the differential game has a saddle point solution. The second method we presented is a hybrid method that combines the SDCP method with the



(A). Optimal control for P

(B). Optimal control for E

(C). Optimal trajectories for s_P^x and s_E^x

(D). Optimal trajectories for s_P^y and s_E^y

FIGURE 5. Optimal solution for version 2: optimal controls and state trajectories generated by the SDCP and hybrid methods.

TABLE 5. Computation times for the SDCNLP and hybrid methods

Version	Computation time (seconds)	
	SDCNLP	Hybrid method
1	*	0.491
2	*	0.247
3	*	0.448
4	*	0.412
5	*	1.893
6	*	0.710
7	*	0.695
8	*	0.665

multiple shooting method to capture the best qualities of each method—namely, the strong convergence properties of the SDCP method and the accuracy of the multiple shooting method. Our numerical results in Section 5.2 indicate that, for the class of spacecraft pursuit-evasion problems considered in this paper, our new methods outperform the existing SDCNLP method.

REFERENCES

[1] M. Bardi, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhauser, Boston, 1997.

[2] L. D. Berkovitz, Necessary conditions for optimal strategies in a class of differential games and control problems, *SIAM Journal on Control and Optimization,* **5** (1967), 1–24.

[3] L. D. Berkovitz, The existence of value and saddle point in games of fixed duration, *SIAM Journal on Control and Optimization,* **23** (1985), 172–196.

[4] M. Breitner, H. Pesch and W. Grimm, Complex differential games of pursuit-evasion type with state constraints, part 2: Necessary conditions for optimal open-loop strategies, *Journal of Optimization Theory and Applications,* **78** (1993), 443–463.

[5] W. H. Clohessy and R. S. Wiltshire, Terminal guidance system for satellite rendezvous, *Journal of the Aerospace Sciences,* **11** (1960), 653–658.

[6] S. D. Conte and C. de Boor, *Elementary Numerical Analysis: An Algorithmic Approach*, Third Edition, McGraw-Hill, New York, 1981.

[7] K. Deb, A fast and elitist multi-objective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation,* **6** (2002), 182–197.

[8] A. Friedman, *Differential Games*, American Mathematical Society, Rhode Island, 1974.

[9] P. E. Gill, W. Murray, M. Saunders and M. H. Wright, *User's Guide for NPSOL (Version 5.0): A Fortran Package for Nonlinear Programming*, Systems and Optimization Lab, Stanford University, California, 1998.

[10] A. L. Herman and B. A. Conway, Direct optimization using collocation based on high-order Gauss-Lobatto quadrature rules, *Journal of Guidance, Control, and Dynamics,* **19** (1996), 592–599.

[11] K. Horie, *Collocation with Nonlinear Programming for Two-Sided Flight Path Optimization*, Ph.D. Thesis, University of Illinois at Urbana-Champaign, Champaign, 2002.

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

[13] R. Isaacs, *Differential Games*, John Wiley and Sons, New York, 1965.

[14] L. S. Jennings, M. E. Fisher, K. L. Teo and C. J. Goh, *MISER 3 Optimal Control Software: Theory and User Manual*, Department of Mathematics, The University of Western Australia, 2002.

[15] C. Jiang, Q. Lin, C. Yu, K. L. Teo and G. R. Duan, An exact penalty method for free terminal time optimal control problem with continuous inequality constraints, *Journal of Optimization Theory and Applications,* **154** (2012), 30–53.

[16] B. Li, K. L. Teo, G. H. Zhao and G. R. Duan, An efficient computational approach to a class of minmax optimal control problems with applications, *ANZIAM Journal,* **51** (2009), 162–177.

[17] B. Li, C. Xu, K. L. Teo and J. Chu, Time optimal Zermelo's navigation problem with moving and fixed obstacles, *Applied Mathematics and Computation,* **224** (2013), 866–875.

[18] B. Li, C. J. Yu, K. L. Teo and G. R. Duan, An exact penalty function method for continuous inequality constrained optimal control problem, *Journal of Optimization Theory and Applications,* **151** (2011), 260–291.

[19] Q. Lin, R. Loxton and K. L. Teo, The control parameterization method for nonlinear optimal control: A survey, *Journal of Industrial and Management Optimization,* **10** (2014), 275–309.

[20] R. C. Loxton, Q. Lin, V. Rehbock and K. L. Teo, Control parameterization for optimal control problems with continuous inequality constraints: New convergence results, *Numerical Algebra, Control, and Optimization,* **2** (2012), 571–599.

[21] R. C. Loxton, K. L. Teo, V. Rehbock and K. F. C. Yiu, Optimal control problems with a continuous inequality constraint on the state and the control, *Automatica,* **45** (2009), 2250–2257.

[22] H. J. Oberle and W. Grimm, *BNDSCO: A Program for the Numerical Solution of Optimal Control Problems,* Inst. für Angewandte Math. der Univ. Hamburg, 2001.

[23] M. Pontani and B. A. Conway, Optimal interception of evasive missile warheads: Numerical solution of the differential game, *Journal of Guidance, Control, and Dynamics,* **31** (2008), 1111–1122.

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

[25] K. Schittkowski, NLPQL: A FORTRAN subroutine for solving constrained nonlinear programming problems, *Annals of Operations Research,* **5** (1986), 485–500.

[26] T. Shima and J. Shinar, Time-varying linear pursuit-evasion game models with bounded controls, *Journal of Optimization Theory and Applications,* **25** (2002), 607–618.

[27] J. Shinar and T. Shima, Guidance law evaluation in highly nonlinear scenarios – comparison to linear analysis, in *Proceedings of the AIAA Guidance, Navigation, and Control Conference,* (1999), 651–661.

[28] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Third Edition, Springer, New York, 2002.

[29] K. L. Teo, C. J. Goh and K. H. Wong, *A Unified Computational Approach to Optimal Control Problems*, Longman Scientific and Technical, Essex, 1991.

[30] L. Y. Wang, W. H. Gui, K. L. Teo, R. Loxton and C. H. Yang, Time delayed optimal control problems with multiple characteristic time points: Computation and industrial applications, *Journal of Industrial and Management Optimization,* **5** (2009), 705–718.

Received November 2013; 1st revision May 2014; 2nd revision July 2014.

E-mail address: song.t.sun@gmail.com
E-mail address: zhangq_h_@hit.edu.cn
E-mail address: r.loxton@curtin.edu.au
E-mail address: bin.li@curtin.edu.au