



Article

A Multi-Stage Optimization Approach for Satellite Orbit Pursuit–Evasion Games Based on a Coevolutionary Mechanism

Jian Wu ¹, Xusheng Xu ², Qiufan Yuan ², Haodong Han ³ and Daming Zhou ^{3,*}

¹ Rocket Force University of Engineering, Xi'an 710025, China

² Aerospace System Engineering Shanghai, Shanghai 201109, China

³ School of Astronautics, Northwestern Polytechnical University, Xi'an 710072, China

* Correspondence: daming.zhou@nwpu.edu.cn; Tel.: +86-182-2907-6887

Abstract: For the satellite orbit pursuit–evasion game problem, this paper proposes a multi-stage optimization-based solution aimed at improving the confrontation strategies between task satellites and target satellites in complex space environments. The approach divides the satellite pursuit–evasion game into two phases: the “approach phase” and the “sustained phase”. It dynamically optimizes the trajectories and strategies of the task and target satellites to achieve adaptive orbit control and behavior optimization. To enhance the global search capability and local convergence of the algorithm, this paper employs the Zebra Optimization Algorithm, introducing a multi-population cooperative evolution mechanism, and integrates differential game theory to improve the stability and reliability of the game strategies. Simulation results demonstrate that the proposed method effectively enhances task efficiency under multiple constraints, dynamically adjusts the strategies of both the pursuer and the evader, and provides an efficient, scalable solution applicable to satellite pursuit–evasion games in complex space environments.

Keywords: zebra optimization algorithm; cooperative evolution; satellite pursuit–evasion; game optimization; differential game theory



Academic Editor: Xiaogong Hu

Received: 26 January 2025

Revised: 10 March 2025

Accepted: 18 March 2025

Published: 17 April 2025

Citation: Wu, J.; Xu, X.; Yuan, Q.; Han, H.; Zhou, D. A Multi-Stage Optimization Approach for Satellite Orbit Pursuit–Evasion Games Based on a Coevolutionary Mechanism. *Remote Sens.* **2025**, *17*, 1441. <https://doi.org/10.3390/rs17081441>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the gradual maturing of in-orbit service and control technologies, the concept of space warfare conducted through satellites has gained a technological foundation [1]. One of the core objectives of satellite warfare is to approach high-value enemy target satellites via attacking spacecraft and implement interference or paralysis. Once enemy military spacecraft can be effectively disabled, strategic advantage can be gained in warfare. On the other hand, the party possessing high-value satellites can intercept incoming enemy satellites through orbital maneuvers, evasion strategies, or deploying defensive satellites to protect the safety of their own satellites. This scenario will become a typical pattern in future space warfare.

Under this background, the Orbital Pursuit–Evasion Game (OPEG) [2–4] has gradually become a research hotspot. The essence of this game is that both the pursuer and the evader have conflicting mission objectives and maneuver strategies. It is a dynamic and continuous confrontation process involving continuous gaming between the two parties over each other's strategies, with significant impacts on the success of missions, resource utilization efficiency, and risk control. Therefore, how to optimize the control strategies for the OPEG under complex constraints has become a key issue that urgently needs to be addressed in space warfare missions.

2. Related Work

In recent years, due to the development and demand for space game confrontation technologies, an increasing number of scholars and institutions have begun to focus on satellite pursuit–evasion games, leading to a series of research achievements. After Isaacs [5] proposed the differential game theory in 1954, Ho et al. [6] utilized the Hamilton–Jacobi–Bellman (HJB) equation to convert the orbital pursuit–evasion game problem into a two-point boundary value problem to find saddle points. However, the HJB equation is difficult to solve in some scenarios. To address this issue, Pontani et al. [7] combined genetic algorithms with nonlinear programming methods to propose a semi-direct collocation method for studying long-range interception game problems, reducing the problem’s dimensions and avoiding the direct solution of boundary value problems. Carr et al. [8] proposed an initial-guess method for costate variables combined with the proportional navigation guidance method and converted the two-point boundary value problem, derived based on differential game theory, into a one-sided problem for numerical solution. Pang et al. [9] focused on the orbital pursuit–evasion game between two spacecraft near an elliptical reference orbit and proposed a method based on the exact gradient and homotopy approach in order to reduce computational burden and improve efficiency. However, due to the large computational load during the solution process, these traditional methods still face certain computational efficiency issues when dealing with nonlinear and large disturbance environments.

As deep space missions continue to evolve, the safe migration and trajectory optimization of spacecraft swarms have gradually become research hotspots. Especially in complex space environments, how to efficiently plan trajectories and avoid collisions for spacecraft swarms has become a key issue. In recent years, many scholars have proposed fast planning methods based on surrogate models to tackle high-dimensional nonlinear optimization problems. Zhou et al. [10] presented a fast planning method based on an adaptive surrogate model for the safe migration of spacecraft swarms in Halo orbits, significantly reducing computational costs and improving optimization performance. Armellin [11] proposed a collision-avoidance method based on multi-impulse convex optimization, which transforms the nonlinear optimization problem into a convex optimization problem by introducing slack variables and linearization techniques, thereby utilizing efficient convex optimization algorithms for solution.

In recent years, methods such as reinforcement learning and optimization algorithms have also been used to solve spacecraft pursuit–evasion problems, achieving good results and becoming new research hotspots. Reinforcement learning is a method in which an agent learns through continuous interaction with its environment, aiming to obtain the optimal strategy to maximize long-term rewards. During the reinforcement learning process, the agent continuously learns and updates its strategy through frequent interactions with the environment. The agent must, based on the current state, continuously try different actions to discover which ones yield higher rewards and then update its strategy accordingly [12]. The aforementioned process is usually modeled using a Markov Decision Process, which serves as the foundation for reinforcement learning. In recent years, research on reinforcement learning in spacecraft dynamics and control has attracted the attention of many scholars [13,14]. Zhu et al. [15] applied neural networks to pursuit–evasion game problems, improving computational efficiency and reducing computation time, with results that are in good agreement with those obtained using direct methods. Yang et al. [16] introduced the Deep Deterministic Policy Gradient (DDPG) algorithm into one-on-one spacecraft game problems under incomplete information conditions, demonstrating a high success rate under the given conditions. Chu et al. [17] used deep Q-learning to effectively avoid collisions while completing cooperative rendezvous missions for spacecraft. To

avoid the curse of dimensionality in continuous spaces, Liu Bingyan et al. [18] proposed a branched deep reinforcement learning architecture with multiple parallel neural networks and shared decision-making modules by constructing a fuzzy inference model to represent continuous spaces. Xu Xusheng et al. [19] proposed a multi-agent deep reinforcement learning method for spacecraft orbital pursuit–evasion games in swarms, using Multi-Agent Deep Deterministic Policy Gradient (MADDPG) to train data and ultimately obtain strategies for each satellite. Zhao et al. [20] considered spacecraft maneuvering capabilities and mission time constraints, and proposed the Predictive–Reflective Detection Multi-Agent Deep Deterministic Policy Gradient (PRD-MADDPG) algorithm for orbital pursuit–evasion problems under impulsive maneuvering scenarios. However, reinforcement learning-based training faces issues such as poor interpretability, difficulty in theoretical proof, and low reliability. Trained decision-making models lack analytical expressions and can only verify the correctness of their decisions through simulation targeting. Additionally, there is a lack of efficient training empowerment methods, and spacecraft often need to play tens of thousands of rounds of games to learn the optimal pursuit–evasion strategy.

Optimization algorithms have also been widely applied in spacecraft pursuit–evasion game scenarios, significantly improving the efficiency and quality of solving game strategies. Prince et al. [21] studied the problem of elliptical orbit rendezvous with time as the optimization objective and applied a genetic algorithm for numerical solution. Yu Dateng [22] established a multi-pulse optimal rendezvous model for chasers based on the Sequential Quadratic Programming (SQP) algorithm, optimized maneuvers using a genetic algorithm, and enhanced the spacecraft’s spatial survival capability. Qi Yinghong [23] introduced the Genetic Algorithm (GA) into the guidance problem of small-satellite pursuit and interception. Wu et al. [24] addressed the design of long-range continuous-thrust interception trajectories under J2 perturbations by establishing a two-point boundary value problem (TPBVP) based on boundary constraints and performance index functions incorporating time and fuel consumption, and then obtaining the optimal control strategy using a hybrid optimization method combining a genetic algorithm and Sequential Quadratic Programming (SQP). Stupik et al. [25] achieved good results in addressing low-thrust pursuit–evasion problems by combining a Particle Swarm Optimization (PSO) algorithm with Kriging interpolation. Sun et al. [26] transformed the game problem into an optimal control problem and solved the two-spacecraft pursuit–evasion problem using the Sequential Dual Convex Programming (SDCP) method, achieving good optimization results in combination with the multiple-shooting method and hybrid optimization strategies. Liu et al. [27] proposed a strategy-solving method combining game theory and optimization for multi-satellite pursuit and evasion scenarios, designed a cost function considering intended purposes, fuel consumption, and maneuver safety, and optimized both sides’ strategies using a Particle Swarm Optimization (PSO) algorithm to obtain optimal strategies for both parties. Wu Qichang [28] proposed the use of the Ant Colony Optimization (ACO) algorithm to optimize spacecraft pursuit–evasion game problems, further enriching the application scope of optimization algorithms in this field. Chen et al. [29] explored a cooperative decoy defense guidance strategy that considers field of view (FOV) and overload constraints in a pursuit–evasion adversarial scenario under incomplete information conditions by introducing a relatively advanced decoy. They combined a parameter delay design method with adaptive backstepping control technology to cope with the incomplete information in pursuit–evasion confrontations. This reduces the complexity of guidance system design and improves the effectiveness and simplicity of the algorithm.

Although there has been extensive research on space game confrontation, current studies primarily focus on pursuit–evasion problems involving continuous-thrust models. Relatively little research has been conducted on pursuit–evasion games based on impulse-

thrust models. In particular, effective path planning and strategy adjustment under complex constraints remain urgent issues to be addressed.

In this paper, the impulse-thrust model is transformed into a constrained control, and a spacecraft relative motion model is established. Based on the coevolutionary mechanism [30–33], Zebra Optimization Algorithm [34], and differential game theory [35,36], an innovative approach is proposed to solve satellite missions in complex space environments using a hybrid coevolutionary algorithm. The mission execution process is split into two phases: the approach phase and the sustained phase. Through multi-population coevolution and dynamic strategy adjustment mechanisms, the stability and global optimization capabilities of strategies for all parties in the game are enhanced. This provides a more efficient and adaptable solution for the satellite pursuit–evasion problem.

Compared with Lyapunov-based control methods, reinforcement learning methods, and traditional optimization methods, the method proposed in this paper can better handle complex constraints and nonlinear problems, exhibits stronger robustness, finds optimal solutions in fewer iterations with good interpretability, and better balances global search and local convergence, thereby improving optimization efficiency.

3. Problem Description

High-value primary satellites and their supporting mission satellites are distributed in geosynchronous Earth orbit. A satellite with an unknown purpose (target satellite) has appeared near the primary satellite, as illustrated in Figure 1.

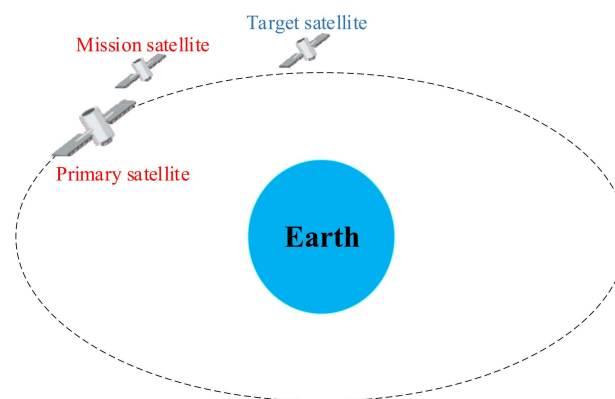


Figure 1. A diagram illustrating the problem.

To ensure the safety of the primary satellite, the mission satellite needs to maneuver its orbit to approach the conical safe approach corridor (capture zone) of the target satellite. The target satellite, however, aims to evade the mission satellite through orbital maneuvers, preventing it from entering this zone within the specified time and, if conditions permit, approaching the primary satellite. Based on the literature [37], when satellite missions can be completed within a few orbital cycles, the effects of atmospheric drag perturbation and J2 perturbation on the spacecraft can be neglected. Therefore, the influences of differences in area-to-mass ratio and atmospheric drag perturbation were not considered in this paper.

As shown in Figure 2, the capture zone is the conical area in the diagram, which is related to the position of the target satellite and always maintains a directional alignment towards the Earth throughout the entire game process. The capture angle (denoted as θ) is the angle between the vector from the target satellite to the Earth's center and the capture zone, used to describe the conditions under which the target satellite enters the capture zone.

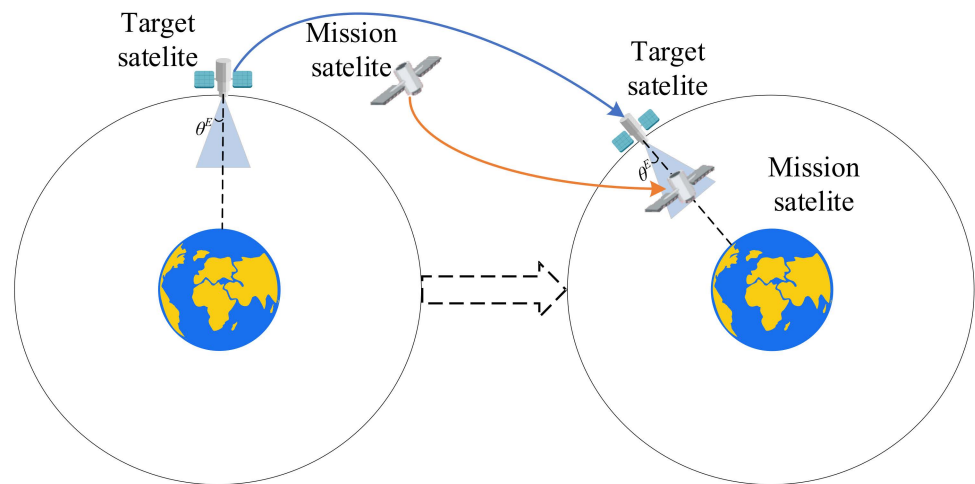


Figure 2. A schematic of the relative motion in the game on-orbit.

In the orbital coordinate system $O_b - XYZ$ with the primary satellite as the origin, the instantaneous acceleration u generated by the spacecraft's unit mass control force is shown in Figure 3 within the coordinate system O_b .

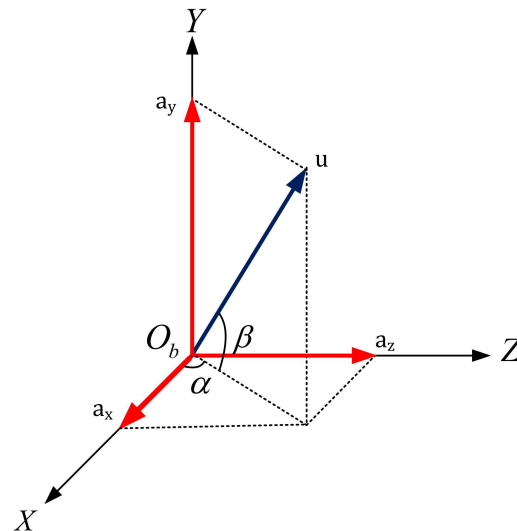


Figure 3. Spacecraft control forces and their decomposition in the orbital coordinate system.

Define β as the angle between u and the XO_bY plane, and the angle between its projection onto the $O_b - XYZ$ plane and the X -axis is α . The components of u along the three axes of $O_b - XYZ$ can be expressed as follows:

$$\begin{cases} a_x = u \cos \beta \cos \alpha \\ a_y = u \cos \beta \sin \alpha \\ a_z = u \sin \beta \end{cases} \quad (1)$$

where $\alpha \in [-\pi, \pi]$, $\beta \in [-\pi/2, \pi/2]$.

The CW equation is employed to describe the motion relationships between the mission satellite and the primary satellite, as well as between the target satellite and the primary satellite.

$$\begin{cases} \ddot{x} - 2n\dot{y} - 3n^2x = a_x \\ \ddot{y} + 2n\dot{x} = a_y \\ \ddot{z} + n^2z = a_z \end{cases} \quad (2)$$

In this context, n represents the angular velocity of the reference orbit, and its expression is given by $n = \sqrt{\mu/(a^3)}$.

Define the state of the spacecraft at time t as $x(t) = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]$, and the instantaneous acceleration applied to the spacecraft at time t_0 as $u(t) = [a_x \ a_y \ a_z]$. Let $\tau = t - t_0$, then the state transition equation of the spacecraft from t_0 to t is as follows:

$$x(t) = \Phi(\tau) \times (t_0) + \Psi(\tau)u(t) \quad (3)$$

where

$$\Phi(\tau) = \begin{bmatrix} 4 - 3\cos(n\tau) & 0 & 0 & \frac{\sin(n\tau)}{n} & \frac{2-2\cos(n\tau)}{n} & 0 \\ 6(\sin(n\tau) - n\tau) & 1 & 0 & \frac{2\cos(n\tau)-2}{n} & \frac{4\sin(n\tau)-3n\tau}{n} & 0 \\ 0 & 0 & \cos(n\tau) & 0 & 0 & \frac{\sin(n\tau)}{n} \\ 3n\sin(n\tau) & 0 & 0 & \cos(n\tau) & 2\sin(n\tau) & 0 \\ 6n\cos(n\tau) - 6n & 0 & 0 & -2\sin(n\tau) & 4\cos(n\tau) - 3 & 0 \\ 0 & 0 & -n\sin(n\tau) & 0 & 0 & \cos(n\tau) \end{bmatrix} \quad (4)$$

$$\Psi(\tau) = \begin{bmatrix} \frac{1-\cos(n\tau)}{n^2} & \frac{2n\tau-2\sin(n\tau)}{n^2} & 0 \\ \frac{2\sin(n\tau)-2n\tau}{n^2} & \frac{4-4\cos(n\tau)}{n^2} - \frac{3\tau^2}{2} & 0 \\ 0 & 0 & \frac{1-\cos(n\tau)}{n^2} \\ \frac{\sin(n\tau)}{n} & \frac{2-2\cos(n\tau)}{n} & 0 \\ \frac{2\cos(n\tau)-2}{n} & \frac{4\sin(n\tau)-3n\tau}{n} & 0 \\ 0 & 0 & \frac{\sin(n\tau)}{n} \end{bmatrix} \quad (5)$$

From the above equation, when the initial state $x(t_0)$ of the spacecraft and the control acceleration difference u at time t are known, the state $x(t)$ of the spacecraft at any time t in the orbital coordinate system can be obtained.

Assuming a fixed decision-making time T , we divide T into N discretized subintervals with a certain step size, where the step size of each interval is T/N . Thus, the entire decision-making time T is simplified into $N + 1$ discrete time points. The state transition equation of the spacecraft from time kT/N to $(k + 1)T/N$ ($k = 0, 1, \dots, N$) is as follows:

$$x\left(\frac{(k+1)T}{N}\right) = \Phi\left(\frac{T}{N}\right) \times \left(\frac{kT}{N}\right) + \Psi\left(\frac{T}{N}\right)u\left(\frac{kT}{N}\right) \quad (6)$$

If the velocity impulse applied to the spacecraft at kT/N is 0, then the state transition equation can be further simplified as

$$x\left(\frac{(k+1)T}{N}\right) = \Phi\left(\frac{T}{N}\right) \times \left(\frac{kT}{N}\right) \quad (7)$$

4. Constraint Conditions

In the space environment, satellites need to consider not only the impact of the environment but also the constraints of mission completion conditions and their own capabilities, such as mission time constraints, impulse magnitude constraints, and constraints on the interval between two impulses.

4.1. Mission Condition Constraints

The mission satellite needs to approach the target satellite's capture zone and remain within it for a certain duration to suppress the target satellite for an extended period.

(1) Relative distance constraint

The mission satellite must maintain a relative distance to the target satellite that is less than a certain minimum distance d_M . This distance limit is usually to ensure that the mission satellite can successfully enter a specific area and avoid being too far away to perform effective mission operations. This constraint can be expressed as

$$d \leq d_M \quad (8)$$

where d_M is the minimum relative distance required by the tracking mission.

(2) Capture angle constraint

When the relative distance between the mission satellite and the target satellite meets the requirement, the mission satellite also needs to enter a specific capture zone. At this point, the capture angle θ is used to describe the condition for the target satellite to enter the capture zone. This constraint can be expressed as

$$\theta \leq \theta_M \quad (9)$$

where θ_M represents the minimum angle required for the mission satellite to enter the capture zone of the target satellite.

(3) Duration constraint

Under the constraints of the above relative distance and capture angle, to ensure prolonged suppression of the target satellite, the duration of the continuous segment is set to T . This constraint can be expressed as

$$T_M \leq T \quad (10)$$

where T_M is the minimum duration required to complete the mission.

4.2. Velocity Impulse Control Constraint

(1) Impulse magnitude constraint

The velocity impulse of the mission satellite has maximum and minimum magnitude limits. The impulse magnitude determines the amplitude of the mission satellite's orbital change, which is the change in velocity. The constraint for setting the velocity impulse magnitude is as below:

$$u_{\min} \leq u \leq u_{\max} \quad (11)$$

where $u_{\min}$ and $u_{\max}$ represent the minimum and maximum values of the velocity impulse, respectively.

(2) Pulse interval constraint

The pulse interval refers to the time interval between two pulse maneuvers. The pulse interval controls the frequency and rhythm of satellite orbit adjustments. The constraint on the pulse interval can be expressed by the following formula:

$$\Delta t_{\min} \leq \Delta t \quad (12)$$

where $\Delta t_{\min}$ represents the minimum value of the pulse interval.

(3) Energy consumption constraint

Satellites have limited energy resources, and each pulse adjustment consumes the satellite's fuel. The energy consumption constraint ensures that the mission satellite can allocate energy reasonably throughout the mission process, avoiding mission interruption

due to fuel shortage. Ignoring changes in satellite weight, it can be converted into the total velocity impulse. This constraint can be expressed as

$$\sum \Delta v \leq \Delta v_{\max} \quad (13)$$

where $\sum \Delta v$ represents the total velocity impulse, and $\Delta v_{\max}$ is the upper limit of the cumulative velocity impulse for the satellite.

4.3. Time Constraint

The total duration of the mission must be completed within the given mission time and cannot exceed the time limit. This constraint can be expressed as

$$t_f \leq T_f \quad (14)$$

where t_f is the total duration of the mission, and T_f is the maximum allowed mission duration.

It should be noted that the mission satellite actively carries out tasks, while the target satellite passively counteracts. Constraints related to the mission, such as relative distance and duration, are independent of the target satellite. For example, the constraints for the target satellite include a pulse magnitude constraint and pulse interval constraint, which are completely independent of the mission satellite.

5. Design and Implementation of Hybrid Cooperative Evolutionary Algorithm

To solve the satellite pursuit–evasion game in complex space environments, this paper proposes a hybrid cooperative evolutionary algorithm by adopting the Zebra Optimization Algorithm (ZOA), combined with the cooperative evolutionary mechanism and differential game theory. The core operation of this algorithm simulates the behavioral characteristics of zebras, such as group migration, information sharing, and cooperation, to guide the mutual evolution between the mission satellite population and the target satellite population, ultimately seeking the optimal strategies for both sides.

5.1. Design of Fitness Function

The space mission can be described as a task satellite first approaching the target satellite's capture zone and then remaining within the capture zone for a certain period of time, forming an effective suppression of the target satellite. Therefore, the entire mission can be divided into two phases: the approach phase and the sustainment phase.

In the approach phase, the mission satellite needs to approach the capture region of the target satellite and ensure that the constraints on relative distance and capture angle are satisfied during the approach. The optimization strategy of the mission satellite is to minimize the mission-related loss function by optimizing control variables within the specified maximum mission time. Precise impulse maneuvers are used to adjust the orbit, ensuring strict compliance with mission requirements. Meanwhile, the target satellite adopts an orbit optimization strategy based on evasion, constantly adjusting its trajectory to move away from the mission satellite while getting as close as possible to the primary satellite, thereby enhancing its evasion capability and maximally interfering with the mission execution of the mission satellite.

In the sustained phase, the mission satellite frequently applies impulse control to compensate for the trajectory deviations caused by the target satellite's evasion maneuvers, ensuring that the target satellite remains within the capture region and prevents its escape. The mission satellite maintains the relative distance and capture angle through multiple impulse maneuvers within the capture region, ensuring that the target satellite cannot

escape. The target satellite continues to maneuver in an attempt to break free from the suppression of the mission satellite and shorten the mission satellite's dwell time within the capture region as much as possible.

The total mission duration t_f is divided into multiple discretized subintervals with a certain velocity pulse interval Δt , simplifying the entire decision-making time t_f into $\lfloor t_f / \Delta t \rfloor + 1$ discrete time points.

For the approach phase, the objective function is the result of the comparison of the relative distance between the task satellite and the target satellite at the termination time, the capture angle, and their respective thresholds. The objective function for the approach phase is designed as below:

$$S_{Appr} = \min_{0 \leq n \leq \lfloor \frac{t_f}{\Delta t} \rfloor + 1} (\max(\|r_1^n - r_2^n\| - d, 0) + \max(\varphi(r_1^n, r_2^n) - \theta, 0)) \quad (15)$$

In this context, r_1^n and r_2^n represent the position vectors of the task satellite and the target satellite at time n , respectively; $\varphi(\cdot)$ denotes the capture angle calculation function; and $\lfloor \cdot \rfloor$ represents the floor function (downward rounding).

For the sustainment phase, the optimization objective function of the task satellite is basically the same as that for the approach phase, with the difference being that in the sustainment phase, the threshold condition needs to be satisfied at every time point. Therefore, the outer minimization is changed to maximization. Specifically, it is as follows:

$$S_{Dur} = \max_{0 \leq n \leq \lfloor \frac{t_f}{\Delta t} \rfloor + 1} (\max(\|r_1^n - r_2^n\| - d_1^M, 0) + \max(\varphi(r_1^n, r_2^n) - \theta, 0)) \quad (16)$$

If the objective function equals 0, the mission is successful and can be terminated; otherwise, continue iterating occurs until the maximum number of iterations is reached, at which point the mission ends and is considered failed.

5.2. Differential Game Model

The actual space mission environment is more complex, and the target satellite does not always passively accept the "offensive" strategy of the task satellite. If the target satellite possesses maneuverability, it can actively adjust its motion state and optimize control parameters to evade the approach or interference of the task satellite, achieving a certain degree of "defense". This two-way dynamic adjustment complicates the interaction between the task satellite and the target satellite, transforming the problem from unilateral strategy optimization to an adversarial issue where both sides determine trajectories and final states through strategic games. Therefore, this paper constructs a differential game model for scenarios where the target satellite possesses maneuverability and proposes a hybrid coevolutionary algorithm to solve this game problem.

The objective function minimized by the task satellite is denoted as S_L , where $L \in \{Appr, Dur\}$ represents the mission phase, namely the approach phase and the sustainment phase. The optimization objectives of the task satellite and the target satellite are different, necessitating separate modeling for each.

During the approach phase, the strategy of the task satellite is to minimize the mission-related loss function by optimizing control variables within the specified maximum mission time T_f . The control parameters of the task satellite include the total mission time t_f , maneuver times $t^A = [t_1^A, t_2^A, \dots, t_{k^A}^A]^T$, impulse magnitudes $u^A = [u_1^A, u_2^A, \dots, u_{k^A}^A]^T$, angles $\beta^A = [\beta_1^A, \beta_2^A, \dots, \beta_{k^A}^A]^T$ between the impulses and the XO_bY plane, and angles $\alpha^A = [\alpha_1^A, \alpha_2^A, \dots, \alpha_{k^A}^A]^T$ between the projections within the XO_bY plane and the X -axis,

where $k^A \leq k_A^{MAX}$ and k_A^{MAX} represents the maximum number of impulse maneuvers for the task satellite. The optimization model for the task satellite can be expressed in the following form:

$$\min_{u^A, t_f^A, \alpha^A, \beta^A} S^{Appr} (u^A, t_f^A, \alpha^A, \beta^A; u^O, t^O, \alpha^O, \beta^O) \quad (17)$$

$$s.t. \begin{cases} t_f^A \leq T_f \\ 1 \leq k^A \leq k_A^{MAX} \\ u_i^A \leq u_A^{MAX} \\ t_{i+1}^A - t_i^A \geq \Delta t_{min}^A, 1 \leq i \leq k^A \\ \pi \leq \alpha_i^A \leq -\pi \\ -\frac{\pi}{2} \leq \beta_i^A \leq \frac{\pi}{2} \end{cases}$$

In this context, $u^O, t^O, \alpha^O, \beta^O$ represent the control strategies of the target satellite, which influence the optimal strategy of the task satellite. t_f^A is the total mission time for the task satellite to execute the approach phase, u_A^{MAX} is the maximum value for a single impulse of the task satellite, and Δt_{min}^A is the minimum time interval between two impulse maneuvers of the task satellite. The constraint conditions ensure that the control variables of the task satellite are within the physically and mission-permitted ranges, guaranteeing the practical feasibility of the optimization problem.

For the sustainment phase, the task satellite does not need to optimize the mission time, and the optimization model is as follows:

$$\min_{u^A, t^A, \alpha^A, \beta^A} S^{Dur} (u^A, t^A, \alpha^A, \beta^A; u^O, t^O, \alpha^O, \beta^O) \quad (18)$$

$$s.t. \begin{cases} 1 \leq k^A \leq k_A^{MAX} \\ u_i^A \leq u_A^{MAX} \\ t_{i+1}^A - t_i^A \geq \Delta t_{min}^A, 1 \leq i \leq k^A \\ \pi \leq \alpha_i^A \leq -\pi \\ -\frac{\pi}{2} \leq \beta_i^A \leq \frac{\pi}{2} \end{cases}$$

The loss function of the target satellite is unrelated to the mission phase, and its optimization objective is not necessarily solely to evade the task satellite. In practice, different objectives can be set for the target satellite based on various requirements to make it more intelligent. This paper designs two objectives for the target satellite: the first is to evade the task satellite's actions to enter the capture area, i.e., to maximize the distance from the task satellite during the game process; the second is to approach the primary satellite as closely as possible and perform certain actions on it. The control variables of the target satellite include maneuver times $t^O = [t_1^O, t_2^O, \dots, t_{k^O}^O]^T$, impulse magnitudes $u^O = [u_1^O, u_2^O, \dots, u_{k^O}^O]^T$, angles $\beta^O = [\beta_1^O, \beta_2^O, \dots, \beta_{k^O}^O]^T$ between the impulses and the XO_bY plane, and angles $\alpha^O = [\alpha_1^O, \alpha_2^O, \dots, \alpha_{k^O}^O]^T$ between the projections within the XO_bY plane and the X -axis, where $k^O \leq k_O^{MAX}$ and k_O^{MAX} represents the maximum number of impulse maneuvers for the target satellite. Similar to the calculation method of the loss function for the task satellite, for a game of duration tr , the loss function of the target satellite can be expressed as follows:

$$R = \max_{0 \leq n \leq \lfloor \frac{tr}{\Delta t} \rfloor + 1} \left(\max(d_R^M - \|r_1^n - r_2^n\|, 0) + \|r_2^n\| \right) \quad (19)$$

In the approach phase, the game duration t_r is equal to the mission duration of the task satellite t_f^A ; in the sustainment phase, the game duration t_r is the continuous duration T of the task satellite within the capture area. The strategy of the target satellite is to

optimize parameters such as impulse times, impulse magnitudes, and angles within the confrontation time with the task satellite to minimize its loss function. The optimization model for the target satellite is as follows:

$$\min_{u^O, t^O, \alpha^O, \beta^O} R(u^A, t^A, \alpha^A, \beta^A; u^O, t^O, \alpha^O, \beta^O) s.t. \quad (20)$$

$$\begin{cases} 1 \leq k^O \leq k_O^{MAX} \\ u_i^O \leq u_O^{MAX} \\ t_{i+1}^O - t_i^O \geq \Delta t_{min}^O, \quad 1 \leq i \leq k^O \\ -\pi \leq \alpha_i^A \leq -\pi \\ \frac{\pi}{2} \leq \beta_i^A \leq \frac{\pi}{2} \end{cases}$$

In this context, $u^A, t^A, \alpha^A, \beta^A$ represent the control strategies of the task satellite, u_O^{MAX} represents the upper limit of the single impulse magnitude for the target satellite, and Δt_{min}^O represents the minimum time interval between two impulse maneuvers of the target satellite.

The pursuit–evasion game problem between a task satellite and a target satellite requires finding a saddle point solution that satisfies the following equation under given constraints:

$$S^L(u^{A*}, t^{A*}, \alpha^{A*}, \beta^{A*}; u^O, t^O, \alpha^O, \beta^O) \leq S^L(u^A, t^A, \alpha^A, \beta^A; u^O, t^O, \alpha^O, \beta^O) \quad (21)$$

$$R(u^A, t^A, \alpha^A, \beta^A; u^{O*}, t^{O*}, \alpha^{O*}, \beta^{O*}) \leq R(u^A, t^A, \alpha^A, \beta^A; u^O, t^O, \alpha^O, \beta^O) \quad (22)$$

In this context, $u^{A*}, t^{A*}, \alpha^{A*}, \beta^{A*}$ represent the optimal control strategies for the task satellite; $u^{O*}, t^{O*}, \alpha^{O*}, \beta^{O*}$ represent the optimal control strategies for the target satellite.

5.3. Design of Strategy Update and Co-Evolution Mechanisms

In the Zebra optimization algorithm, individuals optimize their behavior through mechanisms of collaboration and competition. Within the framework of co-evolution, the relationship between the task satellite and the target satellite is not one of cooperation; instead, they optimize their orbital control strategies through mutual competition and dynamic adjustment. In the approach and sustained phases of the mission, the task satellite tracks the target satellite based on its escape strategy, while the target satellite evades the pursuit of the task satellite based on its tracking strategy. Through this non-cooperative competitive process, both satellites continuously optimize their behavior in interaction to maximize their respective mission objectives.

(1) Population initialization

Task satellite population: It consists of NA individuals. Each individual's strategy is composed of five control variables: total mission duration t_f , impulse maneuver times $t_i^{(A)}$, impulse magnitudes $u_i^{(A)}$, and two orientation angles $\alpha_i^{(A)}$ and $\beta_i^{(A)}$.

$$\mathbb{S}^{(A)} = \{ \mathbf{S}_i^{(A)} \mid i = 1, 2, \dots, N_A \} \quad (23)$$

$$\mathbf{S}_i^{(A)} = [t_f, t_i^{(A)}, u_i^{(A)}, \alpha_i^{(A)}, \beta_i^{(A)}] \quad (24)$$

Among them, the total mission duration t_f is an optimization variable during the approach phase, not exceeding the maximum time limit of the mission, i.e., $t_f \leq T_f$; during the sustained phase, this variable does not need to be optimized, and the total mission duration is the minimum sustained duration T_M required to fulfill the mission. The impulse magnitude needs to consider the maneuvering capability of the task satellite and

is optimized within the range between the minimum and maximum values of the velocity impulse, i.e., $u^{(A)} \in [u_{\min}^{(A)}, u_{\max}^{(A)}]$.

Target satellite population: It consists of N_O individuals. Since the target satellite and the task satellite are in the same game, the total mission duration is the same, and therefore there is no need to optimize the total mission duration for the target satellite. Instead, the total mission time of the task satellite is taken as the game time. The control variables for each individual are the impulse maneuver times $t_i^{(O)}$, impulse magnitudes $u_i^{(O)}$, and two orientation angles $\alpha_i^{(O)}$ and $\beta_i^{(O)}$.

$$\mathbb{S}^{(O)} = \{S_i^{(O)} \mid i = 1, 2, \dots, N_O\} \quad (25)$$

$$S_i^{(O)} = [t_i^{(O)}, u_i^{(O)}, \alpha_i^{(O)}, \beta_i^{(O)}] \quad (26)$$

In this context, $u_i^{(O)} \in [u_{\min}^{(O)}, u_{\max}^{(O)}]$ indicates that the impulse magnitude of the target satellite also needs to be optimized within the range between its minimum and maximum velocity impulse values.

(2) Fitness evaluation

Equations (15) and (16) represent the fitness functions for the task satellite during the approach phase and the sustained phase, respectively, while Equation (19) is the fitness function for the target satellite. The strategy of each individual is evaluated to calculate its performance in the current environment for both the task satellite population and the target satellite population.

(3) Zebra behavior simulation

The zebra population evolves through cooperation and competition, with each individual adjusting based on the state of other members in the population. Task satellites and target satellites continuously exchange information during the optimization process, thereby co-evolving to reach optimal solutions for both parties.

① Foraging phase

After selecting a pioneer zebra within the population, the pioneer zebra guides other zebras to its position within the population. The position update formula is as follows:

$$S_{i,j}^{\text{new}} = S_{i,j} + r * (PZ_j - I * S_{i,j}) \quad (27)$$

$$S_i = \begin{cases} S_i^{\text{new}}, & F_i^{\text{new}} < F_i \\ S_i, & \text{else} \end{cases} \quad (28)$$

In this context, j represents the number of zebra populations, and i represents the number of individual zebras. $S_{i,j}$ denotes the position of a zebra, and $S_{i,j}^{\text{new}}$ denotes the updated position of the zebra. PZ_j is the pioneer zebra of each population, S_i is the position of the pioneer zebra after updating, and F_i is the objective function value of the i -th individual. r is a random number between $[0, 1]$, and $I \in \{1, 2\}$. A higher value of I represents a greater change in the population.

② Phase of resisting predator attacks

Wild zebras on the savanna may encounter two types of predators, with the assumption that the probability of both situations occurring is the same.

Phase 1: When lions attack zebras, the zebra population chooses an escape strategy.

Phase 2: When faced with other predators (such as hyenas, gray wolves, and other smaller predators), the zebra population chooses a strategy of gathering or attacking.

The position update formula is as follows:

$$S_{i,j}^{\text{new}} = \begin{cases} \text{Phase1 : } S_{i,j} + R > (2r - 1)(1 - \frac{n}{M})S_{i,j}, & P \leq 0.5 \\ \text{Phase2 : } S_{i,j} + r(AZ_j - I * S_{i,j}), & \text{else} \end{cases} \quad (29)$$

$$S_i = \begin{cases} S_i^{\text{new}}, & F_i^{\text{new}} < F_i \\ S_i, & \text{else} \end{cases} \quad (30)$$

In this context, P represents the probability of occurrence for the two strategies, ranging from $[0, 1]$. When $P \leq 0.5$, the first scenario is considered to occur; otherwise, the second scenario is considered. AZ_i denotes the position of the attacked zebra. M is the maximum number of iterations, and n is the current number of iterations. R is a constant value of 0.01, in order to maintain a small step size during the position update process of the zebra swarm, avoiding premature convergence or getting trapped in local optima due to excessive position updates. A smaller R value allows the zebra swarm to adjust positions more finely in the search space, thereby enhancing the algorithm global search capability and local convergence accuracy.

(4) Solution Process

The process of integrating the cooperative coevolution algorithm is shown in Figure 4, and the detailed steps are described as follows:

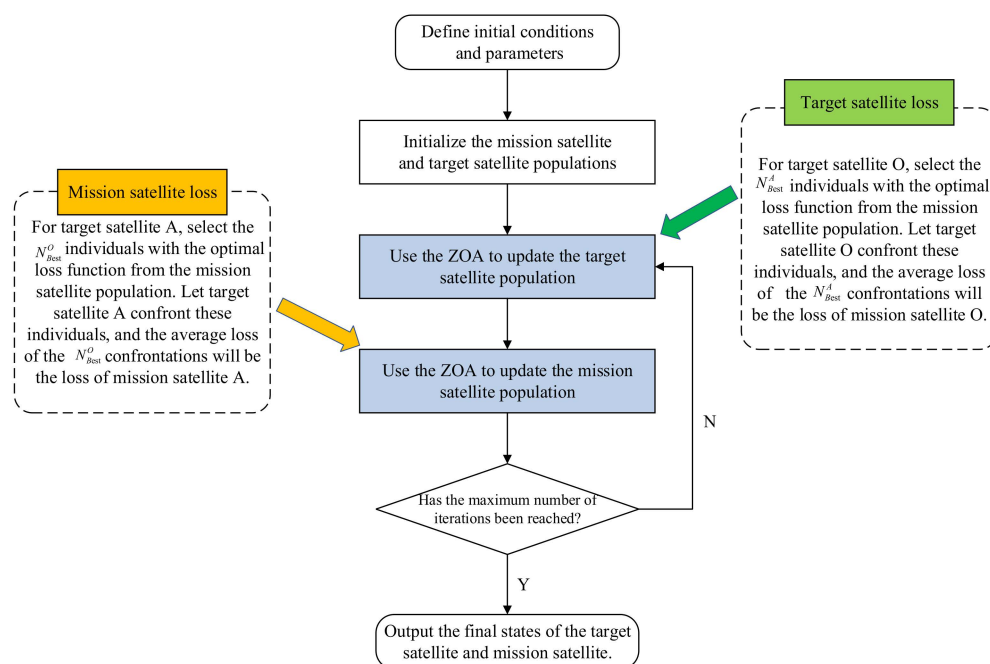


Figure 4. Fusion cooperative evolutionary algorithm process.

Step 1: Define the initial conditions and relevant parameters for the optimization process. Determine the initial states of the task satellite and target satellite, the optimization range of control variables, and the initial parameters of the Zebra Optimization Algorithm (ZOA).

Step 2: Initialize the populations of task satellites and target satellites. For the control variables of the target satellite and task satellite populations, uniformly randomly select an initial position within their optimization ranges as the initial zebra individual.

Step 3: Update the target satellite population using the ZOA. Based on the loss function of the target satellite, update the target satellite population.

Step 4: Update the task satellite population using the ZOA. Based on the loss function of the task satellite, update the task satellite population.

Step 5: Determine if the maximum number of iterations has been reached. Check if the current number of iterations has reached the preset maximum number of iterations. If not, return to Step 3 to continue iterating.

Step 6: Select the optimal individuals from both populations. The final selected optimal individuals represent the best-performing strategies in this adversarial process, ensuring that both the task satellite and target satellite achieve the minimization of their respective losses.

6. Simulation Results and Analysis

6.1. Scenario 1: The Agility of the Mission Satellite Is Twice That of the Target Satellite

The primary satellite is located in an orbit 36,000 km above the Earth's surface. The orbital parameters are as follows: semi-major axis of 42,378 km, eccentricity of 0, orbital inclination of 40 degrees, right ascension of ascending node of 20 degrees, argument of perigee of 0 degrees, and true anomaly of 87 degrees. In the orbital coordinate system, the relative distances and relative velocities between the mission satellite, the target satellite, and the primary satellite are shown in Table 1.

Table 1. Initial states of the mission satellite and target satellite.

States	$x/(m)$	$y/(m)$	$z/(m)$	$\dot{x}/(m)$	$\dot{y}/(m)$	$\dot{z}/(m)$
Mission Satellite	−8.660254	−10.0	15.0	−0.000361	−0.001253	−0.000626
Target Satellite	0	150.0	0	0	0	0

The mission satellite approaches the capture area of the target satellite and establishes a long-term suppression. During each phase, the maneuvering times of the mission satellite do not exceed 20, with each impulse not exceeding 2 m/s. The time interval between two impulse maneuvers is no less than 200 s. In the game between the mission satellite and the target satellite, the mission satellite optimizes its motion control strategy to effectively suppress the target satellite. Meanwhile, the target satellite possesses certain maneuvering capabilities, with its maneuvering times in each phase also not exceeding 20, and each impulse not exceeding 1 m/s. The time interval between two impulse maneuvers is no less than 400 s. During the game with the mission satellite, the target satellite attempts to move away from the mission satellite and get closer to the primary satellite. The maximum confrontation time between the two satellites is 100,000 s, and the duration of each single mission is no less than 1000 s. The simulation step size is 20 s.

The mission satellite orbits around the primary satellite at a distance of 20 km to ensure its safety. The target satellite is located 150 km ahead of the primary satellite on the same orbit. The mission satellite needs to transfer into the capture area of the target satellite and maintain a relative distance of less than 20 km with a capture angle of less than 10 degrees for 1000 s, as shown in Table 2. To achieve this goal, the mission satellite will adjust its orbit through precise impulse maneuvers within the limited mission time to ensure strict compliance with mission requirements. Meanwhile, the target satellite will adopt orbit optimization based on evasion strategies, constantly adjusting its trajectory to move away from the mission satellite while getting as close to the primary satellite as possible, thereby enhancing its evasion capabilities and maximizing interference with the mission satellite's task execution.

Table 2. Parameter configuration for the detection scenario.

Distance (km)	Illumination Angle (°)	Duration (s)	Maximum Mission Time (s)
$d_M = 20$	$\theta = 10$	1000	100,000

Tables 3 and 4, respectively, summarize the impulse times, impulse magnitudes, and angular information for the mission satellite and target satellite during each phase. From the start to 39,320 s, the mission satellite was in the approach phase, completing a total of seven impulse adjustments to meet the mission requirements of the approach segment. During the subsequent 1000 s sustained phase, the mission satellite performed five impulse maneuvers to ensure the stability and accuracy of the relative distance and capture angle. Meanwhile, the target satellite executed eight impulse maneuvers in the approach phase and two in the sustained phase, engaging in confrontation with the mission satellite to hinder its mission completion. Ultimately, the mission satellite successfully accomplished all mission objectives at 40,320 s.

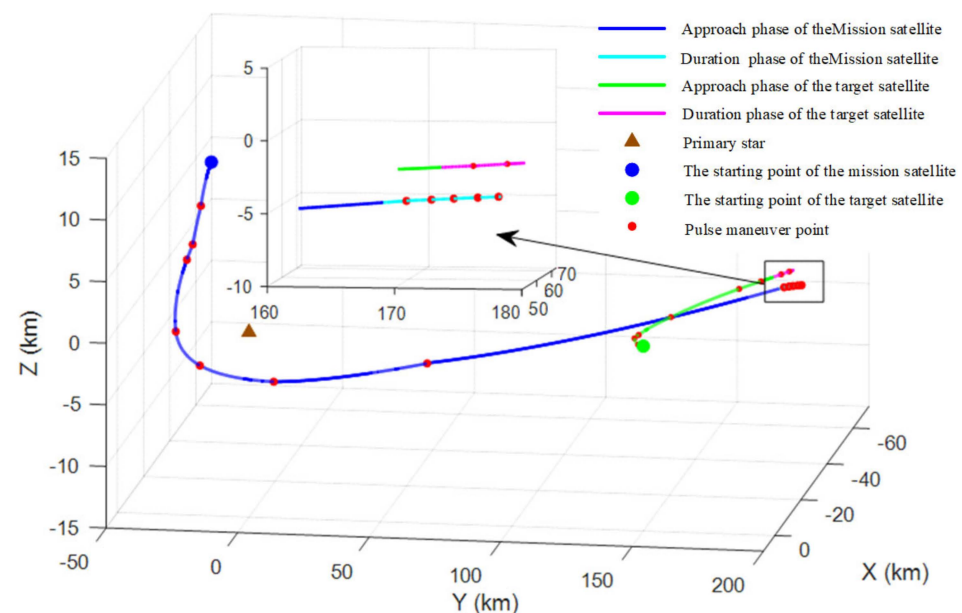
Table 3. Scenario 1: Impulse maneuver parameters of the mission satellite.

Phase	Time (s)	Frequency	Δv (m/s)	α (°)	β (°)
Approach Phase	4368	1	1.047	−1.329	−0.502
	6312	2	1.803	−2.163	0.188
	7334	3	0.361	−0.822	−0.384
	11,775	4	1.168	0.287	0.570
	16334	5	0.307	1.036	0.111
	21,326	6	0.192	0.384	0.657
	27,645	7	0.456	0.128	−0.257
End of Approach Phase	39,320				
Sustained Phase	39,520	8	1.150	−0.334	−0.230
	39,720	9	0.505	0.865	0.148
	39,920	10	0.781	−1.839	−0.171
	40,120	11	0.052	−1.743	0.048
	40,320	12	0.965	−0.469	−0.230
End of Sustained Phase	40,320				

The motion trajectories of the mission satellite and the target satellite in the primary satellite's coordinate system are shown in Figure 5. It can be seen that both the mission satellite and the target satellite performed multiple impulse maneuvers to adjust their trajectories. The mission satellite aimed to transfer to the capture area of the target satellite by adjusting its trajectory and remain in this area for a sufficient duration to effectively suppress the target satellite. In contrast, the target satellite adopted an evasion strategy, using impulse maneuvers to avoid the approach of the mission satellite as much as possible. The frequent impulse control of the mission satellite during the sustained phase successfully compensated for the trajectory deviation caused by the target satellite's evasion maneuvers, ensuring that the target satellite remained within the capture area and preventing its escape. This fully demonstrated the initiative and flexibility of both sides in their control strategies.

Table 4. Scenario 1: Impulse maneuver parameters of the target satellite.

Phase	Time (s)	Frequency	Δv (m/s)	α (°)	β (°)
Approach Phase	15,908	1	0.116	0.367	−0.792
	17,943	2	0.792	−1.723	0.170
	21,295	3	0.867	−2.022	−0.061
	24,124	4	0.571	−0.940	−0.405
	28,311	5	0.256	−1.360	0.311
	32,867	6	0.975	−1.388	−0.043
	37,728	7	0.506	0.485	0.126
	38,782	8	0.080	0.290	0.653
End of Approach Phase	39,320				
Sustained Phase	39,725	9	1.000	0.340	0.172
	40,130	10	0.336	−0.781	−0.062
End of Sustained Phase	40,320				

**Figure 5.** Scenario 1: The spatial path of the mission.

The changes in relative distance and capture angle between the mission satellite and the target satellite during the approach and sustained phases are shown in Figures 6 and 7. According to the results, the mission satellite achieved the transfer and maintenance of the capture area of the target satellite during both phases, mainly reflected in the coordinated changes and dynamic game process of relative distance and capture angle. In the approach phase, the mission satellite gradually optimized its trajectory through multiple impulse maneuvers, reducing the relative distance to the target satellite from a larger value to about 20 km. At the same time, the capture angle also decreased continuously. When the capture angle dropped to about 7° , the mission transitioned from the approach phase to the sustained phase, indicating that the mission satellite had successfully entered the capture area of the target satellite, laying a good foundation for subsequent maintenance within this area. After entering the sustained phase, due to the evasion strategy adopted by the target satellite, there were fluctuations in both the relative distance and capture angle, indicating that the game was still ongoing. However, the mission satellite was still able to correct and optimize its trajectory through multiple impulse maneuvers, stabilizing the relative

distance within the range of 20 km and maintaining the capture angle within 10° for 1000 s. This process fully demonstrates that, in the face of a dynamic confrontation environment, the mission satellite can still achieve successful transfer to and long-term suppression of the target satellite's capture area.

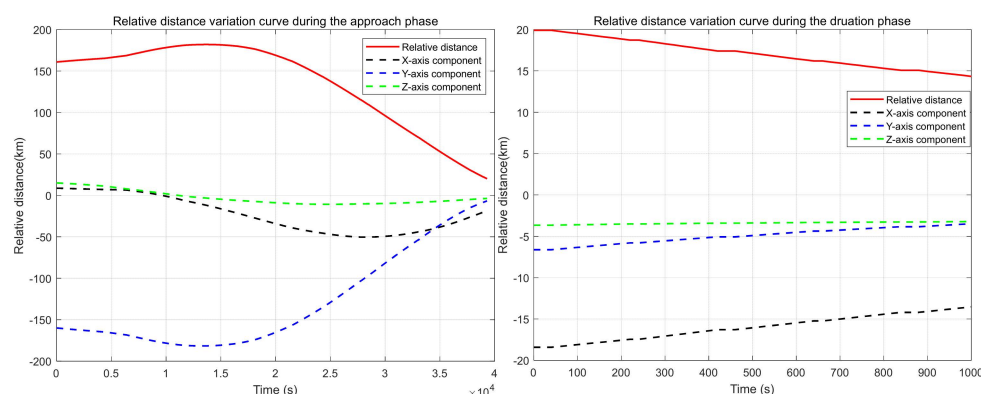


Figure 6. Scenario 1: Relative distance variation.

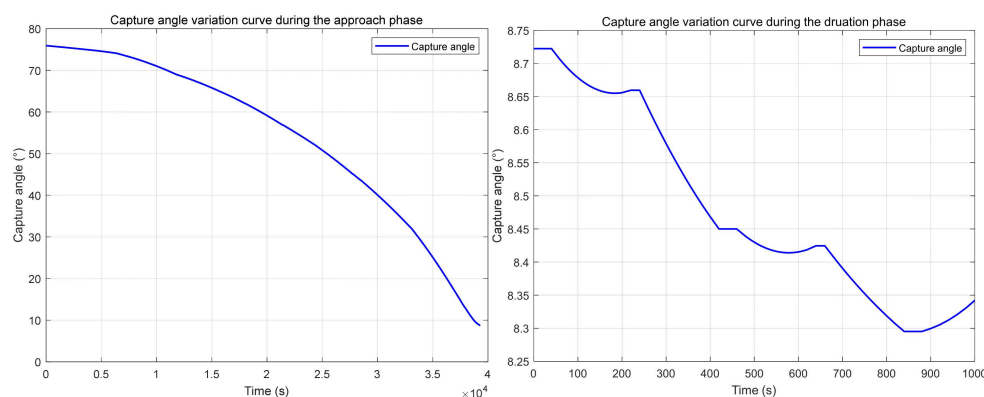


Figure 7. Scenario 1: Variation in capture angle.

From the perspective of mission execution results, the total mission duration was 40,320 s. During this period, the mission satellite successfully transferred to the capture area of the target satellite through 12 impulse maneuvers and remained within this area for 1000 s, successfully completing the mission objectives set in the mission scenario. The total velocity increment of the mission satellite was approximately 8.80 m/s. The target satellite executed 10 impulse maneuvers during the entire confrontation and game process, with a total velocity increment of approximately 5.4996 m/s.

6.2. Scenario 2: The Agility of the Mission Satellite Is 1.5 That of the Target Satellite

In order to test the robustness of the proposed method across different agility levels, the task satellite has a lower agility (1.5 times that of the target satellite), and the following adjustments were made: the time interval between two pulse maneuvers for the task satellite was set to no less than 200 s, and for the target satellite, it was set to no less than 300 s. The impulse maneuver parameters of the mission and target satellites are presented in Tables 5 and 6.

Table 5. Scenario 2: Impulse maneuver parameters of the mission satellite.

Phase	Time (s)	Frequency	Δv (m/s)	α (°)	β (°)
Approach Phase	1618	1	0.451	2.542	0.942
	5164	2	1.598	−1.670	0.096
	18,037	3	0.980	2.425	−0.657
	36,101	4	0.891	−1.220	0.408
	46,508	5	0.345	0.067	0.636
End of Approach Phase	46,780				
Sustained Phase	46,980	6	0.444	−0.614	0.804
	47,181	7	0.805	−0.353	−0.260
	47,382	8	1.509	−1.183	0.239
	47,583	9	1.054	−0.070	0.730
End of Sustained Phase	47,780				

Table 6. Scenario 2: Impulse maneuver parameters of the target satellite.

Phase	Time (s)	Frequency	Δv (m/s)	α (°)	β (°)
Approach Phase	6524	1	0.230	−1.394	−0.008
	14,564	2	0.187	0.253	1.380
	21,307	3	0.337	−0.751	−0.141
	24,709	4	0.194	0.079	0.907
	36,273	5	0.569	−0.354	0.636
End of Approach Phase	46,780				
Sustained Phase	47,080	6	0.649	0.094	0.506
	47,381	7	0.739	−0.093	−0.131
	47,683	8	0.431	1.420	−0.026
End of Sustained Phase	47,780				

From the mission execution results from Figures 8–10, the total mission duration was 47,780 s. During this period, the task satellite successfully transferred to the target satellite's capture zone through nine pulse maneuvers and remained within this zone for 1000 s, thus completing the mission objectives as set in the scenario. The total velocity increment of the task satellite was approximately 8.527 m/s. Throughout the entire adversarial game process, the target satellite executed eight pulse maneuvers, with a total velocity increment of approximately 3.336 m/s. The reduced agility of the task satellite led to an extension of the mission execution time and an increase in the adversarial duration. Despite the increased total mission time, it can be seen from the execution results that the task satellite was still able to fulfill the mission requirements, indicating that the proposed method in this paper exhibits a certain level of robustness, even when the agility of the task satellite is reduced.

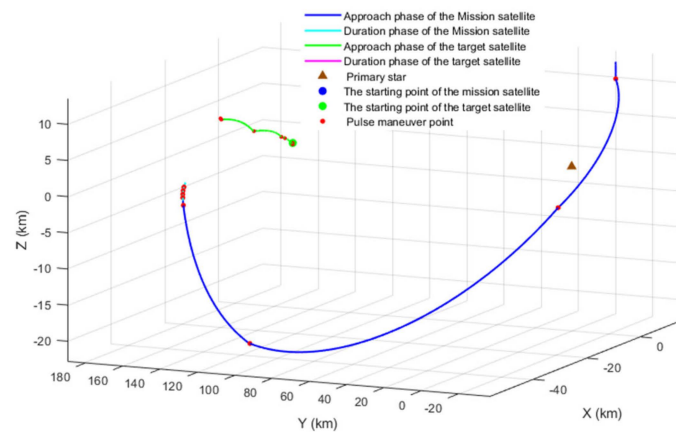


Figure 8. Scenario 2: The spatial path of the mission.

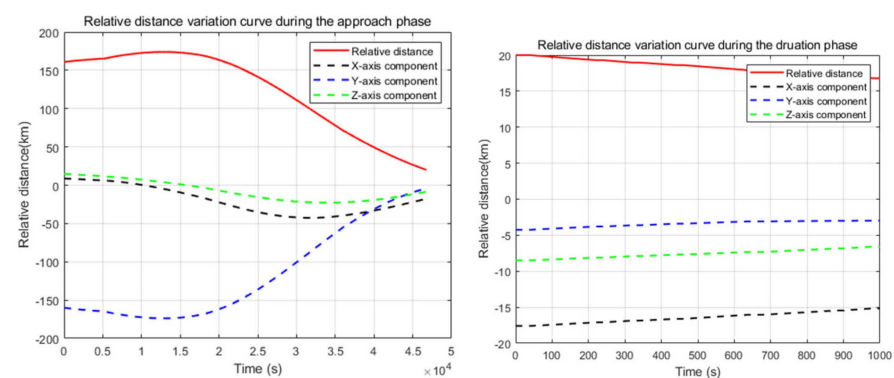


Figure 9. Scenario 2: The relative distance variation.

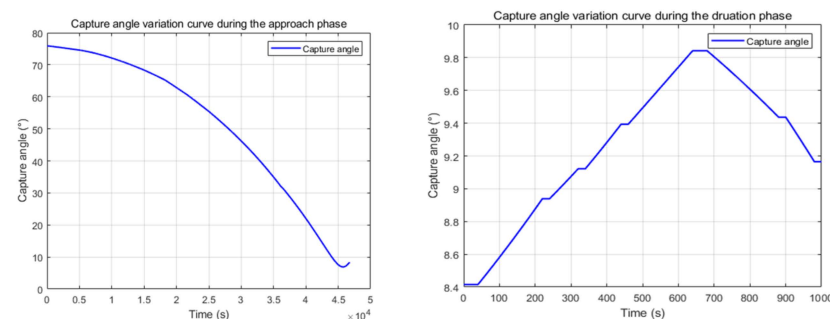


Figure 10. Scenario 2: Variation in capture angle.

7. Conclusions

This paper focuses on the problem of satellite pursuit–evasion games under impulse-thrust models, particularly on the transfer of a mission satellite to the capture area of a target satellite and its sustained presence for a certain duration. Based on the Zebra Optimization Algorithm (ZOA), coevolutionary mechanisms, and differential game theory, a coevolutionary-integrated algorithm is proposed to address satellite pursuit–evasion games in complex space environments. The orbital dynamics problem of impulse-thrust spacecraft is transformed into an optimization problem with multiple constraints. Innovatively, the mission is divided into two phases: the “approach phase” and the “sustained phase,” with optimization models established for each phase to achieve overall mission optimization through two-phase planning.

Simulation results indicate that the proposed coevolutionary-integrated algorithm effectively solves the satellite pursuit–evasion game problem under multiple constraints, exhibiting high mission success rates and strategy reliability. It can cope with maneuvering

changes of the target satellite in dynamic environments and meet the constraint requirements of complex missions. This method is not only applicable to the scenarios described in this paper but can also be extended to other complex scenarios such as space blockades and space exploration.

In future works, it will be intriguing to simulate cooperative pursuit–evasion games among multiple task satellites and target satellites by increasing the number of satellite groups and constraints, and enhancing the complexity of information sharing and strategy adjustment within the coevolution mechanism among multiple satellites. In addition, this study is based on the assumption of it being a complete information game, but in actual missions, the maneuvering strategy of the target satellite may have some uncertainty. In future research, probability game or robust optimization methods can be used to enable the mission satellite to adapt to the orbital adjustment behavior of uncertain targets.

Author Contributions: Conceptualization, J.W., X.X., Q.Y. and D.Z.; methodology, J.W., H.H. and D.Z.; software, J.W., H.H. and D.Z.; validation, X.X., Q.Y., H.H. and D.Z.; formal analysis, J.W., X.X. and Q.Y.; investigation, J.W., X.X. and Q.Y.; resources, J.W., X.X. and Q.Y.; data curation, J.W., H.H. and D.Z.; writing—original draft preparation, J.W., H.H. and D.Z.; writing—review and editing, J.W., H.H. and D.Z.; visualization, J.W., H.H. and D.Z.; supervision, Q.Y. and D.Z.; project administration, J.W., Q.Y. and D.Z.; funding acquisition, J.W., Q.Y. and D.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Shanghai Academy of Spaceflight Technology Funded project (SAST2023-088), National Natural Science Foundation of China (52377215, 51977177), Shaanxi Province Key Research and Development Plan (2024CY-GJHX-41), the “Double world-class project” discipline construction fund of Northwestern Polytechnical University (WH00001049), NSFC Projects (No. 12171481).

Data Availability Statement: The dataset is available on request from the authors.

Conflicts of Interest: Authors Xusheng Xu and Qiufan Yuan are employed by the Aerospace System Engineering Shanghai. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Liao, T. Research on Pursuit-Evasion Game Control and Solution Methods for Spacecraft. Master’s Thesis, Harbin Institute of Technology, Harbin, China, 2021; pp. 1–2.
2. Shen, D.; Pham, K.; Blasch, E.; Chen, H.; Chen, G. Pursuit-evasion orbital game for satellite interception and collision avoidance. In *Sensors and Systems for Space Applications IV*; SPIE: San Francisco, CA, USA, 2011; Volume 8044, pp. 89–97.
3. Luo, Y.Z.; Li, Z.Y.; Zhu, H. Survey on spacecraft orbital pursuit-evasion differential games. *Sci. Sin. Technol.* **2020**, *50*, 1533–1545. (In Chinese) [[CrossRef](#)]
4. Zhao, L.R.; Dang, Z.H.; Zhang, Y.L. Orbital game: Concepts, principles and methods. *J. Command Control* **2021**, *7*, 215–224. (In Chinese)
5. Isaacs, R. Differential Games I: Introduction. 1954. Available online: https://www.rand.org/pubs/research_memoranda/RM1391.html (accessed on 17 March 2025).
6. Ho, Y.; Bryson, A.; Baron, S. Differential games and optimal pursuit-evasion strategies. *IEEE Trans. Autom. Control* **1965**, *10*, 385–389. [[CrossRef](#)]
7. Pontani, M.; Conway, B.A. Numerical solution of the three-dimensional orbital pursuit-evasion game. *J. Guid. Control. Dyn.* **2009**, *32*, 474–487. [[CrossRef](#)]
8. Carr, R.W.; Cobb, R.G.; Pachter, M.; Pierce, S. Solution of a pursuit–evasion game using a near-optimal strategy. *J. Guid. Control. Dyn.* **2018**, *41*, 841–850. [[CrossRef](#)]
9. Pang, B.; Wen, C.X.; Han, H.W.; Qiao, D. Solving Pursuit/Evasion Game Along Elliptical Orbit by Providing Precise Gradient. *J. Guid. Control. Dyn.* **2024**, *47*, 797–807. [[CrossRef](#)]
10. Zhou, X.Y.; Cheng, Y.; Qiao, D.; Huo, Z.X. An adaptive surrogate model-based fast planning for swarm safe migration along halo orbit. *Acta Astronaut.* **2022**, *194*, 309–322. [[CrossRef](#)]

11. Armellin, R. Collision avoidance maneuver optimization with a multiple-impulse convex formulation. *Acta Astronaut.* **2021**, *186*, 347–362. [[CrossRef](#)]
12. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An introduction*; MIT Press: Cambridge, MA, USA, 2018.
13. Huang, X.; Li, S.; Yang, B.; Sun, P.; Liu, X.; Liu, X. Spacecraft guidance and control based on artificial intelligence: Review. *Acta Aeronaut. Astronaut. Sin.* **2021**, *42*, 106–121. (In Chinese)
14. Izzo, D.; Martens, M.; Pan, B.F. A survey on artificial intelligence trends in spacecraft guidance dynamics and control. *Astrodynamics* **2019**, *3*, 287–299. [[CrossRef](#)]
15. Zhu, Q.; Shao, Z.J. Missile real-time receding horizon pursuit and evasion games guidance based on neural network. *Syst. Eng. Elect* **2019**, *41*, 1597–1605.
16. Yang, B.; Liu, P.; Feng, J.; Li, S. Two-stage pursuit strategy for incomplete-information impulsive space pursuit-evasion mission using reinforcement learning. *Aerospace* **2021**, *8*, 299. [[CrossRef](#)]
17. Chu, X.; Alfriend, K.T.; Zhang, J.; Zhang, Y. Q-learning algorithm for path-planning to maneuver through a satellite cluster. In Proceedings of the AAS/AIAA Astrodynamics Specialist Conference, Snowbird, UT, USA, 19–23 August 2018; Univelt Inc.: Escondido, CA, USA, 2018; pp. 2063–2082.
18. Liu, B.Y.; Ye, X.B.; Gao, Y.; Wang, X.B.; Ni, L. Solution of non-cooperative pursuit-evasion game strategy based on branch deep reinforcement learning. *Acta Aeronaut. Astronaut. Sin.* **2020**, *41*, 348–358. (In Chinese)
19. Xu, X.S.; Dang, C.H.; Song, B.; Yuan, Q.F.; Xiao, Y.Z. Orbital pursuit-evasion game method based on multi-agent reinforcement learning. *Aerosp. Shanghai* **2022**, *39*, 24–31. (In Chinese)
20. Zhao, L.; Zhang, Y.; Dang, Z. PRD-MADDPG: An efficient learning-based algorithm for orbital pursuit-evasion game with impulsive maneuvers. *Adv. Space Res.* **2023**, *72*, 211–230. [[CrossRef](#)]
21. Prince, E.R.; Hess, J.A.; Cobb, R.G.; Carr, R.W. Elliptical orbit proximity operations differential games. In *Handbook of Scholarly Publications from the Air Force Institute of Technology (AFIT), Volume 1, 2000–2020*; CRC Press: Boca Raton, FL, USA, 2022; pp. 243–275.
22. Yu, D.T. Research on Maneuvering Methods for Spacecraft Safety Protection and Evasion. Master's Thesis, National University of Defense Technology, Changsha, China, 2017. (In Chinese).
23. Qi, Y.H. Research on Tracking Interception Guidance for Small Satellites. Ph.D. Thesis, Harbin Institute of Technology, Harbin, China, 2009; pp. 21–25. (In Chinese).
24. Wu, W.; Chen, J.; Liu, J. A hybrid optimisation method for intercepting satellite trajectory based on differential game. *Aeronaut. J.* **2023**, *127*, 900–922. [[CrossRef](#)]
25. Stupik, J.; Pontani, M.; Conway, B. Optimal pursuit/evasion spacecraft trajectories in the hill reference frame. In Proceedings of the AIAA/AAS Astrodynamics Specialist Conference, Minneapolis, MN, USA, 13–16 August 2012; p. 4882.
26. Sun, S.; Zhang, Q.; Loxton, R.; Li, B. Numerical solution of a pursuit-evasion differential game involving two spacecraft in low earth orbit. *J. Ind. Manag. Optim. (JIMO)* **2015**, *11*, 1127–1147. [[CrossRef](#)]
27. Liu, Y.; She, H.; Meng, B.; Huang, J. A Method of Surrounding Escapable Space Target by Combining Game with Optimization. In Proceedings of the 2023 42nd Chinese Control Conference (CCC), Tianjin, China, 24–26 July 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 8088–8093.
28. Wu, Q.C.; Zhang, H.B. Spacecraft pursuit-evasion strategy and numerical solution based on survival-type differential game. *Control. Inf. Technol.* **2019**, *4*, 39–43. (In Chinese)
29. Chen, W.; Hu, Y.; Gao, C.; Jing, W. Luring cooperative capture guidance strategy for the pursuit-evasion game under incomplete target information. *Astrodynamics* **2024**, *8*, 675–688. [[CrossRef](#)]
30. Sun, Y.; Chen, Y. Multi-population improved whale optimization algorithm for high dimensional optimization. *Appl. Soft Comput.* **2021**, *112*, 107854. [[CrossRef](#)]
31. Mallipeddi, R.; Suganthan, P.N. Ensemble of constraint handling techniques. *IEEE Trans. Evol. Comput.* **2010**, *14*, 561–579. [[CrossRef](#)]
32. Tian, Y.; Zhang, T.; Xiao, J.; Zhang, X.; Jin, Y. A coevolutionary framework for constrained multiobjective optimization problems. *IEEE Trans. Evol. Comput.* **2020**, *25*, 102–116. [[CrossRef](#)]
33. He, C.; Li, M.; Zhang, C.; Chen, H.; Zhong, P.; Li, Z.; Li, J. A self-organizing map approach for constrained multi-objective optimization problems. *Complex Intell. Syst.* **2022**, *8*, 5355–5375. [[CrossRef](#)]
34. Trojovská, E.; Dehghani, M.; Trojovský, P. Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm. *IEEE Access* **2022**, *10*, 49445–49473. [[CrossRef](#)]
35. Zhu, H. Optimal Control Strategy for Spacecraft Orbital Pursuit-Evasion Based on Differential Game. Master's Thesis, National University of Defense Technology, Changsha, China, 2017. (In Chinese)

36. Zhu, H.; Luo, Y.Z.; Li, Z.Y.; Yang, Z. Orbital pursuit-evasion games with incomplete information in the hill reference frame. In Proceedings of the 27th International Symposium on Space Flight Dynamics (ISSFD), Melbourne, Australia, 24–28 February 2019; pp. 1–6.
37. Wang, Z.K. Research on Dynamics Modeling and Control of Distributed Satellites. Ph.D. Thesis, National University of Defense Technology, Changsha, China, 2006; pp. 18–41. (In Chinese)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.