

On-orbit refueling robust mission scheduling with uncertain duration for geosynchronous orbit spacecraft

*Original*

On-orbit refueling robust mission scheduling with uncertain duration for geosynchronous orbit spacecraft / Yin, Shuai; Li, Chuanjiang; Fadda, Edoardo; Guo, Yanning; Ran, Guangtao; Brandimarte, Paolo. - In: CHINESE JOURNAL OF AERONAUTICS. - ISSN 1000-9361. - ELETTRONICO. - 39:1(2026), pp. 1-15. [10.1016/j.cja.2025.103774]

*Availability:*

This version is available at: 11583/3002481 since: 2025-11-27T14:47:31Z

*Publisher:*

Elsevier

*Published*

DOI:10.1016/j.cja.2025.103774

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)



Chinese Society of Aeronautics and Astronautics  
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn  
www.sciencedirect.com



FULL LENGTH ARTICLE

# On-orbit refueling robust mission scheduling with uncertain duration for geosynchronous orbit spacecraft



Shuai YIN<sup>a</sup>, Chuanjiang LI<sup>a</sup>, Edoardo FADDA<sup>b</sup>, Yanning GUO<sup>a,c</sup>,  
Guangtao RAN<sup>a,\*</sup>, Paolo BRANDIMARTE<sup>b</sup>

<sup>a</sup> Department of Control Science and Engineering, Harbin Institute of Technology, Harbin 150001, China

<sup>b</sup> Department of Mathematical Sciences 'Giuseppe Luigi Lagrange', DISMA, Politecnico di Torino, Torino 10129, Italy

<sup>c</sup> Zhengzhou Research Institute, Harbin Institute of Technology, Zhengzhou 450000, China

Received 27 October 2024; revised 20 January 2025; accepted 6 May 2025

Available online 19 August 2025

## KEYWORDS

Geosynchronous orbit (GEO);  
Hybrid Harris Hawks Optimization algorithm (HHHO);  
Mission scheduling;  
On-orbit refueling;  
Robust optimization

**Abstract** With the increasing number of geosynchronous orbit satellites with expiring lifetime, spacecraft refueling is crucial in enhancing the economic benefits of on-orbit services. The existing studies tend to be based on predetermined refueling duration; however, the precise mission scheduling solution will be difficult to apply due to uncertain refueling duration caused by orbital transfer deviations and stochastic actuator faults during actual on-orbit service. Therefore, this paper proposes a robust mission scheduling strategy for geosynchronous orbit spacecraft on-orbit refueling missions with uncertain refueling duration. Firstly, a robust mission scheduling model is constructed by introducing the budget uncertainty set to describe the uncertain refueling duration. Secondly, a hybrid harris hawks optimization algorithm is designed to explore the optimal mission allocation and refueling sequences, which combines cubic chaotic mapping to initialize the population, and the crossover in the genetic algorithm is introduced to enhance global convergence. Finally, the typical simulation examples are constructed with real-mission scenarios in three aspects to analyze: performance comparisons with various algorithms; robustness analyses via comparisons of different on-orbit refueling durations; investigations into the impacts of different initial population strategies on algorithm performance, demonstrating the proposed mission scheduling framework's robustness and effectiveness by comparing it with the exact mission scheduling.

© 2025 The Authors. Published by Elsevier Ltd on behalf of Chinese Society of Aeronautics and Astronautics. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

\* Corresponding author.

E-mail address: [ranguangtao@hit.edu.cn](mailto:ranguangtao@hit.edu.cn) (G. RAN).

Peer review under responsibility of Editorial Committee of CJA.



<https://doi.org/10.1016/j.cja.2025.103774>

1000-9361 © 2025 The Authors. Published by Elsevier Ltd on behalf of Chinese Society of Aeronautics and Astronautics. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the increasing number of spacecraft launch missions, On-Orbit Servicing (OOS) technologies, mainly On-Orbit Refueling (OOR),<sup>1,2</sup> debris removal,<sup>3-6</sup> and maintenance<sup>7,8</sup> have become the urgent need to achieve the purpose of sustainable space development. The OOR mission is one of the typical OOS missions, which can effectively reduce the cost of space transportation, enhance the spacecraft's adaptability, extend the spacecraft's lifetime, and facilitate the spacecraft's full-load capacity. In addition, Geosynchronous Orbit (GEO) is a precious orbit that plays a vital role in observation, communication, and navigation. According to the data published on the Union of Concerned Scientists website, 590 spacecraft are currently running in GEO orbit until May 1, 2023, due to the limited number of spacecrafts parked in GEO and the high cost of most spacecraft. Therefore, it is crucial to consider the GEO spacecraft OOR mission scheduling problem.

It has been shown that a series of OOR missions have been carried-out by various countries in recent years, such as the orbital express program,<sup>9</sup> the tianyuan-1 program,<sup>10</sup> the tianzhou series of cargo spacecraft,<sup>11</sup> in particular, the Mission Expansion Vehicle (MEV)-1 and MEV-2, the robotic refueling mission,<sup>12</sup> which are refueling missions against GEO targets. The OOS modes extend to one-to-many,<sup>13</sup> many-to-many,<sup>14</sup> and peer-to-peer.<sup>15</sup> However, relying on ground-launched spacecraft for refueling missions has limitations owing to the limited fuel carrying capacity of spacecraft. Therefore, the scholars proposed to consider the construction of Fuel Station (FS) in space for spacecraft fuel refueling concerning the ground vehicle refueling method,<sup>16</sup> which introduces the idea of Service Spacecraft (SSc) fuel recycling refueling to build up a more flexible OOR mode to improve operating economy of OOS. Meng et al.<sup>17</sup> compared three orbital deployment modes, including ground deployment, near-earth orbit deployment, and co-orbital deployment, and proposed the service structure of one FS and multiple SSc. Zhu et al.<sup>18</sup> proposed the problem of FS orbitalization in sun-synchronous orbit spacecraft refueling mission under J2 perturbation. Han et al.<sup>19</sup> adopted the two-stage optimization framework to address the FS orbit design and spacecraft refueling mission scheduling for sun-synchronous orbits. Based on this model, the paper considers one FS in space and multiple round-trip SSc to refuel the target.

The refueling duration is an item to be considered in the OOR mission scheduling since the fuel of the SSc takes a period to refuel the target. Most existing research on OOS mission scheduling problems adopt the fixed refueling duration.<sup>19-21</sup> Nevertheless, the space situation is comprehensive and changeable, on the one hand, there is the transfer accuracy error caused by the ingestion factor in the orbit maneuvering process, the spacecraft needs time to correct to the predetermined orbit, especially in a high orbit such as GEO, where the orbit maneuvering accuracy will further deteriorate; on the other hand, during refueling with actuators such as robots, due to the possibility of actuator faults, the refueling duration are strongly stochastic as opposed to predetermined. The original mission scheduling scheme based on the predetermined refueling duration becomes suboptimal or even infeasible.

The method for solving the mission scheduling problem with uncertainty is divided into two categories. One is a reactive approach, including replanning,<sup>22</sup> rolling adjustment<sup>23</sup> and reinforcement learning.<sup>24-26</sup> The other is the predictive approach, including stochastic planning,<sup>27</sup> fuzzy planning<sup>28</sup> and robust optimization.<sup>29</sup> The reactive approach instantly adjusts the scheduling scheme according to the uncertain situation. However, frequent corrections will dramatically increase the cost of resource allocation. In contrast, the predictive approach considers the influence of uncertainty factors in advance so that the obtained scheduling scheme can be better adapted to the changes in the actual mission execution situation. This paper adopts the robust optimization approach to address the mission scheduling problem of uncertain refueling duration, unlike fuzzy planning and stochastic planning, it is not restricted by known historical data and distribution function information but only needs to consider uncertainty set selection. Compared with the solution proposed by Liang et al.<sup>30</sup> to address the uncertainty issues in on-orbit missions, which derives worst-case optimal solutions via enumeration and extreme value analysis. However, this approach leads to strong conservatism and high computational costs.

Meta-heuristic algorithms have received the keen attention of scholars for their excellent ability to search for optimal solutions in complex optimization problems.<sup>31</sup> Zhang et al.<sup>32</sup> adopted an ant colony optimization meta-heuristic algorithm to solve the many-to-many GEO spacecraft refueling problem. Wei et al.<sup>4</sup> presented a clustering-based adaptive differential evolution algorithm with potential individual reservation to obtain a scheduling optimization scheme for multiple hybrid propulsion servicing spacecraft to execute GEO space debris removal missions. Zhang et al.<sup>33</sup> proposed a hybrid-encoding genetic algorithm to solve the multi-spacecraft refueling problem considering J2 perturbation and rendezvous time window constraints. Xu et al.<sup>34</sup> addressed the OOS mission planning problem for multiple GEO satellites under highly constrained conditions, in which an improved auction algorithm is considered for initialization.

In the past few years, the Harris Hawks Optimization (HHO) algorithm, emerging as a newly introduced and readily-implementable bio-inspired algorithm, has garnered substantial interest among a wide range of scholars.<sup>35</sup> Sassi and Chelouah<sup>36</sup> presented a novel HHO encirclement attack synergy, designed to surpass the original HHO in handling highly multi-modal and high-dimensional optimization problems. Akhter et al.<sup>37</sup> proposed the method to solve the application placement problem in the space-air-ground integrated network using the dynamic weight-configurable HHO algorithm. To address the communication environment challenges in cloud-fog computing, Ali et al.<sup>38</sup> proposed a multi-objective HHO-based mission scheduling algorithm. Zhang et al.<sup>39</sup> proposed a multi-strategy sparrow search enhanced distance vector-hop localization algorithm to address the issues of sea surface sensor node localization accuracy, which updates the sparrow population discoverer's localization with the HHO combined with differential evolution to enhance population diversity. Gharehchopogh et al.<sup>40</sup> introduced an innovative binary multi-objective dynamic HHO algorithm enhanced with mutation operator, and applied it to botnet detection in the

internet of things ecosystem to bolster the security. The remarkable performance of HHO algorithm in achieving excellent and stable optimization results within the realm of nonlinear problems can be primarily ascribed to its multifarious search strategies, elaborate stage-transition mechanism, and adaptive adjustments. Nevertheless, the optimization procedure of meta-heuristic algorithms typically encompasses two distinct phases: exploration and exploitation. The crux of the matter is to strike an appropriate equilibrium between these two aspects. When confronted with diverse real-world problems, meta-heuristic algorithms often encounter challenges such as being entrapped in local optimal solutions and exhibiting sluggish convergence rates. In order to solve the robust mission scheduling model developed in this paper to enhance the global exploitation and exploration of the problem, the HHO algorithm is adopted. Meanwhile, to further enhance its ability, a Hybrid Harris Hawks Optimization (HHO) algorithm is proposed by introducing the crossover idea of Genetic Algorithm (GA) to enhance the global exploration ability, avoid falling into local optimal, and improve the problem-solving iteration efficiency.

The main contributions of this paper can be summarized as follows:

- (1) In order to address the execution risk in actual OOR mission, a robust mission scheduling framework is constructed considering the uncertain refueling duration;
- (2) A HHHO algorithm is proposed to solve the OOR robust mission scheduling problem with uncertain duration, which introduces the crossover idea of GA, overcomes the problem that the HHO algorithm has insufficient exploratory ability and improves the optimization solving efficiency.

The remainder of this work is organized by the following parts. [Section 2](#) constructs a robust mission scheduling framework, including orbital rendezvous strategies, fuel mass computation, and transformation of the problem with uncertain refueling duration into the robust mission scheduling model. [Section 3](#) describes the designed HHHO algorithm. [Section 4](#) gives the simulation examples in the real-world scenario and related comparative experiments to verify the effectiveness of the robust mission scheduling strategy proposed in this paper. Finally, [Section 5](#) summarizes the conclusions of this paper and directions for future research on the problem of OOS with uncertainty.

## 2. Problem formulation

This section presents the mission scenario of spacecrafts refueling, which is considered “one fixed FS + round-trip SSc” servicing mode.<sup>17</sup> Then, the orbital transfer method of planar change and phasing maneuver are introduced, respectively. Meanwhile, the calculation of the propellant consumption is given during the refueling process. Finally, the robust refueling scheduling optimization problem is described mathematically.

### 2.1. Mission scenario

In this work, considering the construction cost and rationality of the FS, the “one fixed FS + round-trip SSc” refueling

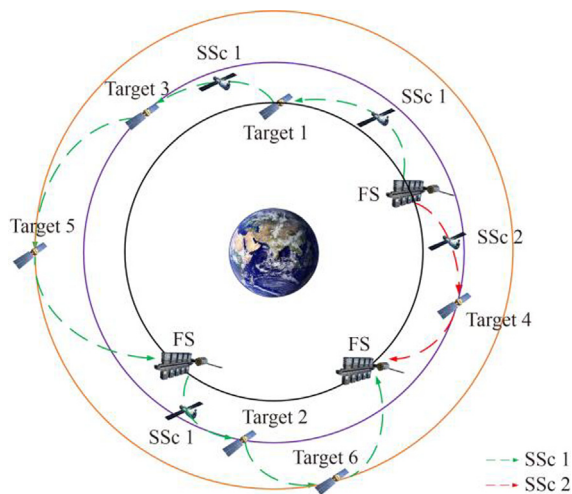


Fig. 1 Scenario of GEO spacecraft refueling.

mode, which belongs to the many-to-many OOS mode, is adopted for the GEO spacecraft refueling problem, i.e., there is an FS on GEO, which carries multiple SSc to and from the FS to refuel targets. The scenario of the GEO spacecraft refueling is given in [Fig. 1](#). The phases of the spacecraft refueling process are introduced, and the GEO spacecraft refueling process is graphically shown in [Fig. 2](#).

**Phase 1:** FS is deployed on GEO circular orbit carrying enough fuel, and a limited number of SSc are parked in FS waiting to execute the mission. At the same time, multiple target spacecrafts are waiting to be refueled in GEO at the same orbital altitude with different orbital inclinations and Right Ascensions of Ascending Nodes (RAANs).

**Phase 2:** At the beginning of the refueling mission, all targets are allocated to different SSc. Meanwhile, the order of servicing and when to return to the FS for refueling are determined for each SSc.

**Phase 3:** Each SSc rendezvous with the target via orbital transfer and performed OOR missions via the payload.

**Phase 4:** Each SSc refuels the next target in the specified order of service until all targets have been refueled.

**Phase 5:** When all the refueling missions are finished, each SSc returns to the FS.

This paper is dedicated to studying the GEO spacecraft’s robust mission scheduling problem with uncertain refueling duration. The specific assumptions are as follows:

- (1) The orbits of the target and the SSc are assumed to be circular, and targets are given equal priority.
- (2) The service time of each SSc is independent, and uncertainty is only considered in the case of refueling duration.
- (3) Uncertainty affects only the refueling duration.
- (4) Thrust engine operating characteristics are not considered, and it is assumed that velocity increments can be done instantaneously.
- (5) The target attitude motion is not considered, and the rendezvous is considered to be completed when the SSc reaches the predetermined position.
- (6) The fuel contained at the FS is sufficient.

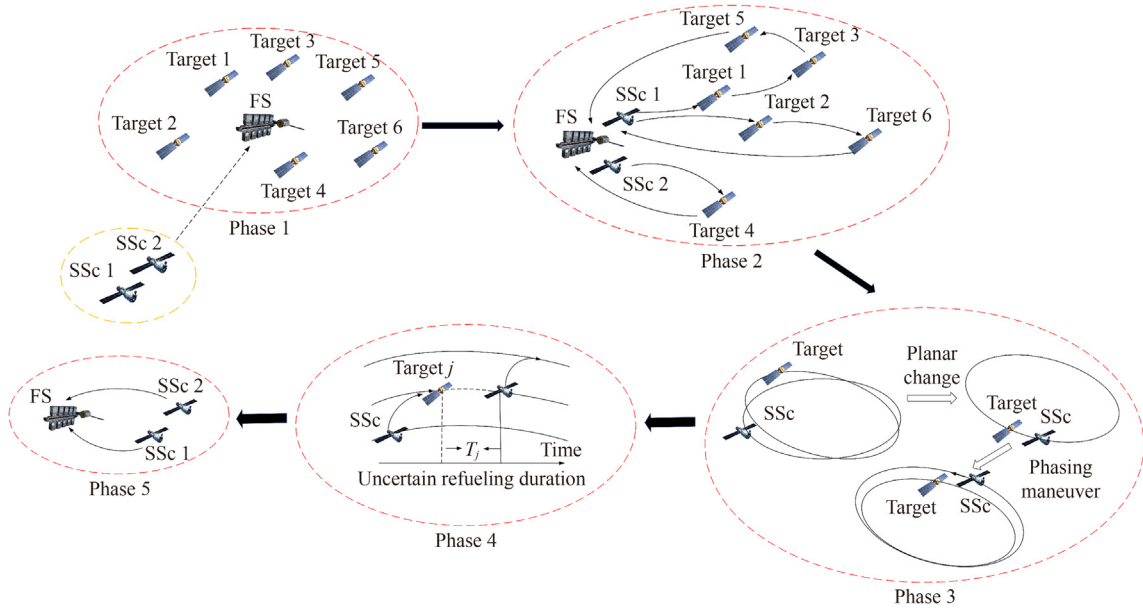


Fig. 2 GEO spacecraft refueling process.

## 2.2. Orbit transfer strategy

This article studied the spacecraft refueling problem, which involves the rendezvous between SSc and target spacecraft in different orbits. First, the planar change method is considered to adjust the SSc's orbital plane to the target spacecraft during the rendezvous transfer. Then, the phasing maneuver is performed to achieve rendezvous.

### 2.2.1. Planar change

The planar change is adopted to effectively adjust the orbital inclination and RAANs of the SSc to the targets at circular orbits. The SSc applies a pulse at the intersection of the two orbits to achieve consistency with the target's orbital plane, that is, the nearest intersection of the orbital plane is chosen from the two points  $P_1$  and  $P_2$ , the planar change process is shown in Fig. 3.

In the planar change process, the key physical quantities are the velocity increment for planar change  $\Delta v_{pc}$  and the angle between the two orbital surfaces  $\theta_{pc}$ . The velocity increment of the planar change orbital transfer is formulated as follows:

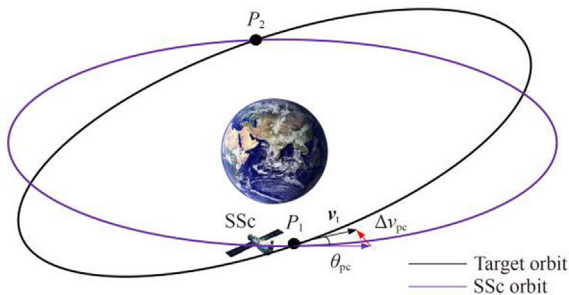


Fig. 3 Planar change process.

$$\Delta v_{pc} = 2|v_t| \sin\left(\frac{\theta_{pc}}{2}\right) \quad (1)$$

$$\Delta v_{pc} = 2|v_t| \sin\left(\frac{\theta_{pc}}{2}\right) \quad (1)$$

$$\theta_{pc} = \arccos(\sin I_s \sin I_t \cos(\Omega_s - \Omega_t) + \cos I_s \cos I_t) \quad (2)$$

where  $\Delta v_{pc}$  is the velocity increment required to adjust the orbital plane;  $v_t$  is the velocity of the GEO target to be refueled;  $\theta_{pc}$  is the angle between the two orbital surfaces;  $I_s$  and  $\Omega_s$  are the SSc's orbital inclination and RAAN, respectively.  $I_t$  and  $\Omega_t$  are the target's orbital inclination and RAAN, respectively. The planar change time  $T_{pc}$  is computed by Han et al.<sup>8</sup>

### 2.2.2. Phasing maneuver

After the planar change of orbit maneuver, there is still a phase difference between the SSc and the target. Therefore, the phase maneuver transfer strategy is adopted in the same-orbit case, and two pulses are applied at the exact position of the orbit

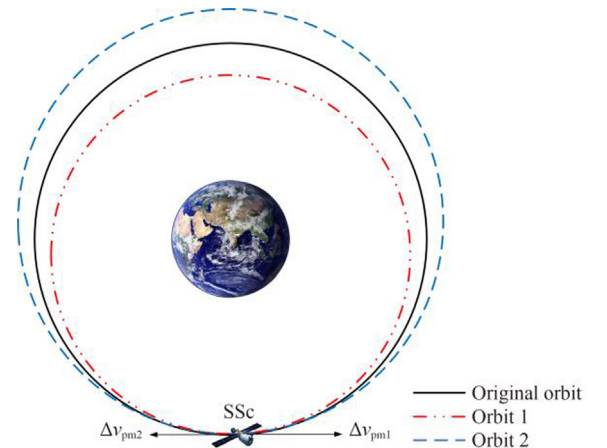


Fig. 4 Phasing orbit transfer diagram.

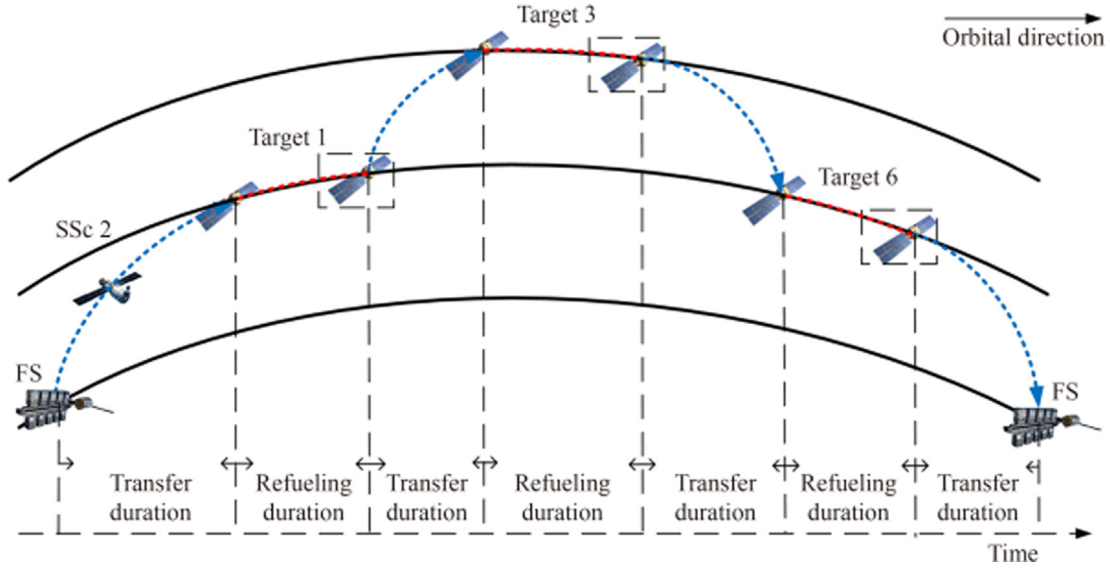


Fig. 5 Single mission refueling process.

to realize the rendezvous between the SSc and the target to be refueled. Fig. 4 shows the phasing orbit transfer diagram, when the target phase is ahead of the SSc, the SSc needs to brake ignition into orbit 1 to rendezvous with the target in a time less than the original orbital period. Otherwise, if the target phase is behind the SSc, the SSc requires pulse ignition to accelerate into orbit 2 to rendezvous with the target in a time more significant than the original orbital period.

The main physical quantities in the phasing maneuver process include the velocity increments for the phasing orbit transfer  $\Delta v_{pm1}$  and  $\Delta v_{pm2}$ , the semi-major axis of the phasing maneuver  $a_{pm}$ , and the time of the phasing maneuver  $T_{pm}$ . The velocity increments of the phasing orbit transfer are expressed as

$$\Delta v_{pm1} = \Delta v_{pm2} = \sqrt{\mu} \left| \sqrt{\left( \frac{2}{r_t} - \frac{1}{a_{pm}} \right)} - \sqrt{\frac{1}{r_t}} \right| \quad (3)$$

where  $\Delta v_{pm1}$  and  $\Delta v_{pm2}$  are the velocity increments for the first pulse and the second pulse in the phasing maneuver transfer. This formula is derived from the vis-viva equation in orbital mechanics, the Earth's standard gravitational parameter  $\mu$ , which is derived from the product of the universal gravitational constant and the Earth's mass. The distance from the central body (the target orbital radius  $r_t$ ), and the semi-major axis of the orbit  $a_{pm}$ .

The semi-major axis of phasing maneuver  $a_{pm}$  computed as

$$a_{pm} = r_t \left( \frac{2\pi n_{pm} + \Delta\theta_{pm}}{2\pi n_{pm}} \right)^{\frac{2}{3}} \quad (4)$$

where  $n_{pm}$  indicates the number of SSc's revolutions in the phasing maneuver;  $\Delta\theta_{pm}$  is the phase difference between the SSc and the target. Moreover, the time of the phasing maneuver  $T_{pm}$  is computed as

$$T_{pm} = \frac{2\pi n_{pm} + \Delta\theta_{pm}}{n_{pm} \omega_t} \quad (5)$$

where  $\omega_t$  is the angular velocity of target.

### 2.2.3. Velocity increment calculation

The total velocity incremental consumption calculation for a rendezvous with a single target can be expressed as

$$\Delta v = \Delta v_{pc} + \Delta v_{pm1} + \Delta v_{pm2} \quad (6)$$

### 2.3. Propellant consumption

Let  $\Psi = \{1, 2, \dots, K\}$  be the set of SSc, and let  $\gamma = \{1, 2, \dots, J\}$  be the set of targets each one of them needs to be refueled by a quantity  $m_j^{(R)}$ . We call  $\bar{\gamma} = \gamma \cup \{0\}$ , where 0 denotes the FS. Each SSc starts from the FS, refuel a subset of targets and return to the FS. We call mission the set of targets refueled continuously by each SSc. An example of mission is represented in Fig. 5, where SSc 2 starts from FS and refuels targets 1, 3, and 6 and returns to FS.

We indicate by  $\gamma_k^i \subset \gamma$  the set of targets visited by SSc  $k$  during its mission  $i$  and we call  $J_k$  the number of missions performed by SSc  $k$ . The set  $\gamma_k^i$  defines a partition of the set  $\gamma$ . For each of them, we define a permutation function  $\sigma_k^i(\cdot)$  which describes the visiting order of the targets in  $\gamma_k^i$ . For example, in Fig. 5,  $\sigma_2^1(1) = 1$ ,  $\sigma_2^1(2) = 3$ , etc. We fix  $\sigma_k^i(0) = \sigma_k^i(|\gamma_k^i| + 1) = 0$  and we use  $|\gamma_k^i|$  to indicate the cardinality of  $\gamma_k^i$  (thus,  $\sigma_k^i(|\gamma_k^i|)$  is the last target of mission  $i$ ).

The goal is to refuel all the targets by minimizing the quantity of fuel consumed. We model fuel consumption by the Tsilolkovsky equation, i.e., if SSc  $k$  has mass  $m_k^i$  after refueling target  $j_1$  and it moves to  $j_2$ , it will have a mass equal to

$$m_k^{j_2} = m_k^i e^{-\Delta v_{j_1 j_2} / (g_{sp}^k)} - m_{j_2}^{(R)} \quad (7)$$

after refueling target  $j_2$ . Where  $\Delta v_{j_1 j_2}$  is the change in speed needed to reach  $j_2$ ,  $g_{sp}^k$  is the specific impulse of the  $k$ th SSc engine, and  $g$  is the earth's gravity constant. Therefore, the final mass  $m_k^{(f)i}$  of SSc  $k$  after mission  $i$  is

$$\begin{aligned}
m_k^{(f)i} = & \left( \left( m_k^{(0)i} \exp(-\Delta v_{0\sigma_k^i(1)} / (gI_{sp}^k)) - m_{\sigma_k^i(1)}^{(R)} \right) \right. \\
& \cdot \exp(-\Delta v_{\sigma_k^i(1)\sigma_k^i(2)} / (gI_{sp}^k)) - m_{\sigma_k^i(2)}^{(R)} \Big) \dots \\
& \cdot \exp(-\Delta v_{\sigma_k^i(|\gamma_k^i|-1)\sigma_k^i(|\gamma_k^i|)} / (gI_{sp}^k)) - m_{\sigma_k^i(|\gamma_k^i|)}^{(R)} \Big) \\
& \cdot \exp(-\Delta v_{\sigma_k^i(|\gamma_k^i|)0} / (gI_{sp}^k)) \Big)
\end{aligned} \quad (8)$$

where  $m_k^{(0)i}$  is the initial mass of SSc  $k$  during mission  $i$ . Fuel consumption can be computed as  $m_k^{(0)i} - m_k^{(f)i}$ , which thanks to Eq. (8) can be written as

$$\begin{aligned}
m_k^{(0)i} - & \left( \left( m_k^{(0)i} \exp(-\Delta v_{0\sigma_k^i(1)} / (gI_{sp}^k)) - m_{\sigma_k^i(1)}^{(R)} \right) \right. \\
& \cdot \exp(-\Delta v_{\sigma_k^i(1)\sigma_k^i(2)} / (gI_{sp}^k)) - m_{\sigma_k^i(2)}^{(R)} \Big) \dots \\
& \cdot \exp(-\Delta v_{\sigma_k^i(|\gamma_k^i|-1)\sigma_k^i(|\gamma_k^i|)} / (gI_{sp}^k)) - m_{\sigma_k^i(|\gamma_k^i|)}^{(R)} \Big) \\
& \cdot \exp(-\Delta v_{\sigma_k^i(|\gamma_k^i|)0} / (gI_{sp}^k)) \Big) \\
= & \left( 1 - \exp(-\sum_{l=0}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k)) \right) m_k^{(0)i} \\
& + m_{\sigma_k^i(1)}^{(R)} \exp(-\sum_{l=1}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k)) + \dots \\
& + m_{\sigma_k^i(|\gamma_k^i|)}^{(R)} \exp(-\Delta v_{\sigma_k^i(|\gamma_k^i|)0} / (gI_{sp}^k))
\end{aligned} \quad (9)$$

From Eq. (9), we can notice that the fuel consumption is a quantity that is increasing in  $m_k^{(0)i}$ . Therefore, the optimal solution starts with the lowest possible value of  $m_k^{(0)i}$ , i.e., the one that ensure to come back with an empty tank to the FS. Leveraging this observation, we can compute  $m_k^{(0)i}$  from Eq. (8), setting  $m_k^{(f)i} = m_k^{(\text{dry})}$ ,  $m_k^{(\text{dry})}$  is SSc's dry mass, thus obtaining

$$\begin{aligned}
m_k^{(0)i} = & \left( m_{\sigma_k^i(1)}^{(R)} \exp(-\sum_{l=1}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k)) \right) + \\
& m_{\sigma_k^i(2)}^{(R)} \exp(-\sum_{l=2}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k)) + \dots \\
& + m_{\sigma_k^i(|\gamma_k^i|)}^{(R)} \exp(-\Delta v_{\sigma_k^i(|\gamma_k^i|)0} / (gI_{sp}^k)) + m_k^{(\text{dry})} \\
& / \left( \exp(-\sum_{l=0}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k)) \right) \\
= & m_{\sigma_k^i(1)}^{(R)} \exp(\Delta v_{0\sigma_k^i(1)} / (gI_{sp}^k)) + \\
& m_{\sigma_k^i(2)}^{(R)} \exp((\Delta v_{0\sigma_k^i(1)} + \Delta v_{\sigma_k^i(1)\sigma_k^i(2)}) / (gI_{sp}^k)) + \dots \\
& + m_{\sigma_k^i(|\gamma_k^i|)}^{(R)} \exp(\sum_{l=0}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k)) \\
& + m_k^{(\text{dry})} \exp(\sum_{l=0}^{|\gamma_k^i|-1} \Delta v_{\sigma_k^i(l)\sigma_k^i(l+1)} / (gI_{sp}^k))
\end{aligned} \quad (10)$$

From Eq. (10), it is possible to observe that in general, it is convenient to first refuel target with big  $m_j^{(R)}$  so that they impact less the fuel consumption of the trip.

The time spent by the SSc travelling from target  $j_1$  to target  $j_2$  is called  $t_{j_1j_2}$ . Here  $j_1$  and  $j_2$  can be zero and  $t_{j_1j_2}$  consider the fuel time of  $j_2$  if  $j_2 \neq 0$ . We compute it as the sum of the planar change  $T_{pc}$ , the phasing maneuver  $T_{pm}$  and the refuel duration of  $j_2$  (which is considered to be null if  $j_2 = 0$ ).

Using these definitions, we can write the optimization problem as

$$\min \sum_{\mathcal{J}^k, \sigma_k^i} \sum_{k \in \mathcal{K}} \sum_{i=1}^{J_k} (m_k^{(0)i} - m_k^{(\text{dry})}) \quad (11)$$

$$\text{s.t. } \sum_{j=0}^{|\gamma_k^i|-1} t_{\sigma_k^i(j)\sigma_k^i(j+1)} \leq T_k^{\max}, \forall k \in \Psi, \forall i \in Y_k \quad (12)$$

$$\bigcup_{k \in \Psi} \bigcup_{i=1}^{\gamma_k} \gamma_k^i = \gamma \quad (13)$$

$$\gamma_{k_1}^{i_1} \cap \gamma_{k_2}^{i_2} = \emptyset, \forall k_1 \neq k_2 \in \Psi, \forall i_1 \neq i_2 \in 1, 2, \dots, \gamma_k \quad (14)$$

The objective function Eq. (11) minimizes the sum of the fuel consumed. Eq. (12) limits the time of the mission to be less than  $T_k^{\max}$ , and Eqs. (13) and (14) enforce the sets  $\gamma_k^i$  to be a partition of  $\gamma$ .

On the real field, it is possible that time in Eq. (12) are not deterministic and factors such as faults in actuators (e.g., robotic arms), spacecraft orbital deviations cause a deviation in the refueling duration. This may lead to extra time that can be very dangerous for the mission. Therefore, we modify model Eqs. (11)–(14) in order to be more robust with respect to these risks. In particular, we first rewrite Eq. (12) using the binary decision variables  $y_{j_1j_2}^{ki}$  equal to one if target  $j_1$  is refueled just before target  $j_2$  in the  $i$ th mission of SSc  $k$ , and 0 otherwise. Therefore, we obtain

$$\sum_{j_1=0}^J \sum_{j_2=0}^J t_{j_1j_2} y_{j_1j_2}^{ki} \leq T_k^{\max}, \forall k \in \Psi, \forall i \in \gamma_k \quad (15)$$

Then, we assume that  $t_{j_1j_2} = \bar{t}_{j_1j_2} + \delta_{j_1j_2} \zeta_{j_1j_2}^{ki}$ ,  $\bar{t}_{j_1j_2}$  denotes the nominal duration,  $\delta_{j_1j_2}$  represents the deviation range, where  $\zeta_{j_1j_2}^{ki}$  belong to

$$\mathcal{W}^{ki} = \left\{ |\zeta_{j_1j_2}^{ki}| \leq 1, \forall j_1j_2; \sum_{j_1=0}^J \sum_{j_2=0}^J |\zeta_{j_1j_2}^{ki}| \leq B^{ki} \right\} \quad (16)$$

This set let all the  $\zeta_{j_1j_2}^{ki}$  to vary in the interval  $[-1, 1]$ , but it limits the number of variations to be at most  $B^{ki}$ . This set is known as budget uncertainty in Bertsimas and Sim.<sup>41</sup> We can rewrite Eq. (15) by considering the worst possible outcome as

$$\max_{\zeta_{j_1j_2}^{ki} \in \mathcal{W}} \sum_{j_1=0}^J \sum_{j_2=0}^J \delta_{j_1j_2} \zeta_{j_1j_2}^{ki} y_{j_1j_2}^{ki} \leq T_k^{\max} - \sum_{j_1=0}^J \sum_{j_2=0}^J \bar{t}_{j_1j_2} y_{j_1j_2}^{ki}, \quad (17)$$

$$\forall k \in \Psi \quad \forall i \in J_k$$

Fixing  $k \in \Psi$  and  $\forall i \in J_k$  for the sake of simplicity, the left-hand side of Eq. (17) can be written as

$$\max_{\substack{\zeta_{j_1j_2}^{ki+}, \zeta_{j_1j_2}^{ki-} \\ \zeta_{j_1j_2}^{ki+}, \zeta_{j_1j_2}^{ki-}}} \sum_{j_1=0}^J \sum_{j_2=0}^J \delta_{j_1j_2} y_{j_1j_2}^{ki} \zeta_{j_1j_2}^{ki+} - \delta_{j_1j_2} y_{j_1j_2}^{ki} \zeta_{j_1j_2}^{ki-} \quad (18)$$

$$\text{s.t. } \zeta_{j_1j_2}^{ki+} + \zeta_{j_1j_2}^{ki-} \leq 1, \forall j_1, j_2 \quad (19)$$

$$\sum_{j_1=0}^J \sum_{j_2=0}^J \zeta_{j_1j_2}^{ki+} + \zeta_{j_1j_2}^{ki-} \leq B^{ki} \quad (20)$$

$$\zeta_{j_1j_2}^{ki+}, \zeta_{j_1j_2}^{ki-} \in \mathbf{R}^+, \quad \forall j_1, j_2 \quad (21)$$

where we set  $\zeta_{j_1j_2}^{ki} = \zeta_{j_1j_2}^{ki+} - \zeta_{j_1j_2}^{ki-}$  to linearize the absolute values in Eq. (16). Now, using duality for LPs, we can write the left-hand side of Eq. (17) as

$$\min_{\lambda_{j_1 j_2}^{ki}, \mu^{ki}} B^{ki} \mu^{ki} + \sum_{j_1=0}^J \sum_{j_2=0}^J \lambda_{j_1 j_2}^{ki} \quad (22)$$

$$\text{s.t. } \lambda_{j_1 j_2}^{ki} + \mu^{ki} \geq \delta_{j_1 j_2} y_{j_1 j_2}^{ki}, \forall j_1, j_2 \quad (23)$$

$$\lambda_{j_1 j_2}^{ki} + \mu^{ki} \geq -\delta_{j_1 j_2} y_{j_1 j_2}^{ki}, \forall j_1, j_2 \quad (24)$$

$$\lambda_{j_1 j_2}^{ki} \in \mathbf{R}^+, \forall j_1, j_2, \mu^{ki} \in \mathbf{R}^+ \quad (25)$$

Therefore, Eq. (17) can be replaced by the following set of constraints

$$\begin{cases} B^{ki} \mu^{ki} + \sum_{j_1=0}^J \sum_{j_2=0}^J \lambda_{j_1 j_2}^{ki} \leq T_k^{\max} - \sum_{j_1=0}^J \sum_{j_2=0}^J \bar{t}_{j_1 j_2} y_{j_1 j_2}^{ki}, \\ \quad \forall k \in \Psi, \forall i \in J_k \\ \lambda_{j_1 j_2}^{ki} + \mu^{ki} \geq \delta_{j_1 j_2} y_{j_1 j_2}^{ki}, \forall k \in \Psi, \forall i \in J_k, \forall j_1, j_2 \in J \\ \lambda_{j_1 j_2}^{ki} + \mu^{ki} \geq -\delta_{j_1 j_2} y_{j_1 j_2}^{ki}, \forall k \in \Psi, \forall i \in J_k, \forall j_1, j_2 \in J \\ \lambda_{j_1 j_2}^{ki} \in \mathbf{R}^+, \forall j_1, j_2 \in J, \mu^{ki} \in \mathbf{R}^+, \forall k \in \Psi, \forall i \in J_k \end{cases} \quad (26)$$

Thus, a robust model for the on-orbit mission scheduling problem considering the uncertainty of refueling duration was constructed. Based on this model, further solving for the mission scheduling scheme can effectively mitigate execution risks from uncertainties, ensure timely refueling as fuel nears depletion, extend spacecraft lifespan, meet high orbit satellite reliability demands, and cut on-orbit maintenance costs.

Notably, this proposed approach exhibits its applicability not merely to GEO spacecraft. Through appropriate modifications and fine tuning, it can be effectively extended and adapted to other categories of orbital spacecraft, prominently including medium earth orbit and low earth orbit spacecraft. Despite the distinct disparities in mission characteristics and environmental factors between non-GEO orbital spacecraft and their GEO counterparts, the fundamental concept of robust mission scheduling, which takes into account uncertain duration and is employed in this research, is capable of fulfilling the OOR mission requisites of diverse orbital spacecraft.

### 3. Mission scheduling method: A hybrid harris hawks optimization

To solve the robust mission scheduling optimization problem with uncertain refueling duration, this paper proposes the HHHO algorithm based on HHO.<sup>35</sup> Unlike HHO, HHHO introduces the chaotic cubic mapping method to optimize the initial population, producing a more uniform population to enhance the quality of the generated initial population and improve the algorithm's convergence rate. Meanwhile, the crossover strategy in the genetic algorithm is combined in HHHO's exploration phase for updating the perching locations of harris hawk individuals, which further improves the algorithm's exploration ability to solve robust mission scheduling optimization problems.

#### 3.1. Initial population

Most of the existing metaheuristic algorithms search the solution space by randomly generating the initial population; however, the randomly generated initial population's quality is

primarily undesirable, affecting the algorithm's search efficiency. The cubic mapping chaos operator's optimization idea can generate a more uniform population to improve the convergence speed. Meanwhile, compared with the general logistic mapping, it produces a better uniformity of the population.<sup>42</sup> This paper uses the chaotic cubic mapping operator to generate the initial population. The formula of the chaotic cubic mapping operator is as follows:

$$\begin{cases} y(i+1) = 4y(i)^3 - 3y(i) \\ -1 < y(i) < 1, \quad y(i) \neq 0, \quad i = 0, 1, 2, \dots \end{cases} \quad (27)$$

where  $y(i)$  is the chaotic variable, the main steps are:

**Step 1.** Set  $y(0)$ . A random value in  $(-1, 1)$  is taken as the initial chaotic variable  $y(0)$ , then Eq. (27) is applied in order to obtain a vector of values  $y(i)$ . This guarantees the uniformity and diversity of the chaotic variables. Generate  $J$  elements and sort them in ascending order, using the original natural number positions to decide the target.

**Step 2.** The values of the  $y(i)$  are mapped into the decision variables. Specifically,  $y(0), y(1), \dots, y(J-1)$  are mapped into the interval  $[0, K]$  to decide the SSc to use for each target by using the linear transformation

$$u_i = l_{\min} + (l_{\max} - l_{\min}) \cdot (y(i) + 1)/2 \quad (28)$$

where  $u_i$  is the value of the variable after mapping, the parameters  $l_{\min}$  and  $l_{\max}$  respectively denote the lower and upper bounds of the admissible value range, and the mapped values are rounded up to the nearest integer. Next,  $y(J-1), y(J), \dots, y(2J-1)$  are used to define the list of targets. Since each target must be visited exactly once, Eq. (28) is unsuitable for this purpose. Instead, we use the list position of the components of the vector  $y(J-1), y(J), \dots, y(2J-1)$  into the sorted vector (in increasing order). For example, if the vector  $y(J-1), y(J), \dots, y(2J-1)$  is  $[0.93, 0.12, 0.11]$ , the sorted order is  $[0.11, 0.12, 0.93]$ , and the list of targets to visit is 3, 2, 1. Indeed, the smallest element is the one in position 3, the second smallest element is the one in position 2 and the biggest one is the one in position 1. The successive chunk, i.e.,  $y(2J-1), y(2J), \dots, y(3J-1)$  is used to compute, if the SSc has to return or not. This decision is based on 0-1 decision variables. When  $y$  is negative, the variable is set to 0; when  $y$  is positive, the variable is set to 1. Finally, the last part of the  $y$  vector is used to decide auxiliary variable  $\mu$  and  $\lambda$ , by using Eq. (28) with  $l_{\min} = 0$  and  $l_{\max} = 6000$ .

**Step 3.** Since in step 2 it is possible to generate infeasible individuals, the population is filtered and the top 80 individuals with better fitness are selected as the initial population.

A complete individual is composed of the SSc variable, the target variable, the return to the FS decision variable, and two auxiliary variables  $\mu^{ki}, \lambda_{j_1 j_2}^{ki}$  (with  $\mu^{ki}$  and  $\lambda_{j_1 j_2}^{ki}$  satisfying Eq. (26)). We call each individual  $X$ . For instance, Fig. 6 shows a mission scheduling coding model for refueling 8 targets using 4 SSc. The scheme is that SSc 1 returns to FS after refueling Target 3, then refuels Target 2 and returns FS; SSc 2 refuels Target 6, then it comes back to the FS, the it refuels Target 7 and Target 1; SSc 3 refuels target 8 and Target 4 in turn; SSc 4 only refuels Target 5.

As depicted in Fig. 6, the fuel consumption calculation for each SSc is presented as follows.

For SSc 1: the parameters in Eq. (10) are  $k = 1, J_1 = 2, |\gamma_1^1| = 1, |\gamma_2^1| = 1, \sigma_1^1(1) = 3, \text{ and } \sigma_2^1(1) = 2$ . In the 1st mission,

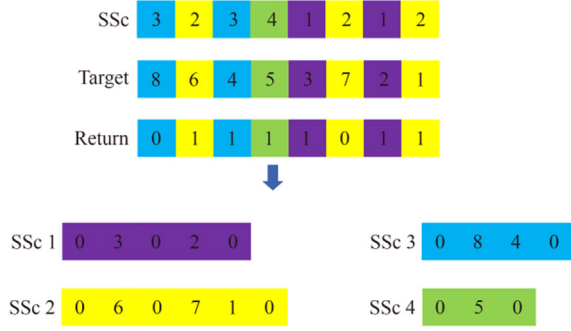


Fig. 6 Mission scheduling coding model.

define  $\delta_1^1 = \exp(\Delta v_{0\sigma_1^1(1)}/(gI_{sp}^1))$ , correspondingly,  $\Phi_1^1 = \exp((\Delta v_{0\sigma_1^1(1)} + \Delta v_{\sigma_1^1(1)0})/(gI_{sp}^1))$ . The fuel consumption is  $m_{\sigma_1^1(1)}^{(R)}\delta_1^1 + m_1^{(dry)}(\Phi_1^1 - 1)$ . The 2nd mission, define  $\delta_1^2 = \exp(\Delta v_{0\sigma_2^1(1)}/(gI_{sp}^1))$  and  $\Phi_1^2 = \exp((\Delta v_{0\sigma_2^1(1)} + \Delta v_{\sigma_2^1(1)0})/(gI_{sp}^1))$ . The fuel consumption is  $m_{\sigma_2^1(1)}^{(R)}\delta_1^2 + m_1^{(dry)}(\Phi_1^2 - 1)$ .

For SSc 2: the parameters in Eq. (10) are  $k = 2$ ,  $J_2 = 2$ ,  $|\gamma_2^1| = 1$ ,  $|\gamma_2^2| = 2$ ,  $\sigma_2^1(1) = 6$ ,  $\sigma_2^2(1) = 7$ , and  $\sigma_2^2(2) = 1$ . Similarly, when considering the 1st mission, assume that  $\delta_2^1 = \exp(\Delta v_{0\sigma_2^1(1)}/(gI_{sp}^2))$  and  $\Phi_2^1 = \exp((\Delta v_{0\sigma_2^1(1)} + \Delta v_{\sigma_2^1(1)0})/(gI_{sp}^2))$ . The fuel consumption is calculated as  $m_{\sigma_2^1(1)}^{(R)}\delta_2^1 + m_2^{(dry)}(\Phi_2^1 - 1)$ . For the 2nd mission, let  $\delta_2^2 = \exp(\Delta v_{0\sigma_2^2(1)}/(gI_{sp}^2))$ ,  $\Phi_2^2 = \exp((\Delta v_{0\sigma_2^2(1)} + \Delta v_{\sigma_2^2(1)\sigma_2^2(2)})/(gI_{sp}^2))$ , and parameter  $\Gamma_2^2 = \exp((\Delta v_{0\sigma_2^2(1)} + \Delta v_{\sigma_2^2(1)\sigma_2^2(2)} + \Delta v_{\sigma_2^2(2)0})/(gI_{sp}^2))$ . The fuel mass consumption is calculated as  $m_{\sigma_2^2(1)}^{(R)}\delta_2^2 + m_{\sigma_2^2(2)}^{(R)}\Phi_2^2 + m_2^{(dry)}(\Gamma_2^2 - 1)$ .

For SSc 3: the parameters in Eq. (10) are  $k = 3$ ,  $J_3 = 1$ ,  $|\gamma_3^1| = 2$ ,  $\sigma_3^1(1) = 8$ , and  $\sigma_3^1(2) = 4$ . The definitions are given for  $\delta_3^1 = \exp(\Delta v_{0\sigma_3^1(1)}/(gI_{sp}^3))$ ,  $\Phi_3^1 = \exp((\Delta v_{0\sigma_3^1(1)} + \Delta v_{\sigma_3^1(1)\sigma_3^1(2)})/(gI_{sp}^3))$ , and parameter  $\Gamma_3^1 = \exp((\Delta v_{0\sigma_3^1(1)} + \Delta v_{\sigma_3^1(1)\sigma_3^1(2)} + \Delta v_{\sigma_3^1(2)0})/(gI_{sp}^3))$ . The fuel consumption is then calculated as  $m_{\sigma_3^1(1)}^{(R)}\delta_3^1 + m_{\sigma_3^1(2)}^{(R)}\Phi_3^1 + m_3^{(dry)}(\Gamma_3^1 - 1)$ .

For SSc 4: with parameters  $k = 4$ ,  $J_4 = 1$ , and  $\sigma_4^1(1) = 5$ . Assume that  $\delta_4^1 = \exp(\Delta v_{0\sigma_4^1(1)}/(gI_{sp}^4))$  and  $\Phi_4^1 = \exp((\Delta v_{0\sigma_4^1(1)} + \Delta v_{\sigma_4^1(1)0})/(gI_{sp}^4))$ . The fuel consumption is calculated as  $m_{\sigma_4^1(1)}^{(R)}\delta_4^1 + m_4^{(dry)}(\Phi_4^1 - 1)$ .

The total fuel consumption in Eq. (11) is obtained by aggregating the fuel consumption of each SSc.

### 3.2. Prey's energy

The HHO algorithm simulates the natural predation process, the transition from the exploration phase to the exploitation phase is achieved by modeling the prey's energy decay process

$$E = 2 \left( 1 - \frac{t}{T_{\max}} \right) \quad (29)$$

where  $E$  is represented as the energy of the prey,  $t$  and  $T_{\max}$  are the current iteration number and the maximum iteration number. Therefore as  $t \rightarrow T_{\max}$ ,  $E \rightarrow 0$ .

### 3.3. Exploration phase

When  $|E| \geq 1$  indicates that the prey has enough energy, the exploration phase is executed and the hawk follows the prey closely. The standard HHO's exploration phase is stochastic. Nevertheless, updating the hawks position based has several limitations in our context, thus the crossover strategy in GA is used in the HHO's exploratory phase. The crossover is sequential and is shown in Fig. 7. In particular, we select a random hawk (i.e., a solution) with a probability  $q$ , we do crossover with another randomly selected individual. Otherwise, crossover is done with the optimal individual in the current population.

### 3.4. Exploitation phase

When  $|E| < 1$ , the prey energy is reduced and the hawk will adopt the exploitation phase to end the hunt (i.e., to attack the prey). In this phase we define  $r$  as a hunting success rate, where  $r$  is a random number in the range  $[-1, 1]$ . Moreover, in the following the position of the rabbit, i.e.,  $\mathbf{X}_{\text{rabbit}}$  is the best solution found so far. There are four possibilities: soft besiege, hard besiege, soft besiege with progressive rapid dives, and hard besiege with progressive rapid dives. We discuss them in the next sections.

#### 3.4.1. Soft besiege

When  $r \geq 0.5$  and  $|E| \geq 0.5$ ,  $r$  is the random value inside  $(0, 1)$ , the hawk adopts soft besiege strategy, the prey has remaining energy to avoid the hawk's predation in this hunting situation, the hawk chooses to hover over the prey and wait for the prey to exhaust its energy to hunt it. This soft besiege behavior is modeled as follows:

$$\mathbf{X}(t+1) = \Delta \mathbf{X}(t) - E|L \cdot \mathbf{X}_{\text{rabbit}}(t) - \mathbf{X}(t)| \quad (30)$$

$$\Delta \mathbf{X}(t) = \mathbf{X}_{\text{rabbit}}(t) - \mathbf{X}(t) \quad (31)$$

where  $\mathbf{X}(t+1)$  is the hawk's position vector in the next iteration,  $\Delta \mathbf{X}(t)$  is the difference between the position vector of the rabbit and the current location in iteration  $t$ . Finally,  $L = 2(1 - \rho)$  represents the random jump strength of the rabbit throughout the escaping procedure with  $\rho$  is a random

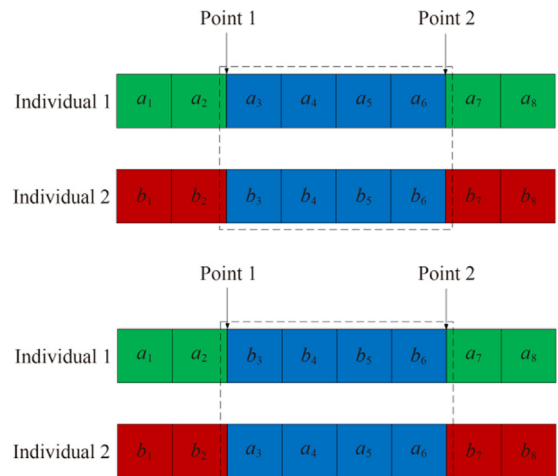


Fig. 7 Sequential crossover.

number inside (0, 1). The value of  $L$  changes randomly in each iteration to simulate the nature of rabbit motions.

### 3.4.2. Hard besiege

When  $r \geq 0.5$  and  $|E| < 0.5$ , the hawks quickly capture exhausted prey and adopt this strategy. The current positions are updated as follows:

$$X(t+1) = X_{\text{rabbit}}(t) - E|\Delta X(t)| \quad (32)$$

where  $\Delta X(t)$  is defined as in Eq. (31).

### 3.4.3. Soft besiege with progressive rapid dives

When  $r < 0.5$  and  $|E| \geq 0.5$ , the prey has enough energy to make several misleading movements to disturb the hawk's predation, and the hawk makes irregular movements to follow the prey, it will quickly hunt the prey at the right time, which is based on

$$Y_{\text{sp}} = X_{\text{rabbit}}(t) - E|L \cdot X_{\text{rabbit}}(t) - X(t)| \quad (33)$$

and

$$Z_{\text{sp}} = Y_{\text{sp}} + LF \cdot S \quad (34)$$

where  $Y_{\text{sp}}$  and  $Z_{\text{sp}}$  are the moving position,  $S$  is a random vector which components are in (0,1) and  $LF$  is the Levy flight function defined as

$$LF = 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}}, \sigma = \left( \frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{\frac{(1+\beta)}{2}}} \right)^{1/\beta} \quad (35)$$

where  $u, v$  are random values inside (0,1),  $\beta$  is a default constant set to 1.5.

Hence, the positions of hawks in the soft besiege with progressive rapid dives phase can be calculated as follows:

$$X(t+1) = \begin{cases} Y_{\text{sp}}, & \text{if } Y_{\text{sp}} \text{ better than } X(t) \\ Z_{\text{sp}}, & \text{if } Z_{\text{sp}} \text{ better than } X(t) \\ X(t), & \text{otherwise} \end{cases} \quad (36)$$

### 3.4.4. Hard besiege with progressive rapid dives

When  $r < 0.5$  and  $|E| < 0.5$ , the hawk's predation success is low even though the prey has lost the ability to escape in this behavior, the hawk increases its predation success by reducing the average distance between itself and the prey. The moving positions  $Y_{\text{hp}}$  and  $Z_{\text{hp}}$  are obtained using new rules

$$X(t+1) = \begin{cases} Y_{\text{hp}}, & \text{if } Y_{\text{hp}} \text{ better than } X(t) \\ Z_{\text{hp}}, & \text{if } Z_{\text{hp}} \text{ better than } X(t) \\ X(t), & \text{otherwise} \end{cases} \quad (37)$$

with

$$Y_{\text{hp}} = X_{\text{rabbit}}(t) - E \left| L \cdot X_{\text{rabbit}}(t) - \frac{1}{N} \sum_{i=1}^N X_i(t) \right| \quad (38)$$

$$Z_{\text{hp}} = Y_{\text{hp}} + LF \cdot S \quad (39)$$

The pseudocode of HHHO for the OOR robust scheduling optimization problem is given in Algorithm 1 and the corresponding flow chart is shown in Fig. 8.

## Algorithm 1. Pseudocode of HHHO

---

**Input:** the initial parameters  
1: The chaotic cubic mapping to generate initial population by Eqs. (27) and (28)  
2: Calculate the fitness function value and save the optimal individual and optimal solution by Eq. (11)  
3: **While**  $t < T$  **do**  
4:     **for** each individual **do**  
5:         Update escaping energy value and randomly select an individual  
6:         **if**  $|E| \geq 1$  **then**  
7:             Random  $q$   
8:             **if**  $q < 0.5$  **then**  
9:                 Crossing current individual with selected individual to update habitat location  
10:             **else**  
11:                 Crossing current individual with optimal individual to update habitat location  
12:             **end if**  
13:             **else**  
14:                 Random  $r$   
15:                 **if**  $r \geq 0.5$  and  $|E| \geq 0.5$  **then**  
16:                     The soft besiege mechanism employs Eq. (30) for individual updates  
17:                     **else if**  $r \geq 0.5$  and  $|E| < 0.5$  **then**  
18:                         The hard besiege mechanism employs Eq. (32) for individual updates  
19:                     **else if**  $r < 0.5$  and  $|E| \geq 0.5$  **then**  
20:                         Soft besiege with progressive rapid dives by Eq. (36)  
21:                     **else if**  $r < 0.5$  and  $|E| < 0.5$  **then**  
22:                         Hard besiege with progressive rapid dives by Eq. (37)  
23:                     **end if**  
24:             **end if**  
25:             **end for**  
26:     Calculate the fitness function value and update the optimal solution and optimal individual  
27: **end while**  
28: Update the optimal solution and the optimal individual  
**Output:** Global optimal solution, optimal individual, optimal value of each generation and population

---

## 4. Simulation examples

This section verifies the effectiveness of the optimization framework proposed in this paper by making three experiments. The first one verifies the performance of the HHHO optimization algorithm presented by comparing it with the GA, HHO algorithm. Instead, the second one verifies the robustness of the OOR mission scheduling strategy considering uncertain refueling duration. Finally, the effect of the initial population strategy on the algorithm's performance is investigated. All the simulations are based on 1 FS, 4 SSC, and 9 targets, in which the orbits are set as GEO circular orbits with the same orbital altitude, and the orbital inclination and RAANs of the GEO satellites in actual operation are adopted. All the experiments have been running on a PC with a 3.2 GHz CPU (AMD R7 5800H) and 16 GB RAM.

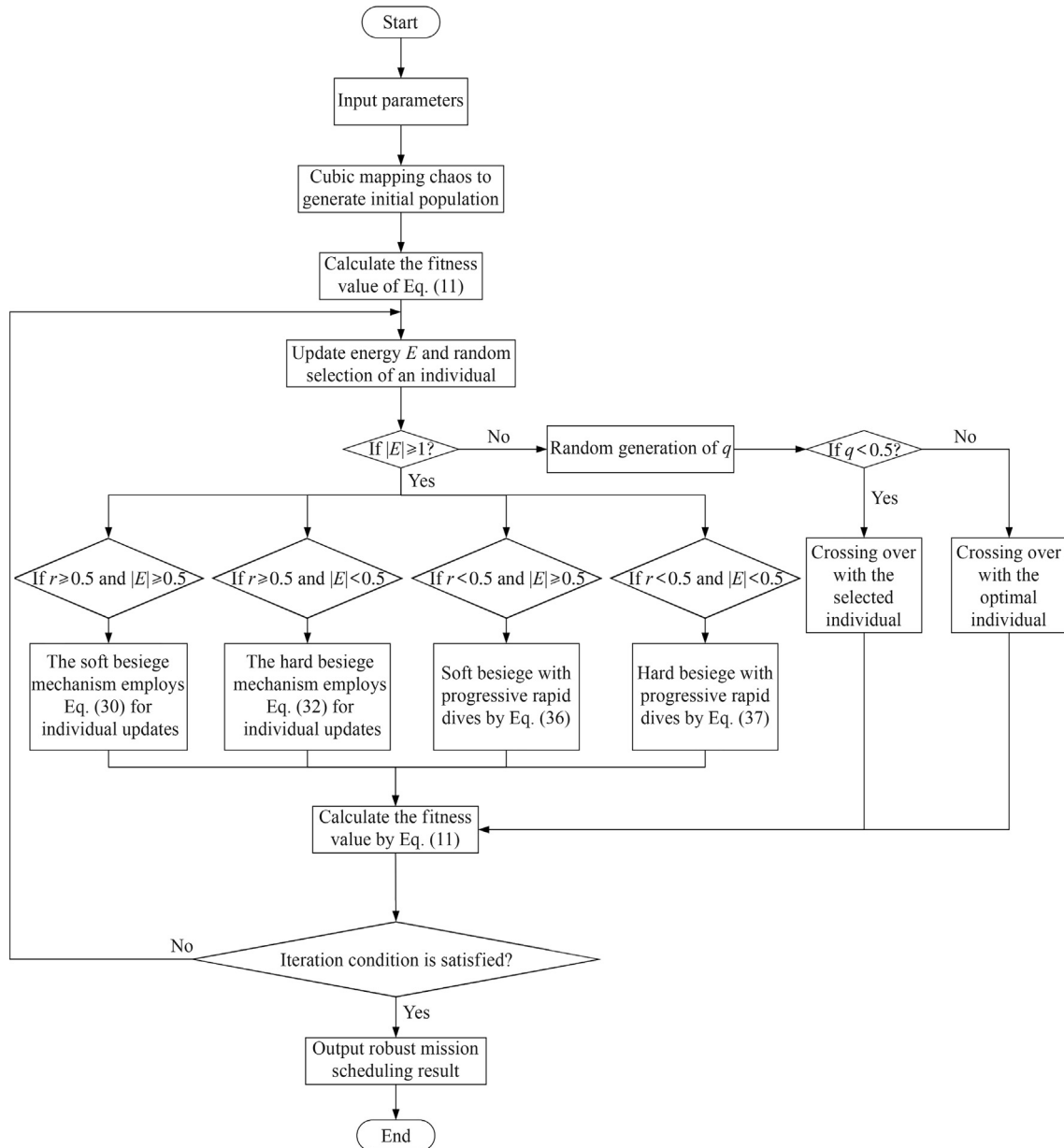


Fig. 8 HHHO' s flow chart.

Table 1 Orbital parameters of FS and SSc.

Parameter	$A$ (km)	$e$	$i$ ( $^\circ$ )	$\Omega$ ( $^\circ$ )	$\omega$ ( $^\circ$ )	$\theta_t$ ( $^\circ$ )
Value	42 164.200	0	0.029 0	117.592 2	0	10.351 7

Table 2 Parameters of SSc.

Parameter	$m_k^{\text{dry}}$ (kg)	$gI_{\text{sp}}$ (km/s)	$m_{\text{max}}$ (kg)
Vaule	500	3	3 000

#### 4.1. Comparison with other optimization algorithms

The orbital parameters of FS and SSc are given in Table 1, where  $a$  is the semi-major axes,  $e$  is the eccentricity,  $i$  represents orbital inclination,  $\Omega$  represents RAAN,  $\omega$  is the perigee argument and  $\theta_t$  is the true anomaly.

In the considered setting, we assume that the total amount of fuel at FS is unlimited,  $m_{\max}$  and  $gI_{\text{sp}}$  are maximum fuel carryover, and the engine impulse parameter, respectively, the detailed data are shown in Table 2. The fuel requirement of each target spacecraft is set to be between 200 kg and 500 kg, and the detailed data are shown in Table 3. The standard duration of OOR is set at 4 h, assuming that there are 4 targets with uncertainties in the refueling duration, namely  $B^{ti} = 4$ .

Concerning the solution method, the population size is 80, the maximum number of iterations of the algorithm is set to 500, and the maximum refueling duration deviation range is 60 min.

The solution obtained by applying the algorithm presented in Section 3 is reported in Table 4. It can be observed that all the SSc typically performs only one OOR operation during a refueling mission. By adopting a fuel-efficient refueling scheme, the SSc aims to minimize the number of consecutive OOR missions. This approach helps reduce its weight, thereby lowering the transfer cost of orbital maneuvers and achieving mission completion with minimal fuel consumption. Furthermore, the mission scheduling results comply with the constraints of OOS missions.

To illustrate the effectiveness of the proposed HHHO, we compare it against the solution obtained by GA, HHO and Large Neighborhood Search-Adaptive Genetic Algorithm (LNS-AGA).<sup>8</sup> Specifically, the best objective function computed by HHHO has a value of 3 557.1677 kg, while GA, HHO and LNS-AGA lead to solution of 3 561.289 3 kg, 3 592.053 1 kg, and 3 557.935 6 kg respectively. Moreover, we also monitor the evolution of the best solution using these different techniques in Fig. 9. From Fig. 9, it can be seen that HHHO has better convergence rate than both GA, HHO and LNS-AGA. This algorithm offers two main advantages. First, the use of chaotic cubic mapping for population initialization enhances both balance and diversity, enabling HHHO to achieve better solutions during the initial phase. Second, the

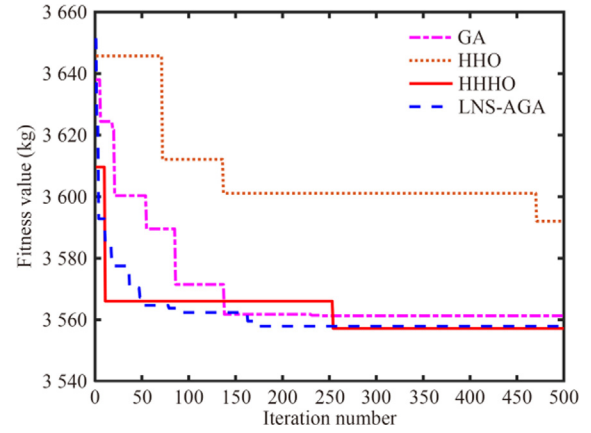


Fig. 9 Convergence curves of four algorithms.

introduction of crossover strategy improves the exploration phase, leading to higher quality solutions. In contrast, the standard HHO lacks sufficient exploration capability for the complex on orbit refueling problem due to its reliance on the average location iteration of population individuals. Furthermore, it can be observed, by combining the data in Tables 3 and 4 that the fuel consumption of orbit maneuvering accounts for the total fuel consumption of 7.23%, which is consistent with the actual engineering practice.

Fig. 10 gives the optimized fuel mass consumed by each SSc containing the fuel mass to be refueled for the target requirement. Each bar corresponds to the serial number of the SSc, and the segments are represented as the initial fuel mass carried by the spacecraft at the beginning of each mission. Similarly, Fig. 11 gives the required mission execution duration.

The time complexity serves as a measure of the time resources it consumes during execution. In addition to comparing the performance of the aforementioned algorithms, we have delved deeper into a comparison of the time complexities of each algorithm, as presented in Table 5.  $T$  is the number of iterations,  $N$  is the number of individuals, and  $D$  is the dimension of decision variables. Due to the introduction of large-neighborhood local search in LNS-AGA, its time complexity is significantly higher. Notably, the HHHO algorithm exhibits the same complexity as the GA and HHO algorithms.

Table 3 Fuel requirement of 9 targets.

Parameter	$\Delta m$ (kg)
Target 1	500
Target 2	200
Target 3	500
Target 4	300
Target 5	200
Target 6	400
Target 7	250
Target 8	450
Target 9	500

Table 4 Mission scheduling results.

Object	Scheduling scheme	$m^0$ (kg)	$T$ (d)
SSc 1	0-1-0	524.190 2	7.825 5
SSc 2	0-8-0-6-0-3-0	1 417.287 0	23.242 1
SSc 3	0-5-0-4-2-0-9-0	1 340.023 8	26.844 8
SSc 4	0-7-0	275.666 7	7.871 3

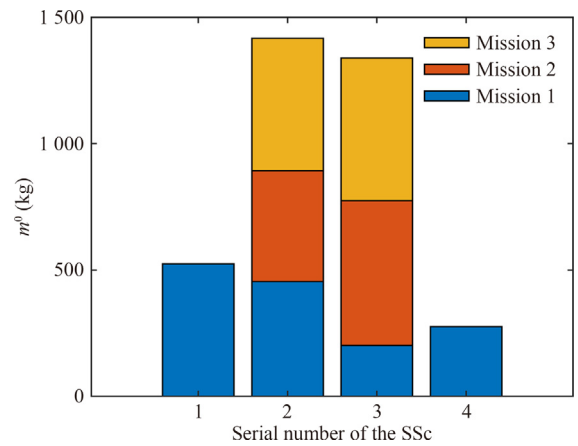
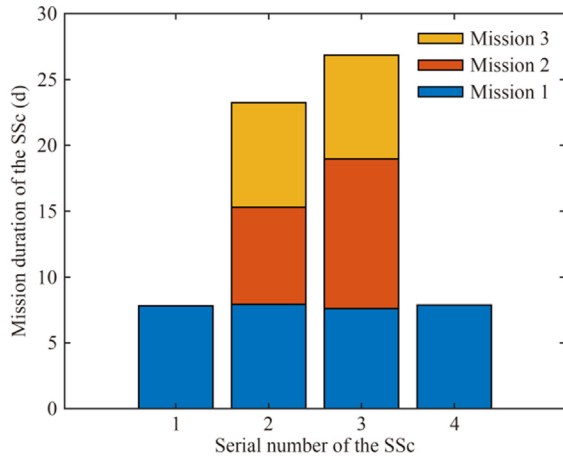


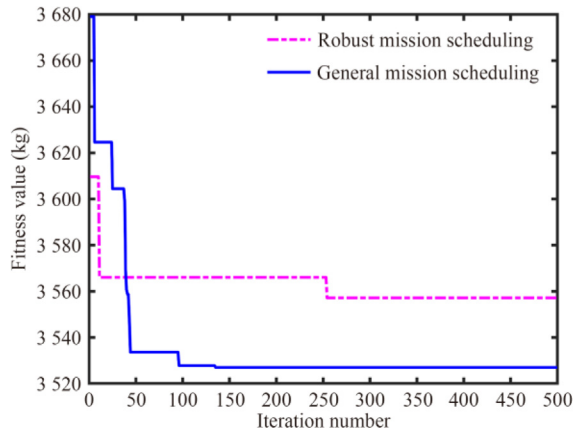
Fig. 10 Fuel mass consumed by each SSc.



**Fig. 11** Mission execution duration of each SSc.

**Table 5** Time complexity.

Algorithm	Time complexity
HHHO	$O(T \times N \times D)$
HHO	$O(T \times N \times D)$
GA	$O(T \times N \times D)$
LNS-AGA	$O(T \times N \times D^2)$



**Fig. 12** Convergence curves for solving models M1 and M2.

This clearly indicates that the improvement strategy put forward in this paper does not lead to an increase in the algorithm's complexity.

#### 4.2. Robustness analysis

This subsection compares different OOR duration on the mission scheduling schemes to show its robustness. As solution method we use the HHHO algorithm. We compare the fitness values obtained by considering and not robustness. The evolution across iterations is shown in Fig. 12.

As can be observed in Fig. 12, incorporating robustness into the model results in a gradual decline in the fitness of the best solution, ultimately stabilizing at higher values. This outcome is reasonable because accounting for robustness involves simulating adverse environmental conditions, which naturally lead to reduced performance levels. By doing so, the model ensures that solutions remain viable and effective under less favorable scenarios, highlighting the trade-off between robustness and optimal performance.

Given 100 sets of different refueling duration, we compare different numbers of targets with uncertain refueling duration, the comparative evaluation of the average values and Standard Deviations (SD) is presented in Table 6. It can be observed that the quality of the optimal scheduling solution derived from standard mission scheduling method deteriorates significantly when confronted with such uncertainties. In contrast, the robust solution that factors in uncertain refueling duration demonstrates much better performance. Additionally, as the scale of uncertainty in the target exceeds a predetermined threshold, the conservatism of the solution further enhanced.

#### 4.3. Population initialization analysis

For comparison with other methods of initializing the population, when the number of sequences is 500, the chaotic cubic map is compared with the random generation strategy, the standard logistic map and the chaotic cubic map. The results of this comparison are shown in Figs. 13–15.

It can be clearly observed from the Fig. 13 that the distribution of random generation map is not uniform, there will be obvious clustering in the final chaotic map value interval during the iteration process. The logistic mapping presented in Fig. 14 does not exhibit obvious clustering phenomena at the end of the iteration. Nevertheless, its uniformity remains less favorable compared to Fig. 15. The scatter plot of the chaotic

**Table 6** Fitness metrics: robust and standard scheduling.

Number of targets	Fitness robust scheduling		Fitness standard scheduling	
	Average (kg)	SD (kg)	Average (kg)	SD (kg)
2	4 251.968 4	0.267 2	7 108.100 97	$2.395 8 \times 10^{-11}$
3	4 252.000 1	0.582 8	7 108.803 70	2.008 8
5	4 252.994 0	6.775 1	7 109.223 80	2.919 3
7	4 254.291 5	11.232 8	7 113.649 20	17.731 1

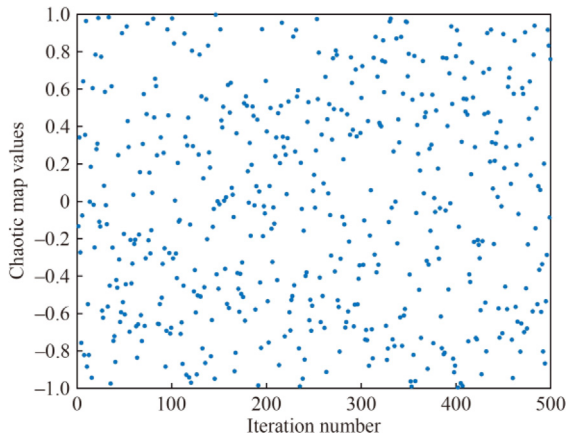


Fig. 13 Random generation map.

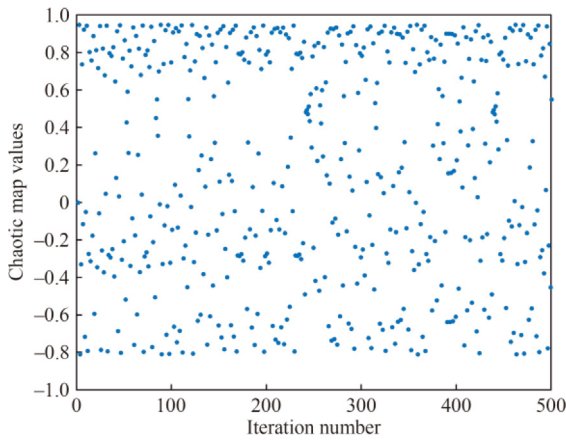


Fig. 14 Standard logistic map.

cubic map is more evenly distributed within the entire range of the ordinate (-1, 1). The number of scatter points in each value interval is relatively balanced, with no obvious changes in density.

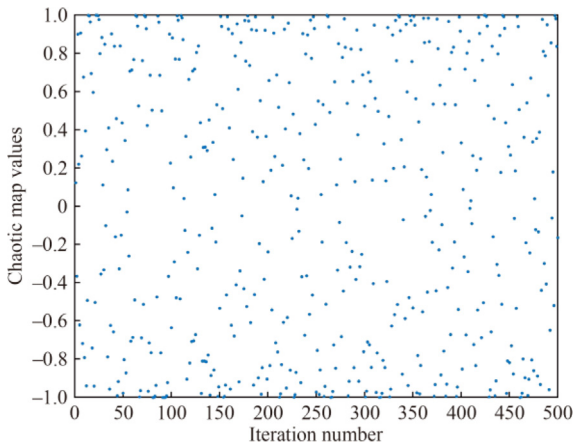


Fig. 15 Chaotic cubic map.

To further investigate the initial population on the performance of the proposed HHHO algorithm, a new mission scenario is presented, comprising 5 SSc and 20 targets, with the parameters of SSc consistent in Tables 1 and 2. The orbital inclination, RAAN, and true anomaly of the targets are randomly generated within ranges of 0–5°, 0–180°, and 0–360° respectively. Table 7 presents the fuel mass to be refueled. Similarly, set  $B^{ki} = 4$ . Specifically, this part compares three initial population generation method—random generation, standard logistic mapping, and chaotic cubic mapping—to assess the performance in solving OOR mission scheduling problem within the HHHO algorithm. The simulation results are shown in Fig. 16.

Fig. 16 shows that the Cubic + HHHO method achieves a better optimal fitness value compared to Random + HHHO and Logistic + HHHO. The chaotic cubic mapping’s superior uniformity enables the initial population it generates to dis-

Table 7 Fuel requirement of 20 targets.

Parameter	$\Delta m$ (kg)
Target 1	350
Target 2	250
Target 3	400
Target 4	350
Target 5	400
Target 6	300
Target 7	350
Target 8	400
Target 9	500
Target 10	350
Target 11	250
Target 12	400
Target 13	250
Target 14	350
Target 15	500
Target 16	400
Target 17	250
Target 18	500
Target 19	350
Target 20	400

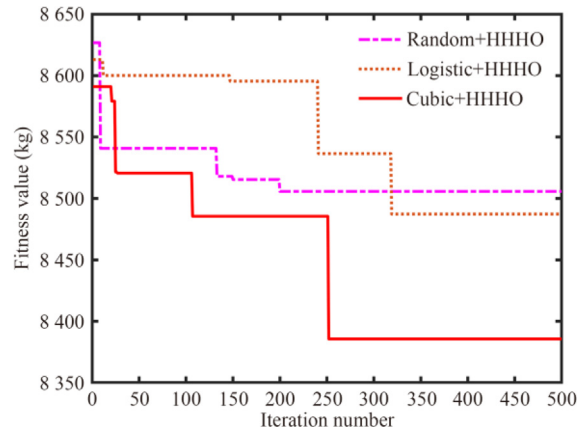


Fig. 16 Convergence curves of three initial population methods in HHHO.

**Table 8** Fitness metrics: varying initial populations.

Method	Average (kg)	SD (kg)
Random + HHHO	8 558.752 2	46.21
Logistic + HHHO	8 549.719 1	25.81
Cubic + HHHO	8 479.467 1	39.08

tribute more evenly across the search space, expanding the algorithm's exploration scope. Cubic + HHHO also has clear advantages in convergence efficiency and optimization accuracy, approaching the optimal solution more efficiently. Meanwhile, each method is tested 20 times under this scenario. Table 8 presents the statistical characteristics of fitness function values, namely the average value and SD.

As revealed in Table 8, the chaotic cubic map-based population initialization strategy consistently yields superior solutions. The results validate HHHO algorithm's efficacy in addressing OOR mission scheduling with uncertain duration.

## 5. Conclusions

This paper investigates the multiple spacecraft on orbit refueling robust mission scheduling problem, and the uncertainty of the refueling duration is considered for the first time. A robust mission scheduling model is established, and the HHHO algorithm is proposed to solve the mission scheduling scheme. Finally, the performance of the proposed scheduling strategy is simulated and analyzed. The main content and future prospects are as follows:

- (1) In order to solve the mission scheduling model with refueling duration uncertainty, the robust optimization idea is introduced to construct the robust mission scheduling model. The simulation results show that the framework is more robust in dealing the refueling mission scheduling problem with uncertain duration.
- (2) The HHHO optimization algorithm is proposed to solve the optimization problem. The algorithm proposed in this paper is compared with GA, HHO, and LNS-AGA, and the results show that it is more efficient in solving the problem. Moreover, the robustness of the proposed mission scheduling framework is verified by comparing different uncertain durations. Finally, the effect of different initialization population methods on the HHHO algorithm is also analyzed.
- (3) In future work, we will try to solve other uncertainties, such as the uncertainty of the fuel to be refueled and random spacecraft faults.

## CRediT authorship contribution statement

**Shuai YIN:** Writing – original draft. **Chuanjiang LI:** Supervision, Funding acquisition. **Edoardo FADDA:** Writing – review & editing. **Yanning GUO:** Writing – review & editing, Funding acquisition. **Guangtao RAN:** Writing – review & editing. **Paolo BRANDIMARTE:** Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

This study was co-supported by the National Natural Science Foundation of China (Nos. 62473110, 62403166), the Fundamental Research Funds for the Central Universities, China (No. 2023FRFK02043), the Natural Science Foundation of Heilongjiang Province, China (No. LH2022F023), the National Key Laboratory of Space Intelligent Control Foundation, China (No. 2023-JCJQ-LB-006-19).

## References

1. Zhai G, Zheng HM, Bin L. Attitude dynamics of spacecraft with time-varying inertia during on-orbit refueling. *J Guid Contr Dyn* 2018;**41**(8):1744–54.
2. Daneshjou K, Mohammadi-Dehabadi AA, Bakhtiari M. Mission planning for on-orbit servicing through multiple servicing satellites: A new approach. *Adv Space Res* 2017;**60**(6):1148–62.
3. Guo J, Pang ZJ, Du ZH. Optimal planning for a multi-debris active removal mission with a partial debris capture strategy. *Chin J Aeronaut* 2023;**36**(6):256–65.
4. Wei Z, Long T, Shi RH, et al. Scheduling optimization of multiple hybrid-propulsive spacecraft for geostationary space debris removal missions. *IEEE Trans Aerosp Electron Syst* 2021;**58**(3):2304–26.
5. Rao HP, Zhong R, Li PJ. Fuel-optimal deorbit scheme of space debris using tethered space-tug based on pseudospectral method. *Chin J Aeronaut* 2021;**34**(9):210–23.
6. Choi JH, Park C, Lee J. Mission planning for active removal of multiple space debris in low Earth orbit. *Adv Space Res* 2024;**73**(9):4800–12.
7. Xue ZH, Liu JG, Wu CC, et al. Review of in-space assembly technologies. *Chin J Aeronaut* 2021;**34**(11):21–47.
8. Han P, Guo YN, Li CJ, et al. Multiple GEO satellites on-orbit repairing mission planning using large neighborhood search-adaptive genetic algorithm. *Adv Space Res* 2022;**70**(2):286–302.
9. Friend RB. Orbital express program summary and mission overview. *Proceedings of sensors and systems for space applications II*. Bellingham: SPIE; 2008.
10. Li GY, Huang YY, Han W, et al. Space liquid transport experiments in Tianyuan-1. *Microgravity Sci Technol* 2022;**34**(5):100.
11. Gu YD, Gao M, Zhao GH. Space research plan of China's space station normal size. *Chin J Space Sci* 2016;**36**(5):595.
12. Breon SR, Boyle RF, Francom MB, et al. Robotic refueling mission-3—an overview. *Proceedings of the IOP conference series: Materials science and engineering*. Bristol: IOP Publishing; 2020.
13. Yan HD, Guo YN, Li CJ, et al. Neural-network-assisted optimization of a close-range multi-spacecraft rendezvous mission based on a multi-impulse maneuvering strategy. *Adv Space Res* 2023;**72**(5):1829–43.
14. Xu H, Song B, Guo YN, et al. Lambert property-based swarm search algorithm for the multiple-rendezvous trajectory optimization. *J Spacecr Rockets* 2024;**61**(3):660–73.
15. Dutta A, Arora N, Russell RP. Peer-to-peer refueling strategy using low-thrust propulsion. *J Spacecr Rockets* 2012;**49**(5):944–54.

16. Bultitude J, Burkhardt Z, Harris M, et al. Development and launch of the world's first orbital propellant tanker. *Proceedings of the 35th annual small satellite conference*. Logan: Utah State University; 2021.
17. Meng B, Huang JB, Li Z, et al. The orbit deployment strategy of OOS system for refueling near-earth orbit satellites. *Acta Astronaut* 2019;**159**:486–98.
18. Zhu XY, Zhang CX, Sun R, et al. Orbit determination for fuel station in multiple SSO spacecraft refueling considering the J2 perturbation. *Aerosp Sci Technol* 2020;**105**:105994.
19. Han P, Guo YN, Wang PY, et al. Optimal orbit design and mission scheduling for Sun-synchronous orbit on-orbit refueling system. *IEEE Trans Aerosp Electron Syst* 2023;**59**(5):4968–83.
20. Zhou Y, Yan Y, Huang X, et al. Multi-objective planning of a multiple geostationary spacecraft refuelling mission. *Eng Optim* 2017;**49**(3):531–48.
21. Li CZ, Xu B. Optimal scheduling of multiple Sun-synchronous orbit satellites refueling. *Adv Space Res* 2020;**66**(2):345–58.
22. Gu Y, Han C, Chen YH, et al. Mission replanning for multiple agile earth observation satellites based on cloud coverage forecasting. *IEEE J Sel Top Appl Earth Obs Remote Sens* 2021;**15**:594–608.
23. Peng SR, Yang X, Ding SX, et al. A dynamic rescheduling and speed management approach for high-speed trains with uncertain time-delay. *Inf Sci* 2023;**632**:201–20.
24. Feng Q, Hai XS, Sun B, et al. Resilience optimization for multi-UAV formation reconfiguration via enhanced pigeon-inspired optimization. *Chin J Aeronaut* 2022;**35**(1):110–23.
25. Xu LW, Zhang G, Qiu S, et al. Optimal multi-impulse linear rendezvous via reinforcement learning. *Space Sci Technol* 2023;**3**:47.
26. Herrmann A, Schaub H. Reinforcement learning for the agile earth-observing satellite scheduling problem. *IEEE Trans Aerosp Electron Syst* 2023;**59**(5):5235–47.
27. Wang L, Shi ZW, Dai W, et al. Two-stage stochastic planning for integrated energy systems accounting for carbon trading price uncertainty. *Int J Electr Power Energy Syst* 2022;**143**:108452.
28. Rigo CA, Seman LO, Camponogara E, et al. A nanosatellite task scheduling framework to improve mission value using fuzzy constraints. *Expert Syst Appl* 2021;**175**:114784.
29. Wang XW, Gu Y, Wu GH, et al. Robust scheduling for multiple agile Earth observation satellites under cloud coverage uncertainty. *Comput Ind Eng* 2021;**156**:107292.
30. Liang WK, Zhi H, Han P, et al. GEO satellite on-orbit refueling and debris removal hybrid mission planning under uncertainty. *Adv Space Res* 2024;**74**(5):2376–87.
31. Huang YX, Wu SF, Zeng ZK, et al. Sequential dynamic resource allocation in multi-beam satellite systems: A learning-based optimization method. *Chin J Aeronaut* 2023;**36**(6):288–301.
32. Zhang TJ, Yang YK, Wang BH, et al. Optimal scheduling for location geosynchronous satellites refueling problem. *Acta Astronaut* 2019;**163**:264–71.
33. Zhang J, Parks GT, Luo YZ, et al. Multispacecraft refueling optimization considering the J2 perturbation and window constraints. *J Guid Contr Dyn* 2013;**37**(1):111–22.
34. Xu H, Song B, Guo YN, et al. An optimization framework with improved auction-based initialization for highly constrained on-orbit servicing mission planning. *Appl Soft Comput* 2023;**149**:110983.
35. Heidari AA, Mirjalili S, Faris H, et al. Harris Hawks optimization: Algorithm and applications. *Future Gener Comput Syst* 2019;**97**:849–72.
36. Sassi M, Chelouah R. HHO-EAS: A new metaheuristic bio-inspired of the win-win hunting synergy between the two predators crow and wolf. *Artif Intell Rev* 2023;**56**(11):12439–504.
37. Akhter N, Mahmud R, Jin J, et al. Configurable Harris Hawks optimisation for application placement in space-air-ground integrated networks. *IEEE Trans Netw Serv Manag* 2024;**21**(2):1724–36.
38. Ali A, Adeel Ali Shah S, Al Shloul T, et al. Multiobjective Harris Hawks optimization-based task scheduling in cloud-fog computing. *IEEE Internet Things J* 2024;**11**(13):24334–52.
39. Zhang L, Deng YJ, Fu J, et al. A DV-Hop localization algorithm corrected based on multi-strategy sparrow algorithm in sea-surface wireless sensor networks. *Wirel Netw* 2025;**31**(2):1405–20.
40. Gharehchopogh FS, Abdollahzadeh B, Barshandeh S, et al. A multi-objective mutation-based dynamic Harris Hawks optimization for botnet detection in IoT. *Internet Things* 2023;**24**:100952.
41. Bertsimas D, Sim M. The price of robustness. *Oper Res* 2004;**52**(1):35–53.
42. Tang AD, Han T, Xu DW, et al. Chaotic multi-leader whale optimization algorithm. *J Beijing Univ Aeronaut Astronaut* 2021;**47**(7):1481–94 [Chinese].