Quantum optimization for multi-target Active Debris Removal missions

Gagliardi et al. *EPJ Quantum Technology* (2025) 12:111
https://doi.org/10.1140/epjqt/s40507-025-00409-3

EPJ Quantum Technology
a SpringerOpen Journal

RESEARCH Open Access

Quantum optimization for multi-target Active Debris Removal missions

Michele Gagliardi[1], Mattia Boggio[1*], Deborah Volpe[2] and Carlo Novara[1]

*Correspondence:
mattia.boggio@polito.it
[1]Politecnico di Torino, Corso Duca degli Abruzzi 24, Turin, 10124, Italy
Full list of author information is available at the end of the article

Abstract

The rapid accumulation of space debris in Low Earth Orbit (LEO) poses a significant challenge to the sustainability of space operations. While preventive measures limit new debris generation, they are insufficient to mitigate the growing population of defunct satellites, rocket stages, and collision fragments. Active Debris Removal (ADR) has emerged as a viable solution, which requires solving NP-hard combinatorial optimization problem similar to the Traveling Salesman Problem (TSP) to maximize mission efficiency by minimizing fuel and mission duration. This work explores the application of Quantum Annealing (QA) and Hybrid Quantum Annealing (HQA) for optimizing multi-target ADR missions. Specifically, it introduces a Quadratic Unconstrained Binary Optimization (QUBO) model for ADR, exploiting quantum computing to enhance solution efficiency. A novel quadratization method is developed to reduce computational complexity, enabling large-scale mission planning. Additionally, a novel constraint-handling strategy is proposed, integrating mission constraints into post-processing to enhance quantum solver efficiency. The proposed approach is validated using real-world satellite debris datasets and benchmarked against classical metaheuristic optimizers, including Simulated Annealing (SA), Tabu Search (TS), and Genetic Algorithms (GA). The obtained results demonstrate the advantages of quantum optimization for ADR mission planning, providing a scalable and computationally efficient solution.

Keywords: QUBO; Quantum Annealer; Hybrid Quantum Annealing; Adiabatic Quantum Computing; Quantum Optimization; Debris Removal; Quantum Computing for Space; Space technology; Quadratization; Constraints Management

1 Introduction

1.1 Motivations

Since the advent of the space age in 1957 with the launch of the first artificial satellite, Sputnik-1, human activities in orbit have led to an increasing population of space debris. Today, Low Earth Orbit (LEO) is becoming significantly congested with non-functional satellites, abandoned launch vehicle stages, and fragments resulting from collisions and explosions. The European Space Agency (ESA) estimates that over 170 million objects larger than 1 mm and approximately 34,000 exceeding 10 cm (see Fig. 1) are currently in orbit [1], posing a serious risk to operational spacecrafts, satellite constellations, and even

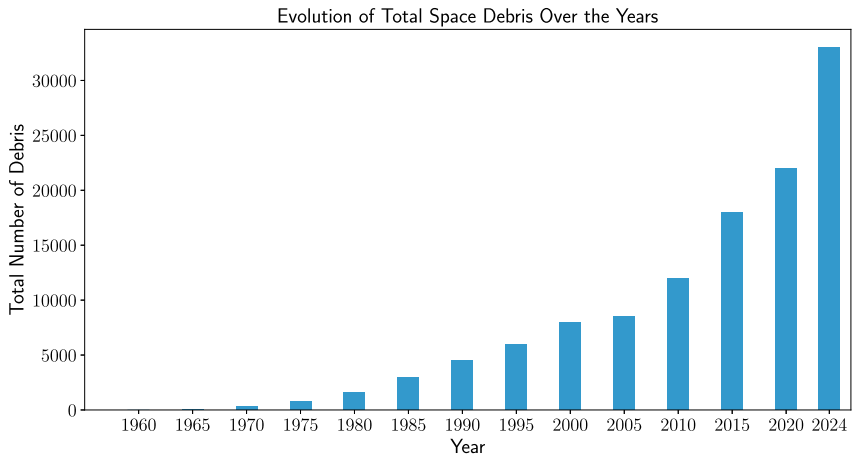


Figure 1 Evolution of the space environment in terms of number of objects greater than 10 cm since 1960

human spaceflight missions. Moreover, the rapid increase in the number of new objects launched, driven by the development of mega-constellations of satellites in LEO, such as Starlink, OneWeb, and Kuiper, has further exacerbated this issue, bringing the risk associated with space collisions to unprecedented levels. A collision in space can produce debris that can collide with other space objects, creating a cascade effect that generates more debris. This phenomenon, known as the Kessler syndrome, was first predicted by Donald J. Kessler in 1978 [2]. According to Kessler's model, the continuous chain reaction of debris collisions could make certain orbital regions unsuitable for future missions, significantly limiting access to space [3].

While international guidelines, such as the Space Debris Mitigation Guidelines published by the Inter-Agency Space Debris Coordination Committee (IADC) in 2002 [4], aim to mitigate the creation of new debris, these preventive measures will probably not be sufficient to reduce, or even stabilize the amount of debris in LEO [5]. This highlights the urgent need for more robust and effective debris mitigation strategies.

Active Debris Removal (ADR) has emerged as a promising approach to mitigate the growing debris problem by physically capturing and deorbiting large, high-risk objects [6–9]. Methodologically, this approach shares several characteristics with other space operations, including continuous rendezvous and docking [10], proximity operations [11], and round-trip servicing missions [12], where a spacecraft must sequentially visit multiple targets to perform tasks such as inspection, refueling, or repair. These missions typically involve solving complex trajectory planning problems under strict constraints on both propellant consumption and mission duration. In this context, past missions such as DARPA's Orbital Express [13], which successfully proved autonomous docking and fluid transfer between the "ASTRO" servicing spacecraft and the "NEXTSat" client satellite, have contributed significantly to the development of autonomous multi-target operation frameworks. However, ADR missions introduce additional layers of complexity. Indeed, unlike traditional servicing operations that generally involve cooperative targets with predictable behavior and pre-defined interfaces, these missions deal with uncooperative and often tumbling debris. This requires stricter safety margins, more advanced guidance and control strategies, and a greater focus on maximizing the number of debris objects removed per mission to achieve cost-effectiveness. As a result, while ADR

shares foundational principles with rendezvous and docking operations, it requires the adaptation and extension of these methodologies to address the challenges of multi-target, non-cooperative scenarios. In this regard, recent missions like Astroscale's ELSA-d, which successfully demonstrated rendezvous and magnetic capture [14, 15], and ESA's upcoming ClearSpace-1 mission (scheduled for launch in 2025) [16] represent significant advancements in ADR capabilities.

However, the efficiency and cost-effectiveness of these kinds of missions remain significant challenges. Research suggests that the removal of at least five large debris objects per year is necessary to prevent exponential growth in the debris population [17–19]. In this regard, while single-object removal missions represent milestones in the implementation of ADR, they are insufficient to address the growing scale of orbital debris. A more promising and cost-effective approach involves multi-target debris removal missions, which aim to deorbit multiple objects within a single mission, optimizing launch costs and resource usage [20, 21]. Nevertheless, designing an effective multi-target ADR mission presents a complex optimization problem. Indeed, selecting the optimal set of debris and planning the trajectory between them corresponds to an optimization problem similar to the Traveling Salesman Problem (TSP) [22], where the objective is to determine the optimal sequence of debris captures that minimizes critical performance metrics such as total mission duration and propellant consumption [23, 24]. The complexity of this optimization problem is further increased by the time-dependent dynamics of orbital mechanics, as debris objects follow distinct trajectories and precession rates. As a result, the ADR trajectory optimization problem is classified as an NP-hard combinatorial optimization problem, making the exhaustive enumeration of all possible solutions computationally feasible only when the dimension of the problem is small. Given the impracticability of obtaining exact solutions for realistic debris removal scenarios, several heuristics and metaheuristic optimization techniques have been developed and employed, over the years, to determine near-optimal solutions within reasonable computational time. Noteworthy heuristic approaches include tree search procedures such as Beam Search [25], Ant Colony Optimization [26], and A* search [27], which iteratively build solutions by adding debris targets sequentially. On the other hand, metaheuristic methods such as Simulated Annealing (SA) [28] and Genetic Algorithms (GAs) [29–31] refine a set of candidate solutions through iterative improvement. More recently, Reinforcement and Machine Learning [32, 33] has also been used for ADR mission planning, providing a data-driven approach for optimizing debris removal sequences.

However, due to the inherent complexity of the ADR optimization problem, even approximate methods may struggle to identify good near-optimal solutions or may require prohibitive computational resources as the problem size increases. Motivated by the limitations of traditional approaches, this paper explores an innovative computational paradigm, i.e., the quantum computing paradigm. In recent years, quantum computing has gained considerable popularity due to its potential to significantly improve the efficiency and quality of solutions to complex and large-scale optimization problems [34, 35]. Quantum optimization techniques can be generally divided into two main categories: gate-based quantum computing [36] and adiabatic quantum computation (AQC) [37]. The gate-based model processes information through unitary operations on qubits, while AQC solves optimization problems by evolving a quantum system from an initial ground state to the ground state of a problem-specific Hamiltonian. Within the AQC paradigm,

a key approach is Quantum Annealing (QA), which leverages the adiabatic principle for solving optimization problems [38]. The first company to develop a real quantum annealer was D-Wave in 2011 [39].

Current literature has provided substantial demonstrations of the use of quantum annealers to address real-world optimization challenges [40], including finance [41], chemistry [42], linear and nonlinear automatic control systems [43, 44], and machine learning [45, 46]. Within the aerospace domain, although the current literature still remains somewhat limited, interest in quantum annealing is growing significantly. In 2016, NASA outlined the opportunities, challenges, and applications of quantum annealing in space, with a particular focus on path planning and scheduling [47]. This interest is largely motivated by the increasing complexity of satellite mission operations, particularly in Earth Observation (EO) systems, where efficient mission planning, scheduling and data management have become critical to ensuring mission success [48–51]. In response to these challenges, recent years have seen significant progress in the application of quantum computing techniques to these domains, specifically in optimizing scheduling problems for EO missions, as demonstrated by pioneering works in [52–54]. Regarding the optimization problem associated with ADR, the literature is even more sparse, with only two research studies identified to date. In [55], the planning of the ADR mission is addressed using a combination of Artificial Neural Network and Fujitsu Digital Annealer [56], i.e., a quantum-inspired digital technology architecture, while [57] employs a quantum annealer for the atmospheric disposal via uncontrolled reentry of small debris.

1.2 Contributions

Building on this limited but promising field of research in quantum optimization for ADR missions, this paper proposes the use of a D-Wave quantum annealer, employing both Quantum Annealing (QA) and Hybrid Quantum Annealing (HQA) [58] algorithms, to optimize a mission designed for removing multiple debris objects using a single chaser spacecraft. It is important to note that HQA is a hybrid classical-quantum algorithm that integrates state-of-the-art classical optimization techniques with an intelligent allocation of the D-Wave Quantum Processing Unit (QPU) to the most suitable parts of the problem. This hybrid algorithm aims to accelerate computation and improve solution quality, particularly for large-scale problem instances where pure QA fails to yield satisfactory results.

The proposed ADR mission strategy involves transferring the debris to a designated disposal orbit (of ≈ 550 km) to ensure deorbiting within 25 years, in compliance with the IADC's Space Debris Mitigation Guidelines [4]. This approach allows the removal of objects with different sizes, mitigating the exponential growth of the debris population [19]. Upon reentry, most debris objects (typically small, unpressurized satellite fragments) are expected to fully disintegrate and burn up in the mesosphere due to intense aerodynamic heating. For high-mass or dense components, standard practices recommend a reentry survivability and casualty risk assessment in accordance with ESA and NASA guidelines, such as NASA's Object Re-entry Survival Analysis Tool (ORSAT) [59] or ESA's Debris Risk Assessment and Mitigation Analysis (DRAMA) [60]. Since our test cases focus on standard-class satellites and fragmentation debris, the projected casualty risk per object remains below the accepted threshold of 1 in 10,000. Therefore, the proposed mission profile aligns with post-deorbit safety and environmental sustainability requirements [61, 62].

In order to optimize the sequence of debris capture, an optimization framework is developed for determining the optimal sequence of debris capture that minimizes operational costs (in terms of time and propellant consumption) while maximizing strategic benefits. The resulting ADR optimization problem is formalized through an objective function that includes three key factors: (i) time of flight (TOF), to minimize total mission duration; (ii) delta-v, to reduce overall propellant consumption; and (iii) debris desirability, to prioritize the removal of high-risk objects.

To solve this optimization problem using quantum computing, we formulate it as a Quadratic Unconstrained Binary Optimization (QUBO) model, a mathematical representation particularly well-suited for implementation on quantum annealers. To the best of our knowledge, this study presents the first rigorous and comprehensive QUBO formulation specifically designed for ADR missions.

A key challenge in formulating the QUBO model for this type of problem lies in the presence of high-degree polynomial terms, primarily introduced by TOF factor. A standard approach to addressing this issue involves replacing polynomial terms with auxiliary variables and imposing constraints to ensure consistency between the new variables and the original expressions. However, this method often leads to a rapid increase in problem complexity due to the introduction of auxiliary variables, which can become computationally prohibitive. To enhance scalability for real-world applications, this paper introduces a novel method that directly reformulates the problem into a native quadratic form, making it computationally feasible while preserving its fundamental structure. To show the advantages of the proposed formulation, a detailed analysis of computational complexity has been conducted. Furthermore, to enable the application of pure QA in realistic debris removal scenarios, we propose a novel constraint-handling strategy that incorporates constraints exclusively during post-processing. This approach enhances the likelihood of obtaining optimal solutions more efficiently and consistently.

The resulting QUBO problems are implemented on a real quantum annealer, using both QA and HQA algorithms, and tested with real satellite debris datasets. The performance of these quantum algorithms is then evaluated against well-established classical metaheuristic solvers, including Simulated Annealing (SA) [63], Tabu Search (TS) [64], and Genetic Algorithms (GA) [65], showcasing the potential benefits of quantum solvers in addressing ADR optimization challenges.

In summary, the main contributions of our study are the following:

1. Formulation of a novel ADR optimization problem with a disposal orbit compliant with IADC guidelines, integrating total time of flight, delta-v and debris desirability. Translation of the formulation into a QUBO problem.
2. Development of an original Quadratization Method to simplify the formulation, ensuring computational feasibility.
3. Efficient approach for handling constraints via post-processing.
4. Implementation and benchmarking of the QUBO formulation on a real quantum annealer using a real satellite debris dataset, comparing performance against state-of-the-art classical QUBO solvers.

1.3 Outline

The paper is organized as follows. Section 2 introduces the mission and the mathematical formulation of the space debris collection problem. Section 3 presents the translation of

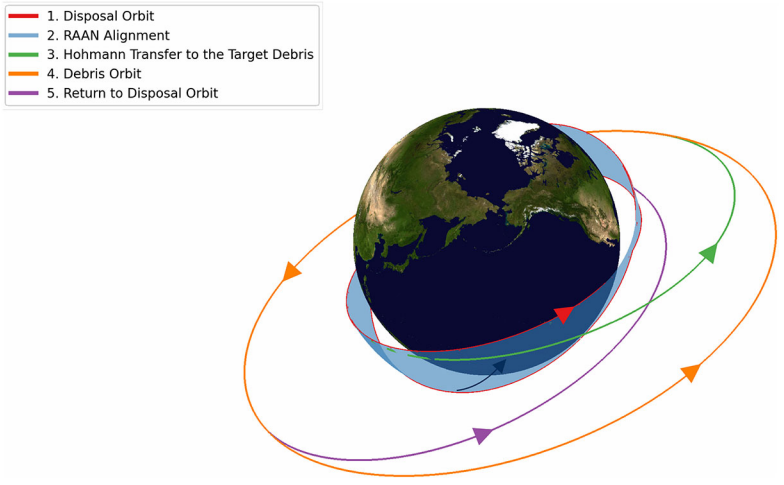


Figure 2 Mission maneuvers: depart from the disposal orbit, debris capture and return to the disposal orbit

this formulation into a QUBO problem, the analysis of its computational complexity and a novel quadratization approach. In Sect. 4, the classical and quantum solvers used for the resolution of the debris removal problem are presented. Additionally, a new method for managing constraints in the quantum annealing case is introduced. Section 5 provides a comparative analysis of the results obtained using the quantum and classical solvers, followed by a detailed discussion. Finally, the conclusions are drawn in Sect. 6.

[48, 50, 51]

2 The space debris removal problem

This article aims to address the problem of disposing and removing multiple space debris items using a single chaser spacecraft. As shown in Fig. 2, the mission involves a sequence of maneuvers for the chaser:

- departing from a designated disposal orbit;
- rendezvousing and docking with each target debris;
- transporting and releasing each debris into the disposal orbit.

This process is repeated for all scheduled debris items. The disposal orbit is designed by the IADC guidelines, ensuring complete deorbit within 25 years. Compliance with international debris mitigation standards helps reduce long-term orbital congestion.

The debris population considered in this study is derived from real satellite debris data obtained from Two-Line Element (TLE) files, ensuring accurate orbital parameter representation. Since all debris items are in LEO with low eccentricity, their orbits can be approximated as circular. To optimize fuel efficiency, the selection of the target debris is restricted to objects with orbital inclination similar to that of the chaser's disposal orbit, while variations in altitude and Right Ascension of the Ascending Node (RAAN) are possible.

For a mission designed to perform C capture operations on a set of D debris objects, the capture sequence consists of a series of orbital maneuvers, executed in a structured sequence:

1. *Departure from the Disposal Orbit:* The chaser departs from its disposal orbit, characterized by an initial altitude a_{ril} and RAAN $\Omega_{c_{in}}$.

2. *RAAN Alignment via J_2 Perturbation:* To align its RAAN with that of the target debris ($\Omega_{d(i)}$, where $i = 1, \ldots, D$), the chaser exploits the Earth's oblateness effect, specifically the J_2 gravitational perturbation. This gradual drift in RAAN is a passive strategy that offers several advantages: i) it significantly reduces propellant consumption by avoiding costly active maneuvers; ii) it lowers operational complexity, decreasing the likelihood of execution errors; iii) it exploits the predictable nature of J_2-induced RAAN variation, which enables accurate scheduling of alignment windows and enhances overall mission planning. This method is particularly beneficial in multi-target debris removal missions, where efficiency and reliability are critical.
3. *Optimal Rendezvous Timing:* Once the chaser's RAAN aligns with that of the target debris, initiate a precisely timed waiting period. This ensures that, after executing a Hohmann transfer, the chaser arrives at the target orbit at the optimal position for rendezvous.
4. *Hohmann Transfer to the Target Debris:* The chaser executes a Hohmann transfer, an energy-efficient orbital maneuver, to move from its current orbit to the target debris orbit.
5. *Capture Operation:* Upon reaching the target orbit, the chaser executes a precise relative positioning maneuver, enabling the execution of the capture operation within a predefined period.
6. *Return to Disposal Orbit:* After capture, the chaser performs another Hohmann transfer to return to the disposal orbit.
7. *Debris Release:* Upon reaching the disposal orbit, the chaser remains in position for a specified period while releasing the captured debris.
8. *Cycle Repetition:* The process is repeated for the subsequent debris in the capture sequence until all target objects are processed.

In order to optimize the sequence of debris capture by minimizing operational costs (in terms of time and propellant consumption) while maximizing strategic benefits, the optimization problem includes three terms, described in the following.

Time of flight (TOF) The Time of flight (TOF) represents the total time required to complete the capture of a single debris. For each debris object, the TOF is computed as the sum of several time components. These components account for the various orbital maneuvers and waiting periods needed to complete the capture process. The TOF for the i^{th} debris include:

1. *RAAN Alignment Time*: This time refers to the period required for the chaser's orbital plane to align with that of the target debris. Since Earth's gravitational field is not perfectly uniform, a natural drift occurs in the orientation of an orbit over time. This drift can be exploited to gradually and passively adjust the chaser's orbit until it aligns with the plane of the debris. For every capture, the chaser must adjust its orbital plane, transitioning from alignment with the actual RAAN to that of the next target. This step ensures efficient sequential capture of debris objects.
2. *Rendezvous Waiting Time*: The rendezvous waiting time is the time the chaser spacecraft must wait before starting the Hohmann transfer to meet the debris at the correct location. This time depends on the relative motion between the chaser and debris, ensuring the chaser arrives at the right phase angle for rendezvous. For multiple captures, each waiting time considers the previous ones to maintain correct

timing for each rendezvous. This sequential dependency represents the bottleneck of this approach, making the optimization problem very complex. A solution to this issue is described in Sect. 3.2.

3. *Hohmann Transfer Time* (t_H): This is the time required to perform an orbital change using a Hohmann transfer, which involves two distinct propulsion maneuvers. The first maneuver moves the chaser into an elliptical intermediate orbit, with a periapsis at the initial orbit and an apoapsis at the final orbit. The second maneuver occurs at the apoapsis (if moving to a higher orbit) or periapsis (if moving to a lower orbit) to match the debris's velocity, fully circularizing the orbit.
4. *Capture Time*: Once the chaser successfully reaches the debris, a specific period is allocated for the actual capture process. This phase includes: i) ensuring that the spacecraft is positioned and aligned with the debris, minimizing any relative motion, ii) deploying capture mechanisms (such as robotic arms or nets), iii) securing the debris, and iv) verifying that the capture is successful. The duration of this phase depends on the complexity of the capture system and the characteristics of the debris object.
5. *Release Time*: After securing the debris, it must be moved to the designated disposal orbit where it is either deorbited or stored safely. The release process includes positioning the debris in the correct disposal path and disengaging it from the chaser. Similar to the capture phase, this step requires careful execution to ensure that the debris follows the intended trajectory post-release.

Delta-v (Δv) Delta-v represents the change in velocity required to perform the orbital maneuvers. It is directly related to fuel consumption, making it a critical mission parameter.

In this study, Δv calculations are based on classical orbital mechanics. For a Hohmann transfer, the required velocity changes at both periapsis and apoapsis are computed using the vis-viva equation, which determines the velocity at any point in an orbit.

The total delta-v required for capturing a piece of debris consists of the sum of two components:

1. Δv for reaching the debris, which represents the velocity change required to transfer from the chaser's initial orbit to the debris orbit.
2. Δv for returning to the release orbit, which is the velocity change required to bring the captured debris back to a designated disposal orbit.

It is important to note that, in general, the Δv required to perform orbital maneuvers depends also on the instantaneous mass of the chaser spacecraft. However, the analysis presented in this study is based on publicly available TLE data, which do not provide information on the mass or physical characteristics of the debris. As a result, it is not possible to model the time-varying mass of the chaser with sufficient accuracy throughout the mission. Moreover, this work is primarily focused on the development of a high-level mission planning optimization framework for ADR operations. Within this context, the assumption of constant mass of the chaser represents a reasonable and practical simplification – particularly when the targets are relatively small debris fragments, such as those resulting from satellite fragmentation events.

Desirability Each debris object is assigned a desirability metric to quantify its priority for capture. While TLE files do not provide direct information on a debris object's mass

or cross-sectional area, they include the drag term and other orbital parameters. By processing these parameters through the SGP4 (Simplified General Perturbations Model 4) propagator [66], we can estimate decay times. In this study, desirability is defined as the time required for a debris object to naturally decay to the disposal orbit. Consequently, debris with longer natural decay times is given higher desirability scores, prioritizing their capture over objects with shorter decay times.

For more mathematical details about TOF and Delta-v please refer to the Appendix A.1.

The objective function for the debris collection problem is formulated as a multi-objective optimization problem, including TOF, Delta-v, and Desirability.

3 QUBO formulation for space debris collection

In order to optimize the multi-target ADR mission using a quantum annealer, the objective function is formulated in the QUBO form

$$H = \zeta^{\top} \mathbf{Q}\, \zeta\,, \tag{1}$$

where $\mathbf{Q} \in \mathbb{R}^{n\times n}$ is a real-valued matrix and $\zeta \in \{0,1\}^n \subset \mathbb{N}^{n\times 1}$ is a binary variable vector.

The first step in formulating the optimization problem as a QUBO is to define the decision variables. As in TSP [67], we introduce a matrix of binary variables x_{ij} (Fig. 3), where:

$$x_{ij} = \begin{cases} 1, & \text{if the } j^{\text{th}} \text{ debris is captured in the } i^{\text{th}} \text{ capture maneuver,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

To ensure feasible solutions, it is necessary to impose the following assignment constraints:

1. Each debris object can be captured at most once, preventing duplicate captures.
2. Each capture maneuver can select at most one debris object, ensuring that no more than one debris is collected per event.

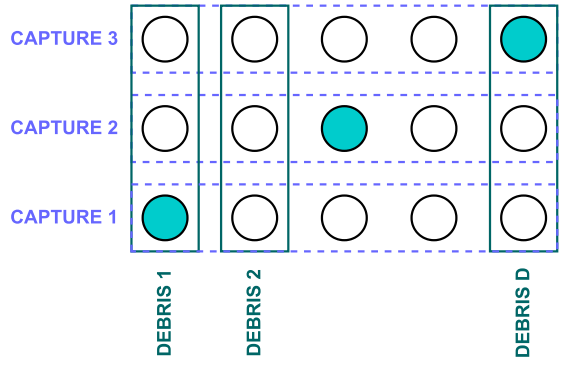


Figure 3 QUBO data structure, representing a valid solution. The highlighted circles are variables set to one, i.e. represent the debris-capture association

The first constraint is included in the QUBO formulation via the following penalty function

$$P_{\text{column}} = \sum_{k=1}^{D} \sum_{j=1}^{C} \sum_{i=1}^{j-1} x_{ik} x_{jk} \,, \tag{3}$$

which penalizes cases where a debris object is assigned to multiple capture maneuvers. Specifically, it forces to reach a final configuration where the active variables in each column are minimized, ensuring that each debris object is assigned to a single capture maneuver, as shown in Fig. 3.

The second constraint, which enforces that at most one debris object is captured per event, is implemented as follows:

$$P_{\text{row}} = \sum_{k=1}^{C} \sum_{j=1}^{D} \sum_{i=1}^{j-1} x_{ki} x_{kj} \,, \tag{4}$$

where C is the number of capture maneuvers and D the number of debris. Similarly, it ensures that the final solution contains only a single variable at one for each row, as in the configuration shown in Fig. 3.

Additionally, to control the number of selected debris, we introduce the following penalty function

$$P_{\text{matrix}} = \sum_{k=1}^{C} \sum_{j=1}^{D} x_{kj} \,. \tag{5}$$

Unlike the first two constraints, which focus on preventing rule violations, this function acts more like a regularization term in the optimization process. Indeed, by summing up all binary decision variables x_{kj}, this function aims at keeping the total number of selected debris limited. Depending on how it is weighted in the objective function, it can promote more or fewer debris captures, which controls the total number of active selections in the binary decision matrix. In other words, this function provides a mechanism to regulate the total number of selected debris objects, preventing solutions that are either too sparse or excessively dense. By adjusting its weight in the objective function, it is possible to balance the trade-off between capturing more debris objects and maintaining computational efficiency. A higher weight discourages excessive selections, favoring solutions with fewer active assignments, while a lower weight allows for more flexibility in the number of captured debris.

The optimization goal is to determine the optimal debris capture sequence by minimizing operational costs in terms of Δv and time, while maximizing the strategic benefits.

The Δv required for capturing each debris object is incorporated into the objective function as

$$f_{\Delta v} = C_V \sum_{i=1}^{C} \sum_{j=1}^{D} \Delta v_j x_{ij} \,, \tag{6}$$

where Δv_j represents the total velocity change required to perform the two Hohmann maneuvers for capturing the j^{th} debris, and C_V is a weighting parameter that balances the contribution of Δv with respect to the other terms in the objective function.

The total TOF required for the capture sequence is minimized using the following term:

$$f_{TOF} = C_T \sum_{(j_1,\dots,j_C)\in S} TOF_{(j_1,j_2,\dots,j_C)} \prod_{i=1}^{C} x_{i,j_i}\,, \tag{7}$$

where C_T is a weighting coefficient, and S represents the set of all feasible debris capture sequences that satisfy operational constraints

$$\begin{aligned} S = \{&(j_1,j_2,\dots,j_C) \in \{1,2,\dots,D\}^C \mid j_i \neq j_{i'} \text{ for all } i \neq i', \\ &\text{and } (j_1,j_2,\dots,j_C) \text{ satisfies all operational constraints}\}. \end{aligned} \tag{8}$$

It should be noted that, the term $TOF_{(j_1,j_2,\dots,j_C)}$ represents the total time of flight associated with a specific sequence of debris capture, where the indices $(j_1,j_2,\dots,j_C)$ indicate the order in which debris objects are collected. To ensure that $TOF_{(j_1,j_2,\dots,j_C)}$ contributes to the objective function only when the corresponding sequence is selected, it is multiplied by the product $\prod_{i=1}^{C} x_{i,j_i}$. This product serves as an activation mechanism:

- Each binary decision variable x_{i,j_i} is equal to 1 if debris j_i is captured in the i^{th} event and 0 otherwise.
- The product $\prod_{i=1}^{C} x_{i,j_i}$ evaluates 1 only if all x_{i,j_i} values in the sequence are equal to 1, meaning the entire sequence $(j_1,j_2,\dots,j_C)$ has been selected.

Since the function f_{TOF} is not inherently quadratic, an additional quadratization step is required, as discussed in Sect. 3.2. Moreover, as it is discussed in the following, constructing this component of the optimization problem is the most computationally expensive, as it scales factorially with the number of debris due to the need to account for all possible combinations.

The optimization also accounts for the desirability of each debris target through

$$f_D = -C_D \sum_{i=1}^{C}\sum_{j=1}^{D} des_j x_{ij}\,, \tag{9}$$

where the negative sign ensures maximization of the overall desirability score, C_D is a weighting parameter, and des_j represents the desirability score assigned to the j^{th} debris object.

The final cost function integrates all the above components:

$$\begin{aligned} f(x) &= f_{\Delta v} + f_{TOF} + f_D + \lambda_1 P_{\text{column}} + \lambda_2 P_{\text{row}} + \lambda_3 P_{\text{matrix}} = \\ &= C_V \sum_{i=1}^{C}\sum_{j=1}^{D} \Delta v_j x_{ij} + C_T \sum_{(j_1,\dots,j_C)\in S} TOF_{(j_1,j_2,\dots,j_C)} \prod_{i=1}^{C} x_{i,j_i} - C_D \sum_{i=1}^{C}\sum_{j=1}^{D} des_j x_{ij} + \\ &\quad + \lambda_1 \sum_{k=1}^{D}\sum_{j=1}^{C}\sum_{i=1}^{j-1} x_{ik}x_{jk} + \lambda_2 \sum_{k=1}^{C}\sum_{j=1}^{D}\sum_{i=1}^{j-1} x_{ki}x_{kj} + \lambda_3 \sum_{k=1}^{C}\sum_{j=1}^{D} x_{kj}\,, \end{aligned} \tag{10}$$

where λ_1, λ_2, and λ_3 are penalty weights enforcing feasibility constraints. The weighting factors C_V, C_T and C_D are used to balance the trade-off between the three competing terms – delta-v, TOF, and debris desirability – within the multi-objective cost function. To guarantee that each component contributes comparably to the overall objective, all these three terms are normalized. This normalization brings their numerical ranges to a consistent scale, preventing any single term from dominating the optimization due to unit or magnitude disparities. It also allows the use of weighting factors of similar order without introducing undesirable scaling effects. It should be noted that in this study, for the sake of simplicity and neutrality, all weights –i.e., λ_1, λ_2, λ_3, C_V, C_T and C_D – are assumed to be equal. However, the proposed framework is designed to be inherently flexible and modular, allowing users to adjust the weighting coefficients based on specific mission requirements or operational priorities.

3.1 Computational complexity

This section analyzes the computational complexity of the objective function, focusing on how the solution space scales with the number of debris objects D and capture maneuvers C, identifying computational bottlenecks, and exploring potential mitigation strategies. The objective function consists of linear terms Δv and *des*, and a high-degree polynomial term TOF, which introduces exponential growth in the solution space, significantly impacting computational feasibility.

Considering the function f_{TOF} in Equation (7), we can observe that when evaluating a single sequence $(j_1, j_2, \ldots, j_C)$, the term $\prod_{i=1}^{C} x_{i,j_i}$ requires $\mathcal{O}(C)$ operations. This is because the binary nature of x_{ij} means that each multiplication is trivial (i.e., it is a simple check for 0 or 1), and thus the computational cost for evaluating this product is minimal. However, when considering all possible admissible sequences S in the optimization problem, the complexity increases dramatically. In particular, the number of admissible sequences $|S|$ grows combinatorially with the number of debris objects D and capture maneuvers C as follows:

$$|S| = \frac{D!}{(D-C)!} \approx \mathcal{O}(D^C). \tag{11}$$

Therefore, since for each admissible sequence we also need to evaluate the product $\prod_{i=1}^{C} x_{i,j_i}$, the full TOF term results in an overall complexity of $\mathcal{O}(C \cdot D^C)$.

In contrast, the linear terms, $f_{\Delta v}$ and f_D have a significantly lower computational cost of $\mathcal{O}(C \cdot D)$, making their contribution to the total computational complexity comparatively negligible.

Then, as D increases, the number of admissible sequences grows as $\mathcal{O}(D^C)$, leading to a rapid combinatorial explosion, even for moderate values of D and C. For example, with $D = 20$ and $C = 5$, the number of sequences is $\mathcal{O}(3.2 \times 10^6)$, while for $C = 6$, it exceeds 64 million. This exponential growth severely limits the feasibility of exact approaches.

3.2 Quadratization method

The debris capture optimization problem presents a key challenge due to the presence of higher-order terms in the cost function, particularly in relation to TOF optimization. The standard approach to address this challenge involves replacing polynomial terms with auxiliary variables and enforcing constraints to maintain consistency between the new variables and the original polynomials, as shown in the following example.

Example 1 Let's consider three binary variables x, y, and z and the following polynomial expression:

$$xyz\,.$$

In order to write it as a second-order function, it is necessary to create a variable $k = xy$, substitute it in the function and impose the constraint $xy = k$, which can be written as $xy - 2(x + y)k + 3k$. Therefore, the cost function becomes:

$$kz + \lambda(xy - 2(x + y)k + 3k)\,.$$

Specifically, the reduction algorithm scans each polynomial term, leaving quadratic terms unchanged, while higher-degree terms ($D > 2$) are iteratively decomposed by introducing auxiliary variables to replace products of two variables, enforcing their equivalence through penalty terms in the objective function, and repeating the process until all terms are reduced to quadratic form. However, despite being well-established, this method has certain limitations. First, the rapid growth in problem complexity due to the introduction of auxiliary variables can become excessive. Second, the constraints imposed to maintain consistency between the original higher-order problem and the QUBO formulation may be violated if an appropriate penalty weight cannot be determined. In this case, discrepancies may occur between the optimal solutions of the original problem and the reduced one.

Considering the limitation of the standard method, we propose an alternative method, which simplifies the computation of TOF by approximating the waiting time term t_W using the synodic period for reducing the problem's degree. The synodic period is the time required for two orbiting bodies — in this case, the chaser and the debris — to realign in the same relative configuration. Operationally, it defines the worst-case scenario: if the chaser misses an optimal departure opportunity, it must wait for a duration equal to the synodic period before achieving the ideal alignment for a new rendezvous. This leads to the following approximation:

$$t_{W(0,1,\ldots,j)} \approx T_{\text{synodic},j}\,, \tag{12}$$

where $T_{\text{synodic},j}$ depends only on the j^{th} debris (i.e., the variable x_{ij} associated with the current event) rather than on the entire capture sequence. With this approach, the TOF contribution becomes linear, making the overall problem natively quadratic. While this approximation leads to an overestimation of the waiting time, its impact on solution optimality is minimal, as the dominant contribution to TOF typically comes from the RAAN Alignment time. Only in specific cases, such as multiple coplanar operations where several debris share the same RAAN but differ in altitude, could the approximation effect be more significant—though such scenarios are highly unlikely.

Therefore, the QUBO model obtained exploiting this method is equal to

$$\begin{aligned} f(x) &= f_{\Delta v} + f_{TOF} + f_D + \lambda_1 P_{\text{column}} + \lambda_2 P_{\text{row}} + \lambda_3 P_{\text{matrix}} = \\ &= C_V \sum_{i=1}^{C}\sum_{j=1}^{D} \Delta v_j x_{ij} + C_T\left(\sum_{j=1}^{D}\sum_{k=1,k\neq j}^{D} TOF'_{\text{f,jk}} x_{1,j} x_{2,k} + \sum_{i=2}^{C-1}\sum_{j=1}^{D}\sum_{k=1,k\neq j}^{D} TOF'_{jk} x_{ij} x_{i+1,k}\right) \end{aligned}$$

$$- C_D \sum_{i=1}^{C} \sum_{j=1}^{D} des_j x_{ij} + \lambda_1 \sum_{k=1}^{D} \sum_{j=1}^{C} \sum_{i=1}^{j-1} x_{ik} x_{jk}$$
$$+ \lambda_2 \sum_{k=1}^{C} \sum_{j=1}^{D} \sum_{i=1}^{j-1} x_{ki} x_{kj} + \lambda_3 \sum_{k=1}^{C} \sum_{j=1}^{D} x_{kj}\,, \tag{13}$$

where $TOF'_{\mathrm{f,jk}}$ is the TOF for the first two capture maneuver. This depends only from the synodic time, RAAN time for the transfer from the disposal orbit to the j^{th} debris and from the j^{th} to the k^{th} and the Hohmann time. Similarly, TOF'_{jk} is the TOF for the other captures and is composed of the synodic time, RAAN time for the transfer from the j^{th} to the k^{th} and the Hohmann time. More details are available in Appendix A.2.

The new methodology significantly reduces computational complexity avoiding the employment of auxiliary variables, and generating a more compact QUBO model while preserving solution accuracy through the synodic period approximation, which keeps the optimal solutions as RAAN time remains unchanged. Moreover, its scalability allows for a quadratic formulation regardless of the number of captures, reducing the problem's complexity from exponential $\mathcal{O}(D^C)$ in the traditional approach to polynomial $\mathcal{O}(C \cdot D^2)$, enabling QUBO-based optimization for large-scale debris capture scenarios.

4 Solvers and QA constraints management method

This section presents the solvers exploited in this article to solve debris collection problem and introduces a novel method to manage constraints for quantum annealer.

4.1 Classical solvers

The classical metaheuristic algorithms selected as a reference in this work are *Tabu Search* (*TS*) [68, 69], *Genetic Algorithm* (*GA*) [70] and *Simulated Annealing* (*SA*) [71], which are well-established methods for efficiently exploring the solution space of QUBO formulations.

The TS algorithm, as formulated by Palubeckis in [69] for QUBO problems, employs a short-term memory list (tabu tenure) to avoid recently explored solutions and escape local minima. It combines this with a local search strategy to refine solutions and approach near-optimal results.

A GA is a metaheuristic optimization method inspired by natural selection and evolution. It iteratively evolves a population of candidate solutions via selection, crossover, and mutation. Starting with a random population evaluated by a fitness function, the best individuals recombine through crossover, and mutation introduces variation to keep diversity. This process continues until a stopping criterion — like a set number of generations or convergence — is reached.

SA is inspired by metallurgical annealing, where heating and gradual cooling give a stable, low-energy state. It applies this principle to an optimization problem by evaluating random neighboring solutions. If the energy decreases, the new solution is accepted. Otherwise, it may still be accepted based on a temperature-dependent probability. High temperatures favor exploration, while a cooling schedule gradually lowers the temperature, guiding the system toward an optimal or near-optimal solution.

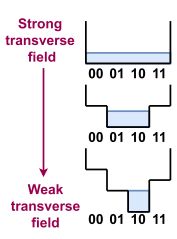
4.2 Quantum annealers

There are two primary approaches to exploiting quantum computers for optimization: the quantum circuit model [36] and *Adiabatic Quantum Computation* (*AQC*) [37]. The former executes a sequence of unitary gate operations on quantum bits (qubits), culminating in a measurement that extracts the solution. On the other hand, AQC evolves an initial quantum state adiabatically from the ground state of a simple Hamiltonian to the ground state of a final Hamiltonian that encodes the optimization problem. This work focuses on Quantum Annealing paradigm [38, 72, 73], which is a heuristic implementation of AQC, exploiting the adiabatic principle but operating in a non-ideal regime where thermal effects and hardware constraints influence the evolution.

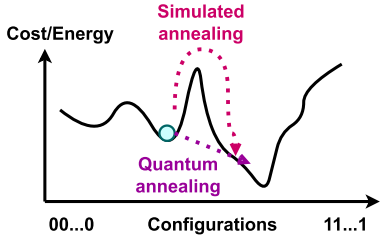
Quantum Annealers (*QAs*) are specialized quantum devices designed for efficiently solving QUBO problems, offering potential computational speedups over classical methods [74, 75]. A QA operates as a network of n interacting qubits, physically implemented using superconducting loops [73, 76]. Unlike classical bits, which assume discrete states, qubits can exist in superposition, enabling quantum parallelism. Additionally, quantum tunneling allows the system to bypass energy barriers, reducing the probability of being trapped in local minima (see Fig. 4b). Together, these properties enhance the QA's capability to solve optimization problems efficiently.

The quantum annealing evolution is composed of the steps described below. These will be commented on using the spin evolution illustrated in Fig. 4c and the commonly used hydraulic model metaphor shown in Fig. 4a.

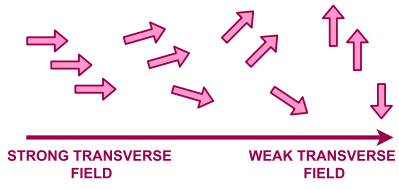
1. *Initialization:* The system is prepared in a superposition state, representing all possible solutions, where each qubit is an equal-weight superposition of $|0\rangle$ and $|1\rangle$.



(a) Quantum annealer system evolution explained through an *hydraulic model*.



(b) Quantum tunneling *vs.* thermal climbing. The first represents the exploration mechanism of quantum annealing, while the second is that of simulated annealing.



(c) Spins evolution during annealing. When a strong transverse field is applied, spins are in a superposition state. Reducing the transverse field, spins assume a spin-down or spin-up state to reach the lowest energy configuration.

Figure 4 Overview of quantum annealing

This corresponds to the ground state of the initial Hamiltonian, H_I. This can be seen as the spins aligned on the x-axis of Fig. 4c or the flat bottom of the tank in the hydraulic model usually exploited for explaining QA system evolution of Fig. 4a.

2. *Adiabatic Evolution:* The Hamiltonian evolves continuously from H_I to the final Hamiltonian H_F, which encodes the optimization problem. This evolution follows an adiabatic process governed by $H_{QA} = \varphi(t)H_I + (1 - \varphi(t))H_F$, where $\varphi(t)$ transitions from 1 to 0 over the annealing time T_A. Quantum tunneling helps the system avoid local minima during this process. This process is graphically illustrated in Fig. 4a as the gradual reshaping of the tank's bottom to represent the objective function, causing the water to flow to the lowest points. Alternatively, Fig. 4c depicts it as the spins gradually reorienting to reach the minimum energy configuration.
3. *Measurement:* At the end of the annealing process, the final spin configuration σ^* is measured, yielding the solution $\zeta^* = (\sigma^* + I)/2$.
4. *Post-processing:* Due to noise and hardware imperfections, additional classical optimization techniques (e.g., simulated annealing or tabu search) may refine the solution.
5. *Repetitions:* The process is repeated multiple times to improve the probability of obtaining the optimal solution.

The effectiveness of QA relies on the Adiabatic Theorem, which ensures that the system remains in the ground state if the Hamiltonian evolves slowly enough. The minimum required annealing time T_A satisfies

$$T_A \geq \frac{1}{\min_{t\in[0,T_A]} \Delta E(t)^2}, \tag{14}$$

where ΔE is the energy gap between the two lowest states. If this condition is not met, the system may transition to higher-energy states, leading to suboptimal solutions. In practice, a trade-off exists between computation time and solution quality. While QA offers a potential quadratic speedup over classical methods [74, 75], practical constraints such as thermal fluctuations and background noise limit the feasibility of arbitrarily long annealing times.

Modern QAs, such as those developed by D-Wave [77], use superconducting qubits arranged in a lattice structure, where qubits interact via programmable couplers. These couplers mediate interactions by implementing the Ising model's — directly associated with the QUBO formulation [78] — spin-spin coupling term

$$H_F = \sum_i h_i\sigma_i + \sum_{i,j} J_{ij}\sigma_i\sigma_j, \tag{15}$$

where h_i represents individual qubit biases, and J_{ij} defines coupler strengths. Couplers play a crucial role in shaping the energy landscape: overly strong couplings can suppress quantum tunneling and restrict exploration, while weak couplings risk inconsistent constraint enforcement. Proper tuning of these parameters is critical to achieving high-quality solutions.

Despite its potential, QA faces practical challenges, including qubit coherence limitations, noise, and embedding constraints. Effective parameter tuning, particularly of coupler strengths and constraint weights, remains crucial for ensuring high-quality solutions.

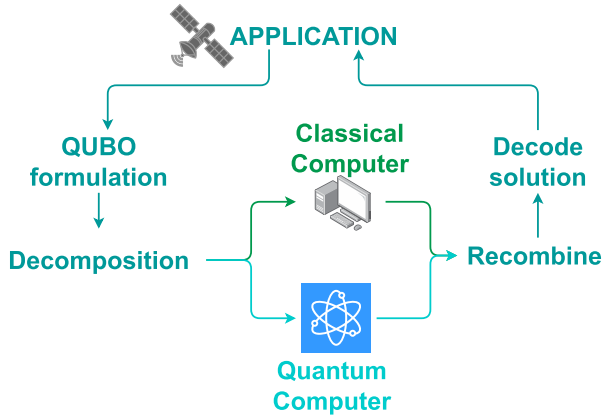


Figure 5 Hybrid Quantum Solver

Future advancements in hardware and hybrid techniques are expected to further enhance QA's practical applicability for real-world optimization problems.

In this context, a promising alternative is *Hybrid Quantum Annealing* (*HQA*), which combines heuristic classical solvers with quantum optimization. As shown in Fig. 5, HQA decomposes complex optimization problems into smaller subproblems, which are selectively solved on the quantum processing unit (QPU), while classical components handle pre-processing, and solution refinement [79]. This hybrid strategy effectively mitigates hardware limitations — such as sparse qubit connectivity and the overhead of minor embedding — and improves the overall scalability, robustness, and applicability of quantum annealing in practical contexts.

4.3 Innovative constraints management method for quantum annealer

Experimental results revealed a critical issue with traditional *one-hot* quadratic constraints in QA: the annealer frequently converged to high-energy configurations that violated the constraints, even while following a descending energy trajectory. A natural approach to enforcing constraints in QA is to increase the penalty coefficients. However, excessively large λ values negatively impact annealing by producing high coupler strengths, modifying the energy landscape, and creating abrupt local variations, as discussed in [80]. This behavior leads the annealer into suboptimal minima and compresses the solution space, preventing proper exploration.

To address this issue, we proposed to adopt a new strategy. Specifically, to insert in the cost function only the P_{matrix} constraint and tune the `chain_strength` parameter for constraint enforcement. The omitted constraints are instead applied in a post-processing step, where solution feasibility is checked and invalid sequences (e.g., with duplicate debris assignments) are discarded. This hybrid constraint-handling strategy preserves solution validity while minimizing the QUBO size and complexity, improving performance on current-generation quantum annealers. Therefore, this approach provides two main advantages:

- *Reduced Coupler Count:* Lowering quadratic terms significantly decreases the number of required couplers, improving scalability for large debris capture problems.
- *Enhanced Exploration of Valid Solutions:* By adjusting the `chain_strength` parameter, we prevent the annealer from prematurely converging to the trivial

solution ($x_{ij} = 0$) and promote a gradual transition in which the number of "1"s gradually decreases. This enables the identification, through a post-annealing analysis of the samples, of configurations that satisfy the constraints and have relatively low energy, facilitating the selection of the optimal solution.

In QA, logical variables are mapped onto qubit chains. If the chain strength is too low, *chain-breaking* occurs, leading to invalid solutions. On the other hand, if the chain strength is set too high, it creates strong connections between the qubits in the chain. This leads to a highly constrained system with reduced flexibility, thereby limiting its ability to explore different configurations.

Therefore, we decided to adopt different constraint management strategies for the different solvers considered:

- *Quantum Annealer (QA):* Using only P_{matrix} reduces coupler count and simplifies embedding, improving scalability for large debris capture problems. Chain strength tuning enables progressive exploration of valid solutions, ensuring that the lowest-energy configuration is selected.
- *Hybrid Quantum Annealing/Simulated Annealing/Tabu Search/Genetic Algorithm (HQA/SA/TS/GA):* These methods can handle full constraint enforcement (including P_{row} and P_{column}) due to their tolerance for more complex QUBO models. However, excessive penalty terms still increase the coupler count, potentially limiting scalability.

5 Results

In this section, the obtained results are reported and analyzed, describing the mission scenario considered for the debris capture. The problems were solved using different solvers provided by D-Wave. In particular: i) the *Dwave Advantage_system4.1* for the QA ii) the *D-Wave Leap Hybrid Sampler* for the HQA, iii) the *D-Wave SimulatedAnnealingSampler* for the SA, and iv) the *D-Wave TabuSampler* for the TS. Moreover, the *pymoo* implementation was considered for GA.

The employed D-Wave Advantage systems are characterized by approximately 5000 physical qubits arranged according to the Pegasus topology. This architecture offers significantly higher connectivity than earlier Chimera-based generations, with each qubit connected to up to 15 others. However, fully connected or densely structured problems still require the use of minor embedding techniques, which map logical variables onto chains of physical qubits. This increases the total qubit usage and introduces potential sources of error, such as chain breaks and noise. In our case, problem instances involving more than ~ 100 logical variables could not be directly embedded onto the hardware. Due to these limitations, we also evaluated the potential of the quantum annealer using D-Wave's hybrid solver, which combines quantum and classical solvers to handle larger problem sizes.

5.1 Mission scenario and setting

For our study, we relied on TLE data corresponding to the debris generated by the collision between the Iridium 33 and Cosmos 2251 satellites. This significant event, which occurred in February 2009, produced a huge amount of debris in LEO and has since served as a critical benchmark for debris modeling and mitigation efforts. The TLE files were collected during the time frame of January 19–20 2025 from the NORAD elements repository hosted on CelesTrak [81].

A key aspect of our analysis was the selection of an appropriate disposal orbit. We chose a release altitude of approximately 550 km since debris objects at this altitude, or lower, are generally expected to naturally deorbit within 25 years, in compliance with the relevant guidelines. Once the debris information was extracted and the release orbit defined, we applied a filtering process to exclude any debris already situated below the designated release altitude.

Following the initial filtering, we computed the mean orbital inclination of the remaining debris and adopted it as the working orbit inclination for the chaser spacecraft. To maintain operational efficiency, we further limited our selection to debris objects within a narrow inclination band, removing those deviating by more than 5° from the mean. This step was taken to ensure that the chaser spacecraft remains in an orbit capable of accessing the majority of targets without the need for costly inclination changes. Since the debris in our study largely originated from the same satellite breakup event, this filtering step minimally affected the overall dataset. In cases where the number of debris objects D was fewer than the total remaining after filtering, we retained only the D debris with inclinations closest to the mean for further analysis.

Furthermore, in each scenario considered, the initial position and orbit of the chaser—defined by its argument of latitude and RAAN—were generated randomly. This approach allowed us to account for a broader range of possible mission configurations and prevented the optimization process from converging to the same solution across all cases. By varying these orbital parameters, we ensured a more diverse and robust assessment of feasible mission trajectories.

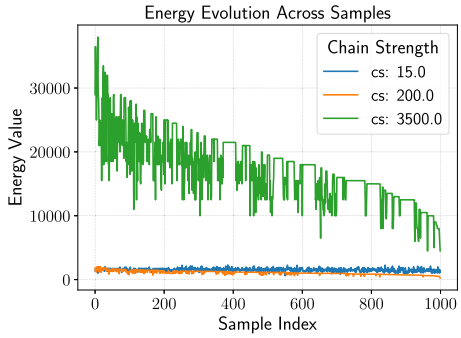
To standardize mission parameters and maintain consistency across scenarios, we imposed constant arbitrary durations for key mission phases. Specifically, the capture phase was set to a duration of 2 hours, while the release phase was set to 1 hour. These standard timeframes allow for uniform analysis of mission performance and operational feasibility across different scenarios.

Regarding the parameters of the cost function, we used equal weighting factors, i.e., $C_V = C_T = C_D = 1$, to maintain a neutral trade-off among the competing objectives. This choice avoids favoring any particular criterion (delta-v, time of flight, or debris desirability) and provides an unbiased assessment of the optimization framework.

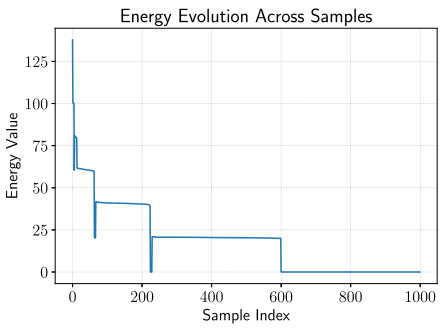
5.2 Quantum annealing constraint management

The energy evolutions of QA reported in Fig. 6 clearly demonstrate a significant improvement when using only the P_{matrix} penalty term. Initially, we evaluated the full set of constraints — specifically, P_{row} and P_{column} — across various chain strength settings in a scenario with 2 capture maneuvers and 50 debris objects from the Iridium 33 dataset (Fig. 6a). Although the best energy value was achieved with a chain strength of 200, this configuration still yielded an invalid solution. Lowering the chain strength further resulted in highly oscillatory energy trends due to frequent chain breaks, while excessively high chain strength limits the solution space exploration; in this case, the energy decreased too slowly to reach any valid configurations. In essence, despite optimal tuning efforts, the conventional approach failed to enable effective exploration of the valid solution space, making it impossible to identify feasible — let alone optimal — solutions.

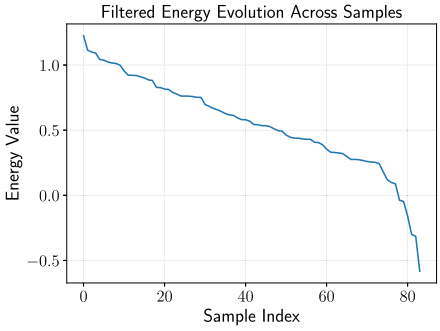
In contrast, using only the P_{matrix} penalty term yields a distinctive *staircase* search behavior, as shown in Fig. 6b. In this approach, the annealer initially explores configurations



(a) Evolution variation with the chain strenght



(b) Evolution considering only the P_{matrix} constraints



(c) Evolution considering only the P_{matrix} constraints and filtering solutions not satisfying the P_{row} and P_{column} constraints

Figure 6 QA energy evolutions, considering 2 capture maneuvers and 50 debris objects from the Iridium 33 dataset

with all the variables active (i.e., variables set to 1), then transitions sequentially to configurations with all active variables except one, and so on, until it reaches the trivial all-zeros solution. This stepwise exploration allows for a straightforward post-processing strategy: filtering the results to retain only those solutions with the correct number of active variables. Specifically, the solutions that satisfy the matrix constraints are maintained and that with lowest energy is selected (Fig. 6c).

This methodology allows for a more efficient exploration of the valid solution space compared to the traditional full-constraint approach. Even in relatively simple scenarios with few debris objects and capture maneuvers, the conventional method with full constraints consistently fails to lead any valid solutions when used with QA. On the other hand, the optimized P_{matrix}-based approach not only increases the likelihood of finding valid solutions but also enables more consistent identification of optimal solutions.

5.3 Solver comparison and scalability

The scalability of our approach is evidenced by the experimental results obtained for both the Iridium and Cosmos datasets (Table 1 and 2).

For the Iridium dataset, with 2 capture maneuvers, the QA solver consistently delivered competitive energy values while maintaining remarkably low computational times — even as the number of debris objects (D) increased from 15 to 114. For example, the QA

Table 1 Results obtained from the Iridium 33 datasets changing the number of capture maneuvers and debris objects

C	D	Energy value					Time (s)				
		QA	HQA	SA	GA	TS	QA	HQA	SA	GA	TS
2	15	**−0.56**	**−0.56**	**−0.56**	**−0.56**	**−0.56**	**0.12**	2.99	0.75	6.57	21.00
	20	**−0.65**	**−0.65**	**−0.65**	**−0.65**	**−0.65**	**0.21**	2.99	1.10	8.61	21.00
	25	**−0.58**	**−0.58**	**−0.58**	**−0.58**	**−0.58**	**0.14**	3.00	1.90	10.72	21.01
	50	**−0.58**	**−0.58**	**−0.58**	**−0.58**	**−0.58**	**0.19**	2.99	4.95	30.84	21.02
	75	−0.92	**−0.93**	**−0.93**	**−0.93**	**−0.93**	**0.22**	2.99	7.80	63.30	21.04
	100	**−1.37**	**−1.37**	**−1.37**	**−1.37**	**−1.37**	**0.25**	2.99	19.33	192.86	21.25
	114	−1.36	**−1.48**	**−1.48**	**−1.48**	**−1.48**	**0.27**	2.99	22.33	282.11	21.56
3	15	–	**−0.67**	**−0.67**	−0.29	**−0.67**	–	**2.99**	4.79	27.77	21.05
	20	–	**−0.64**	**−0.64**	0.07	−0.38	–	**2.99**	11.64	60.51	23.52
	25	–	**−0.94**	**−0.94**	0.60	−0.62	–	**2.99**	21.80	112.33	30.57
	50	–	−0.43	**−0.47**	0.75	–	–	**7.02**	91.57	2381.04	535.75
	75	–	−0.80	**−0.89**	–	–	–	**19.98**	400.57	5357.58	1253.38
	100	–	−0.66	**−0.88**	–	–	–	**43.95**	808.34	6632.49	3531.35
	114	–	**−1.21**	−1.06	–	–	–	**69.37**	1320.92	8696.00	27,803.17
4	25	–	−0.07	**−0.21**	–	–	–	**8.43**	235.79	–	–
	50	–	0.69	**0.26**	–	–	–	**73.86**	3704.42	–	–

Table 2 Results obtained from the Cosmos 2251 datasets changing the number of capture maneuvers and debris objects

C	D	Energy value					Time (s)				
		QA	HQA	SA	GA	TS	QA	HQA	SA	GA	TS
2	25	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**0.20**	2.98	1.25	22	21
	50	**−0.81**	**−0.81**	**−0.81**	**−0.81**	**−0.81**	**0.16**	2.99	2.58	60	21
	75	−0.93	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**0.22**	2.99	4.75	130	21
	150	−0.88	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**0.27**	3.00	34	480	22

solver's runtime increased only marginally from approximately 0.12 seconds to 0.27 seconds, whereas classical solvers such as SA and GA experienced substantial increases, with GA's time escalating from about 6.6 seconds to over 280 seconds.

In scenarios with 3 capture maneuvers, although QA results were not available, the HQA approach demonstrated stable energy performance with relatively low computational times compared to SA, GA, and TS. It is possible to notice that as the problem size grew, classical methods exhibited a dramatic rise in computation time: GA and TS, for instance, required several orders of magnitude more time (e.g., GA reaching over 2.300 seconds for $D = 50$ and TS exceeding 25.000 seconds for $D = 114$) compared to HQA.

The unavailability of QA results for 3 or more capture maneuvers stems from the nature of the problem formulation. Since the degree of the problem is equal to the number of capture maneuvers, cases with more than 2 captures required a quadratization process, which in turn introduced a substantial number of auxiliary variables. Experimental results revealed that this excessive number of auxiliary variables hindered the QA solver's ability to effectively explore the valid solution space. Specifically, the newly introduced constraints from quadratization significantly restricted QA's search dynamics, leading to inaccurate and inefficient exploration of feasible solutions. As a result, the solver frequently produced constraint-violating solutions, making it impractical for cases involving more than 2 capture maneuvers.

However, this limitation is effectively addressed by the new quadratization method introduced in this work. By transforming the problem into a native quadratic form without excessive auxiliary variables, the new approach enables QA to handle problems with a higher number of capture maneuvers while preserving solution quality and computational efficiency. With this improved quadratization technique, QA can now be applied to scenarios involving larger numbers of captures without encountering the inefficiencies and constraints observed in the previous formulation.

Similar trends are observed in the results from Cosmos dataset (Table 2). For 2 capture maneuvers, QA maintained a low runtime (ranging between 0.16 and 0.27 seconds) while delivering energy values equivalent to HQA, SA, GA, and TS. In contrast, classical solvers again showed significant increases in computation time as the number of debris objects increased — for example, GA required nearly 480 seconds for $D = 150$.

Table 3 reports the figures of merit for evaluating the mission performance obtained by the best QUBO solver in terms of final energy and solving time for the larger problem solved. It is possible to observe that the best solver is mainly a quantum-based solver.

Figures 7 and 8 show the execution time required to solve the debris collection problem using the Iridium 33 datasets, as a function of the number of debris objects. The results highlight the rapid growth in execution time for classical solvers as the problem size increases, while QA and HQA exhibit nearly constant performance, proving their scalability advantage.

Table 3 The best results obtained for the larger problems of the Iridium 33 and Cosmos 2251 dataset in terms of energy and time

Dataset	CxD	Best energy value				Best time			
		Solver	TOF (days)	Δv	des	Solver	TOF (days)	Δv	des
Iridium	2x114	HQA	52.62	458.90	1.71	QA	52.62	442.86	1.55
	3x114	HQA	555.40	495.09	1.72	HQA	555.40	495.09	1.72
	4x50	SA	910.22	492.39	1.05	HQA	994.53	375.99	0.089
Cosmos	2x150	HQA	11.40	218.21	0.45	QA	105.48	175.69	0.25

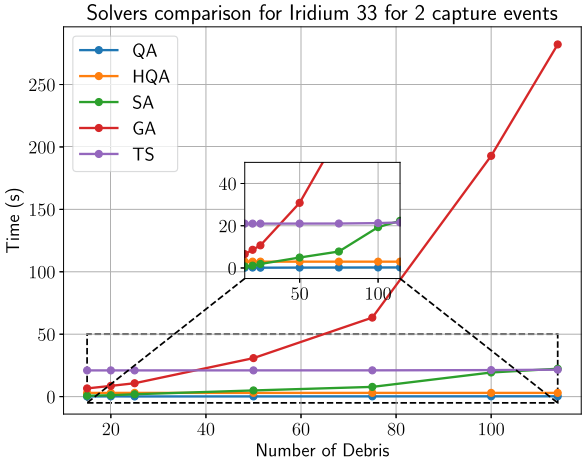


Figure 7 Time considering 2 capture maneuvers from Iridium 33 varying the number of debris

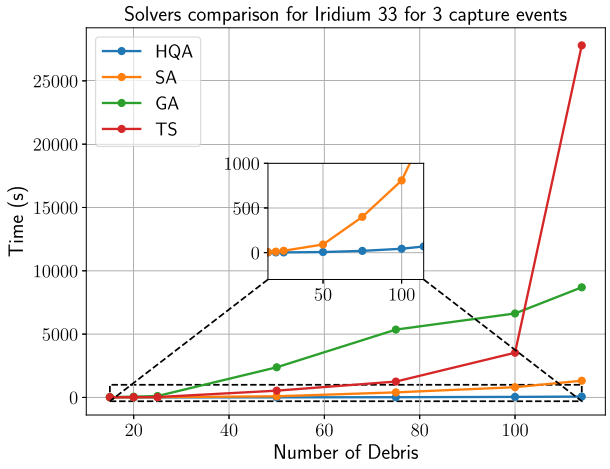
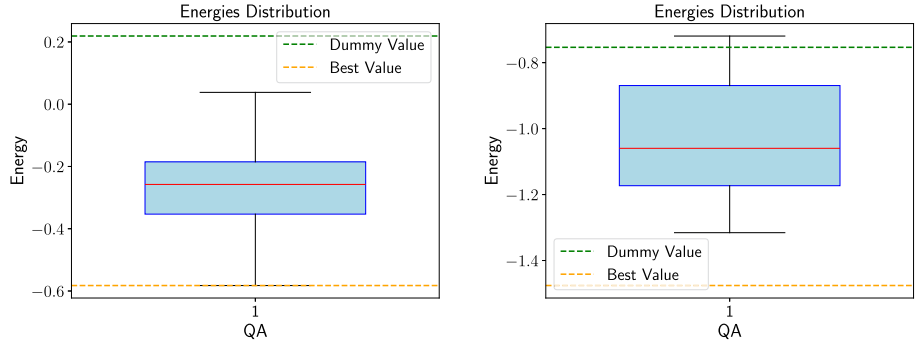


Figure 8 Time considering 3 capture maneuvers from Iridium 33 varying the number of debris



(a) Final energy variation across 100 QA runs, considering 2 captures involving 50 debris from the Cosmos dataset

(b) Final energy variation across 100 QA runs, considering 2 captures involving 114 debris from the Iridium dataset

Figure 9 Final energy variation across 100 of QA and HQA

To evaluate the robustness of the QA, the distribution of final energies obtained over *100 independent runs* has been analyzed for two benchmark scenarios: the *2-capture, 50-debris* problem from the *Cosmos* dataset, and the *2-capture, 114-debris* case from *Iridium*. In particular, we evaluated both the *average final energy* — indicated by the *red line* — and the *variation across runs*, providing valuable insights into the solver's *stability and consistency*. As shown in Figs. 9a and 9b, this analysis enables to evaluate not only the typical performance of the solver, but also its *run-to-run variability*, which is crucial for evaluating its *reliability in practical applications*.

Overall, these results demonstrate that the quantum annealing-based approaches, particularly QA and HQA, scale better with increasing problem complexity. The minimal increase in computational time with larger datasets, combined with competitive energy performance, suggests that quantum methods offer a promising pathway for tackling large-scale active debris removal mission planning.

Table 4 Results obtained from the Iridium 33 datasets changing the number of capture maneuvers and debris objects with the new quadratization method

C	D	Energy value				Time (s)			
		HQA_quad	SA_quad	GA_quad	TS_quad	HQA_quad	SA_quad	GA_quad	TS_quad
3	15	**−0.67**	**−0.67**	**−0.67**	**−0.67**	3.00	**1.67**	9.25	21.00
	20	**−0.64**	**−0.64**	**−0.64**	**−0.64**	3.00	**2.63**	12.77	21.02
	25	**−0.94**	**−0.94**	**−0.94**	**−0.94**	**2.99**	3.53	17.82	21.01
	50	**−0.59**	**−0.59**	**−0.59**	**−0.59**	**3.00**	9.00	52.05	21.03
	75	**−1.25**	**−1.25**	**−1.25**	**−1.25**	**2.99**	23.88	108.51	21.07
	100	**−1.38**	**−1.38**	**−1.38**	**−1.38**	**3.00**	28.09	204.14	21.15
	114	**−1.55**	**−1.55**	**−1.55**	**−1.55**	**2.99**	34.15	269.05	21.53
4	25	**−0.76**	**−0.76**	**−0.76**	**−0.76**	**2.99**	3.29	21.44	21.01
	50	**−0.69**	**−0.69**	−0.66	**−0.69**	**3.00**	9.32	73.68	21.05
5	114	**−4.79**	−4.75	−4.24	−4.77	**2.99**	34.91	486.12	25.99
10	114	−8.69	−8.76	−6.92	**−9.41**	**3.25**	93.59	1107.29	49.09

Table 5 Results obtained from the Cosmos 2251 dataset with the new quadratization method

C	D	Energy value				Time (s)			
		HQA_quad	SA_quad	GA_quad	TS_quad	HQA_quad	SA_quad	GA_quad	TS_quad
5	200	−4.37	−4.46	−4.47	**−4.52**	**3.00**	111.16	1489.61	42.40

Table 6 The best results obtained for the larger problems of the Iridium 33 and Cosmos 2251 datasets in terms of energy and time with the new quadratization method

Dataset	CxD	Best energy value				Best time			
		Solver	TOF (days)	Δv	des	Solver	TOF (days)	Δv	des
Iridium	3x114	HQA_quad	554.02	687.38	2.60	HQA_quad	554.02	687.38	2.60
	4x50	HQA_quad	253.60	358.91	1.01	HQA_quad	253.60	358.91	1.01
	5x114	HQA_quad	880.43	1126.35	3.86	HQA_quad	880.43	1126.35	3.86
	10x114	TS_quad	375.38	1737.30	5.10	HQA_quad	1755.26	1833.76	5.15
Cosmos	5x200	TS_quad	123.52	532.14	1.84	HQA_quad	121.86	591.29	1.85

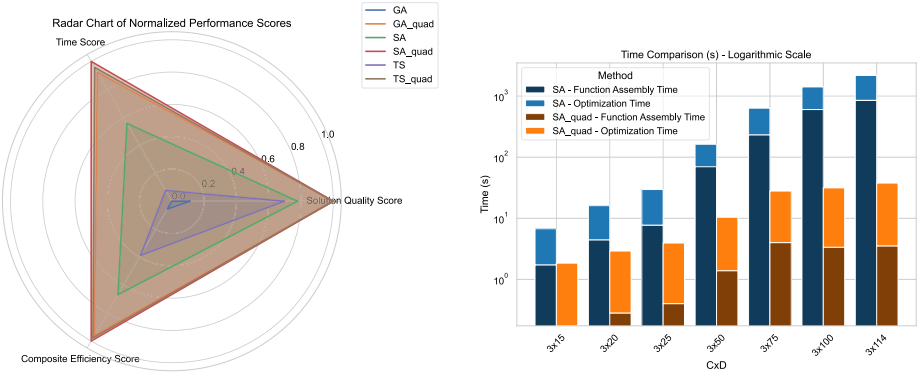
5.4 Advantage of the new quadratization method

The introduction of our novel quadratization technique offers substantial advantages in both solution quality and computational efficiency. As reported in Tables 4, 5, 6, and 7 and in Figs. 10, the new method significantly reduces the complexity of the objective function by transforming high-degree polynomial constraints into a native quadratic form without introducing auxiliary variables.

A key aspect of this improvement is illustrated in Table 4. For scenarios involving 3 and 4 capture maneuvers, the reported energy values are not directly derived from the new function but instead obtained by evaluating the old function on the optimal solutions found using the new quadratization approach. This allows for a direct comparison between the two methodologies. However, for cases with 5 and 10 capture maneuvers, we report the energy values directly from the new method. This is because computing the energy using the old method becomes prohibitively time-consuming, and no corresponding results were available for direct comparison. The advantage of the new quadratization method is intuitively reported in terms of time, solution quality and composited efficiency in Fig. 10a. The radar chart shows three normalized performance scores, each normalized in the range $[0, 1]$, where 1 represents the best performance. The scores are:

Table 7 Assembly time with the old and the new quadratization method

Dataset	C	D	Function assembly time (s)	
			Old method	New method
Iridium	3	15	0.51	0.02
		20	1.34	0.04
		25	2.67	0.07
		50	44.18	0.21
		75	146.72	0.52
		100	188.59	0.95
		114	278.65	1.18
	4	25	321.30	0.08
		50	5949.25	0.31
	5	114	–	2.24
	10	114	–	4.6079
	10	114	–	4.61
Cosmos	5	200	–	6.90



(a) Average normalized performance score of the Iridium 33 tests, considering three captures

(b) Assembly and optimization time

Figure 10 Time, energies and scores considering 3 capture events collecting debris from Iridium 33 datasets with and without the new quadratization methods proposed

- *Solution Quality Score:* This metric measures the closeness between the energy of the obtained solution and the best-known energy — score one. The worst-case reference — score zero —, used for normalization, is defined as the energy of the trivial solution, i.e., collecting the debris in sequential order.
- *Time Score:* This score evaluates execution time performance, where higher values indicate better efficiency. The fastest execution time observed for a given file defines the minimum value of the range, while the slowest execution time defines the maximum.
- *Composite Efficiency Score:* This score provides an overall measure of performance by balancing both solution quality and execution time. It is derived from a combination of two normalized scores: one for solution quality and one for execution time, each contributing equally. The highest efficiency score corresponds to the best trade-off between these two aspects, while lower scores indicate either poorer solution quality or longer execution times.

One of the most relevant advantages of the new quadratization technique is the drastic reduction in function assembly times, i.e. the time required for creating the QUBO. As evidenced in Table 7 and in Fig. 10b, the new approach reduces these times by several orders of magnitude. For instance, in the Iridium dataset with 3 capture maneuvers, the function assembly time decreased from 0.5127 seconds (old method) to just 0.0241 seconds (new method) for 15 debris objects. Similarly, for 114 debris objects, the time was reduced from 278.65 seconds to just 1.1763 seconds. For 4 capture maneuvers, the improvements are even more pronounced: the assembly time dropped from 321.30 seconds to 0.0846 seconds for 25 debris objects and from 5949.25 seconds to just 0.3078 seconds for 50 debris objects.

Beyond the significant reductions in computational time, the new quadratization method also provides meaningful improvements in the energy values obtained by the solvers. Although the optimal solutions remain consistent between the two methods, the new approach's intrinsic quadratic formulation — avoiding the overhead introduced by auxiliary variables —dramatically reduces the problem's complexity. This streamlined approach allows solvers to explore a significantly less convoluted energy landscape, leading to better energy values and, consequently, higher-quality solutions.

These dual benefits — reduced computational charge and improved solution quality — are particularly critical in extreme scenarios. With the new quadratization method, cases involving 10 capture maneuvers with 114 debris objects or even 5 capture maneuvers with 200 debris objects become computationally tractable. In contrast, these scenarios would have been infeasible with the old method due to excessive function assembly times and high problem complexity.

Furthermore, to evaluate the robustness of the QA, the distribution of final energies obtained over *100 independent runs* has been analyzed for the *5-capture, 114-debris* case from *Iridium*. In particular, we evaluated both the *average final energy* — indicated by the *red line* — and the *variation across runs*, proving solver *stability and consistency*. As shown in Fig. 11, this analysis allows to evaluate not only the typical performance of the solver, but also its *run-to-run variability*, which is crucial for proving its *reliability in practical applications*.

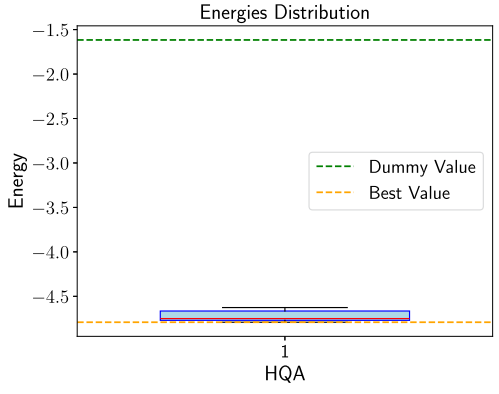


Figure 11 Final energy variation across 100 HQA runs, considering 5 captures involving 114 debris from the Cosmos dataset, considering synodic time

By substantially reducing both computational overhead and problem difficulty, the new quadratization technique enables solvers to efficiently identify better solutions. This advancement provides a promising pathway for scaling ADR mission planning to larger and more complex datasets, facilitating the development of real-world applications in space debris management.

The validity of the proposed method was evaluated by comparing the final solutions obtained with and without the application of the synodic time approximation, using the *Spearman's rank correlation coefficient* (ρ) as the evaluation metric. This coefficient is a standard measure of the strength and direction of a monotonic relationship between two ranked variables, taking the value +1 for a perfect positive correlation, −1 for a perfect negative correlation, and 0 when no monotonic relationship exists. For example, in the 3-capture, 50-debris problem, *a Spearman coefficient of* $\rho = 0.97$ was obtained, indicating a strong agreement between the two solution sets and thus supporting the validity of the proposed approximation.

5.5 Discussion

The results of this work highlight the potential of quantum optimization techniques — in particular, QA and HQA — in addressing the complex combinatorial challenges associated with *multi-target ADR missions*. By formulating the ADR problem as a *QUBO* model, we proved that quantum solvers can efficiently explore the solution space, reducing computational complexity compared to classical metaheuristic methods such as SA, GA, and TS.

The key findings of this work are:

- *Computational Efficiency and Scalability:* Quantum-based methods, particularly HQA, maintained stable performance and outperformed classical solvers in terms of computational time for large-scale ADR scenarios. As problem size increased, classical methods exhibited *exponential growth in computational time*, whereas quantum solvers scaled more efficiently. For example, in the 5-capture, 114-debris problem from the Iridium dataset, HQA achieves an 8× speedup over the fastest classical solver (TS), with a Relative Gap to Best Known Solution (RGBKS) of 0.4%. In the more complex 10-capture, 114-debris Iridium instance, HQA is 15× faster than TS, but at the expense of a slightly degraded solution quality, with an RGBKS of −8%. On the other hand, in the 2-capture, 150-debris problem from the Cosmos dataset, QA achieves a substantial 81× speedup, at the expense of a −12% RGBKS, while HQA provides a more balanced result with a 7× acceleration and a final solution that matches the optimal one.
- *New Quadratization Method:* The new quadratization approach significantly *reduces function assembly times and problem complexity* by transforming high-degree polynomial constraints into a native quadratic form without introducing auxiliary variables. This enables the application of QUBO-based solvers to larger ADR instances that were previously infeasible.
- *Constraint Handling Strategy:* Our modified constraint management approach for QA improves the probability of obtaining valid solutions while mitigating common issues related to energy landscape distortion.
- *Competitive Energy Solutions:* Across multiple test scenarios, quantum solvers lead to *near-optimal solutions*, proving their potential for real-world ADR mission planning.

These outcomes establish *quantum computing as a feasible tool for orbital debris management*, offering a pathway toward more efficient and scalable ADR mission planning.

6 Conclusions

The article proposes a new method for using QA and HQA to optimize multi-target ADR missions. We demonstrated how quantum computing approaches may be used to solve difficult combinatorial issues more efficiently — i.e., faster and with lower energy — than conventional classical heuristics by reformulating the ADR trajectory optimization problem as a QUBO model. The introduction of a quadratization technique, which enables the conversion of high-degree polynomial constraints into a quadratic form without the need for auxiliary variables and improves scalability, is a key contribution of this work.

By comparing quantum solvers to traditional optimization methods like SA, TS, and GA, the efficiency of quantum solvers in handling the ADR mission planning is evaluated. The results show the benefits of quantum optimization for ADR mission planning, offering a computationally efficient and scalable solution.

Additionally, we presented a different approach to handling the constraints, evaluating them after the post-processing stage instead of directly in the QUBO formulation. This approach preserved the computational benefits of quantum solvers while exhibiting significant improvements in solution feasibility.

Even though these are encouraging results, there are still several challenges to be addressed. The quality of the solution is impacted by the limitations of the current quantum annealing hardware, which include qubit connectivity, noise, and embedding constraints. To further improve quantum optimization efficiency for space applications, future research should investigate quantum-inspired solvers and error mitigation approaches.

Finally, this study provides a scalable method for optimizing ADR procedures and proves the promise of quantum computing in space mission planning. This research represents one of the first applications of quantum computing to orbital debris management, contributing to the advancement of sustainable space operations.

Appendix

A.1 Metrics

Time of Flight (TOF) The Time of Flight (TOF) represents the total duration required to capture a debris. For the i^{th} debris in the sequence, the TOF is given by

$$\text{TOF}_{(0,1,\dots,i)} = t_{R(i-1,i)} + t_{W(0,1,\dots,i)} + 2 \cdot t_{H(i)} + t_{\text{cat}} + t_{\text{ril}}, \quad \forall i = 1, 2, \dots, C, \tag{16}$$

where the quantities appearing in the equation are defined as follows.

- *RAAN Alignment Time* t_R: This is the time required for the chaser's RAAN to align with that of the i^{th} debris object by exploiting Earth's J_2 gravitational perturbation. The RAAN rate is given by:

$$\dot{\Omega} = -\frac{3}{2} J_2 \left(\frac{R_E}{a(1-e^2)} \right)^2 n \cos(i), \tag{17}$$

  where J_2 is Earth's second zonal harmonic coefficient, R_E is Earth's equatorial radius, a is the semi-major axis, e is the orbital eccentricity, i is the orbital inclination, and

$n = \sqrt{\frac{\mu}{a^3}}$ is the mean motion, with $\mu = 3.986 \times 10^{14}$ m^3/s^2 being the Earth's gravitational parameter. The direction of RAAN drift depends on the inclination:

– For prograde orbits ($i < 90°$), RAAN decreases over time.
– For retrograde orbits ($i > 90°$), RAAN increases over time.
– For polar orbits ($i = 90°$), there is no RAAN variation.

The alignment time is computed as:

$$t_{R(0,1)} = \frac{\text{mod}\,(\Omega_{d(1)} - \Omega_{c_{\text{in}}}, \pm 2\pi)}{\dot{\Omega}}, \tag{18}$$

$$t_{R(i-1,i)} = \frac{\text{mod}\,(\Omega_{d(i)} - \Omega_{d(i-1)}, \pm 2\pi)}{\dot{\Omega}}, \tag{19}$$

where $t_{R(0,1)}$ is the time required for the chaser's initial RAAN $\Omega_{c_{\text{in}}}$ to align with the RAAN of the first debris object $\Omega_{d(1)}$, and $t_{R(i-1,i)}$ is the time required for the RAAN of the $(i-1)^{\text{th}}$ debris object $\Omega_{d(i-1)}$ to align with that of the $(i)^{\text{th}}$ debris object $\Omega_{d(i)}$, ensuring sequential targeting.

- *Hohmann Transfer Time t_H*: This is the time required to perform an orbital change using a Hohmann transfer maneuver. Specifically, the transfer time to the i^{th} debris is given by

$$t_{H(i)} = \pi \sqrt{\frac{a_{H(i)}^3}{\mu}}, \tag{20}$$

where the semi-major axis $a_{H(i)}$ of the transfer orbit is defined as

$$a_{H(i)} = \frac{a_{d(i)} + a_{\text{ril}}}{2}. \tag{21}$$

Here $a_{d(i)}$ is the semi-major axis of the debris orbit and a_{ril} is that of the release orbit.

- *Rendezvous Waiting Time t_W*: This waiting time ensures that, after executing the Hohmann maneuver, the chaser arrives at the debris object's orbit at the correct position for rendezvous. The lead angle, α_{lead}, is the angular displacement between the initial position of the target and the rendezvous point, given by

$$\alpha_{\text{lead}} = \omega_d \cdot t_H, \tag{22}$$

where ω_d is the angular velocity of the debris. Additionally, the final phase angle ϕ_f is the angular displacement between the chaser and the target at rendezvous:

$$\phi_f = \pi - \alpha_{lead}. \tag{23}$$

To ensure correct timing, the chaser must wait such that:

$$\phi_f = \phi_i + (w_d - w_{ril}) \cdot t_W, \tag{24}$$

where w_{ril} is the angular velocity of the chaser in its release orbit and ϕ_i represents the initial phase angle. This angle is defined as

$$\phi_i = u_d - u_c, \tag{25}$$

with u_d and u_c representing the arguments of latitude for the debris and the chaser, respectively, after RAAN alignment. Solving Eq. (24) for t_W, we obtain:

$$t_W = \frac{\phi_f - \phi_i}{w_d - w_{ril}} = \frac{\pi - w_d \cdot t_H - (u_d - u_c)}{w_d - w_{ril}}. \tag{26}$$

If t_W is negative, the Eq. (26) is adjusted using

$$t_W = \frac{\text{mod}\,(\pi - \omega_d t_H - (u_d - u_c), -2\pi)}{\omega_d - \omega_{\text{ril}}}. \tag{27}$$

Since we consider circular orbits ($e = 0$), the variation in the argument of latitude u is influenced only by the mean anomaly M:

$$\dot{u} = \dot{M} = n\left(1 + \frac{3}{4}J_2\left(\frac{R_E}{a(1-e^2)}\right)^2\sqrt{1-e^2}(3\cos^2(i) - 1)\right). \tag{28}$$

The chaser's argument of latitude u_c is updated as follows:

$$u_{c(i-1,i)} = u_{c_{\text{fin}}(i-1)} + \dot{u} \cdot t_{R(i-1,i)}\,. \tag{29}$$

For sequential captures, the debris argument of latitude is:

$$u_{d(0,1,\ldots,i)} = u_{d_{\text{in}}(i)} + \omega_{d(i)}\left(t_{R(i-1,i)} + \sum_{j=0}^{i-1}\text{TOF}_{(0,1,\ldots,j)}\right). \tag{30}$$

thus, the waiting time for the i^{th} capture is:

$$t_{W(0,1,\ldots,i)} = \frac{\text{mod}\,(\pi - \omega_{d(i)} \cdot t_{H(i)} - (u_{d(0,1,\ldots,i)} - u_{c(0,1,\ldots,i)}), -2\pi)}{\omega_{d(i)} - \omega_{\text{ril}}}. \tag{31}$$

This mathematical model accounts for all prior captures, emphasizing the sequential dependency in computing waiting times.

- *Capture Time* t_{call}: This is the time required to capture the debris object.
- *Release time* t_{rill}: This is the time needed to release the debris object after it has been successfully captured.

Delta-v Delta-v (Δv) denotes the instantaneous change in velocity that must be applied to the chaser to perform the required orbital maneuvers. Since Δv is directly proportional to fuel consumption, it represents a critical factor in the optimization process. For Hohmann transfer maneuvers, the velocity on the transfer orbit can be determined using the vis-viva equation:

$$v_{H,p} = \sqrt{\mu\left(\frac{2}{r_{H,p}} - \frac{1}{a_H}\right)}, \tag{32}$$

$$v_{H,a} = \sqrt{\mu\left(\frac{2}{r_{H,a}} - \frac{1}{a_H}\right)}. \tag{33}$$

Here, $v_{H,p}$ and $v_{H,a}$ denote the velocities at the periapsis and apoapsis of the Hohmann transfer orbit, respectively, while $r_{H,p}$ and $r_{H,a}$ are the radial distances at the periapsis and apoapsis. The total velocity Δv to execute the Hohmann transfer is expressed as:

$$\Delta v = \begin{cases} |v_f - v_{H,a}| + |v_i - v_{H,p}|, & \text{if } r_i < r_f\,, \\ |v_f - v_{H,p}| + |v_i - v_{H,a}|, & \text{if } r_i > r_f\,, \end{cases} \tag{34}$$

where v_f and v_i represent the orbital velocities at the final and initial orbits, respectively, and are given by:

$$v_f = \sqrt{\frac{\mu}{r_f}}, \qquad v_i = \sqrt{\frac{\mu}{r_i}}. \tag{35}$$

The total $\Delta v_{tot(i)}$ required for debris capture consists of two Hohmann maneuvers: one to reach the debris and another to return to the release orbit. The velocity components for the capture maneuver are given by:

$$v_{Hcat,p(i)} = \sqrt{\mu\left(\frac{2}{a_{ril}} - \frac{1}{a_{H(i)}}\right)}, \tag{36}$$

$$v_{Hcat,a(i)} = \sqrt{\mu\left(\frac{2}{a_{d(i)}} - \frac{1}{a_{H(i)}}\right)}, \tag{37}$$

where $v_{Hcat,p(i)}$ and $v_{Hcat,a(i)}$ are the velocities at the periapsis and apoapsis of the Hohmann transfer orbit used for capturing the debris. The corresponding velocity change required for the capture maneuver is:

$$\Delta v_{cat(i)} = \left|\sqrt{\frac{\mu}{a_{d(i)}}} - v_{Hcat,a(i)}\right| + \left|\sqrt{\frac{\mu}{a_{ril}}} - v_{Hcat,p(i)}\right|. \tag{38}$$

Similarly, the velocities for the return transfer to the release orbit are:

$$v_{Hril,p(i)} = \sqrt{\mu\left(\frac{2}{a_{d(i)}} - \frac{1}{a_{H(i)}}\right)}, \tag{39}$$

$$v_{Hril,a(i)} = \sqrt{\mu\left(\frac{2}{a_{ril}} - \frac{1}{a_{H(i)}}\right)}, \tag{40}$$

where $v_{Hril,p(i)}$ and $v_{Hril,a(i)}$ represent the velocities at the periapsis and apoapsis of the return transfer orbit. The corresponding velocity change for the return maneuver is:

$$\Delta v_{ril(i)} = \left|\sqrt{\frac{\mu}{a_{ril}}} - v_{Hril,a(i)}\right| + \left|\sqrt{\frac{\mu}{a_{d(i)}}} - v_{Hril,p(i)}\right|. \tag{41}$$

Thus, the total $\Delta v_{tot(i)}$ required for the complete mission, including both capture and return, is:

$$\Delta v_{tot(i)} = \Delta v_{cat(i)} + \Delta v_{ril(i)}\,. \tag{42}$$

Table 8 Environmental setting and orbitals in the various phases. All angles (RAAN and *u*) are expressed in degrees, normalized between 0° and 360°

Capture	Phase	COE (chaser)	DOE (debris)
1	Waiting Orbit (Initial)	a=6928.0 km, RAAN=47.34°, u=120.57°	a=7140.6 km, RAAN=160.41°, u=0.27°
	RAAN Alignment (J2 drift)	a=6928.0 km, RAAN=160.41°, u=41.19°	a=7140.6 km, RAAN=160.41°, u=164.26°
	Rendezvous Waiting	a=6928.0 km, RAAN=160.41°, u=207.56°	a=7140.6 km, RAAN=160.41°, u=211.57°
	Hohmann Transfer to Debris	a=7140.6 km, RAAN=160.41°, u=27.56°	a=7140.6 km, RAAN=160.41°, u=27.56°
	Capture Phase	a=7140.6 km, RAAN=160.41°, u=99.20°	a=7140.6 km, RAAN=160.41°, u=99.20°
	Return to Disposal Orbit	a=6928.0 km, RAAN=160.41°, u=279.20°	a=6928.0 km, RAAN=160.41°, u=279.20°
	Release Phase	a=6928.0 km, RAAN=160.41°, u=145.03°	a=6928.0 km, RAAN=160.41°, u=145.03°
2	Waiting Orbit (Initial)	a=6928.0 km, RAAN=160.41°, u=145.03°	a=6961.2 km, RAAN=145.68°, u=239.11°
	RAAN Alignment (J2 drift)	a=6928.0 km, RAAN=145.68°, u=31.54°	a=6961.2 km, RAAN=145.68°, u=122.29°
	Rendezvous Waiting	a=6928.0 km, RAAN=145.68°, u=52.91°	a=6961.2 km, RAAN=145.68°, u=53.55°
	Hohmann Transfer to Debris	a=6961.2 km, RAAN=145.68°, u=232.91°	a=6961.2 km, RAAN=145.68°, u=232.91°
	Capture Phase	a=6961.2 km, RAAN=145.68°, u=321.35°	a=6961.2 km, RAAN=145.68°, u=321.35°
	Return to Disposal Orbit	a=6928.0 km, RAAN=145.68°, u=141.35°	a=6928.0 km, RAAN=145.68°, u=141.35°
	Release Phase	a=6928.0 km, RAAN=145.68°, u=7.18°	a=6928.0 km, RAAN=145.68°, u=7.18°
3	Waiting Orbit (Initial)	a=6928.0 km, RAAN=145.68°, u=7.18°	a=7144.3 km, RAAN=144.19°, u=65.33°
	RAAN Alignment (J2 drift)	a=6928.0 km, RAAN=144.19°, u=341.11°	a=7144.3 km, RAAN=144.19°, u=8.78°
	Rendezvous Waiting	a=6928.0 km, RAAN=144.19°, u=144.73°	a=7144.3 km, RAAN=144.19°, u=148.80°
	Hohmann Transfer to Debris	a=7144.3 km, RAAN=144.19°, u=324.73°	a=7144.3 km, RAAN=144.19°, u=324.73°
	Capture Phase	a=7144.3 km, RAAN=144.19°, u=36.04°	a=7144.3 km, RAAN=144.19°, u=36.04°
	Return to Disposal Orbit	a=6928.0 km, RAAN=144.19°, u=216.04°	a=6928.0 km, RAAN=144.19°, u=216.04°
	Release Phase	a=6928.0 km, RAAN=144.19°, u=81.87°	a=6928.0 km, RAAN=144.19°, u=81.87°

Table 9 HQA performance and QUBO penalty parameters

Metric	Value
Penalty Parameter λ	1000.0
Total Computation Time [s]	69.366
Best Objective Value	−1.2136
Sequence (Capture Order)	[1, 91, 40]
Debris IDs (SATCAT)	[35,052, 33,867, 33,887]
Total Time of Flight [s]	47,986,254.86
Total Time of Flight [days]	555.396
Total Δv [m/s]	495.091
Desirability Score	1.7202
Simulated TOF [s]	4.7986×10^7
Simulated Δv [m/s]	495.0910

A.2 Quadratization method

The approximated TOF' exploited in the alternative quadratization approach are defined as follows:

$$TOF'_{\mathrm{f,jk}} = t_{R(j,k)} + t_{R_{\mathrm{in}}(j)} + T_{\mathrm{synodic},k} + 2 \cdot t_{H(j)}, \quad \forall j,k = 1,2,\ldots,D, \quad j \neq k\,, \tag{43}$$

where $t_{R_{\mathrm{in}}(j)}$ is the RAAN time for the transfer from the initial disposal orbit to the first debris collected, and

$$TOF'_{jk} = t_{R(j,k)} + T_{\mathrm{synodic},j} + 2 \cdot t_{H(j)}, \quad \forall j,k = 1,2,\ldots,D, \quad j \neq k\,. \tag{44}$$

In this approximation, the waiting time is replaced by the synodic time, avoiding its dependence on the entire sequence of captures and instead considering only the previously collected debris. Moreover, the capture and release times are neglected, as they are constant and do not influence the optimization process.

A.3 Settings

In this section, we report the environmental context and QUBO configuration used for the 3-capture, 114-debris test scenario performed with the HQA solver, as a representative example. Specifically, Table 8 summarizes the environmental parameters and the orbital elements at each mission phase, while Table 9 presents the selected QUBO penalty parameters along with the performance metrics achieved by the HQA solver.

Acknowledgements
We would like to thank D-Wave Systems for providing access to their quantum annealer through the Leap Quantum LaunchPad program.

Author contributions
Conceptualization, all authors; methodology, all authors; software, M.G.; validation, M.G., M.B. and D.V.; formal analysis, M.G., M.B. and D.V.; writing—original draft preparation, all authors; writing—review and editing, all authors; supervision, C.N.; project administration, C.N. All authors have read and agreed to the published version of the manuscript.

Funding information
Not Applicable.

Data availability
No datasets were generated or analysed during the current study.

Declarations

Research involving human and /or animals
Not Applicable.

Consent for publication
Not Applicable.

Competing interests
The authors declare no competing interests.

Author details
[1]Politecnico di Torino, Corso Duca degli Abruzzi 24, Turin, 10124, Italy. [2]Istituto Nazionale di Geofisica e Vulcanologia, Rome, Italy.

Received: 18 March 2025 Accepted: 13 August 2025 Published online: 29 September 2025

References

1. European Space Agency. ESA Space Environment Report 2024. https://www.esa.int/Space_Safety/Space_Debris/ESA_Space_Environment_Report_2024.

2. Kessler DJ, Cour-Palais BG. Collision frequency of artificial satellites: the creation of a debris belt. J Geophys Res Space Phys. 1978;83(A6):2637–46.
3. Kessler DJ, Johnson NL, Liou J, Matney M. The Kessler syndrome: implications to future space operations. Adv Astronaut Sci. 2010;137(8):2010.
4. Inter-agency Space Debris Coordination Committee. IADC Space Debris Mitigation Guidelines. UN COPUOS 40th session, Vienna. 2002.
5. Liou J-C, Johnson NL. Instability of the present Leo satellite populations. Adv Space Res. 2008;41(7):1046–53.
6. Bonnal C, Ruault J-M, Desjean M-C. Active debris removal: recent progress and current trends. Acta Astronaut. 2013;85:51–60.
7. Mark CP, Kamath S. Review of active space debris removal methods. Space Policy. 2019;47:194–206.
8. Zhao P, Liu J, Wu C. Survey on research and development of on-orbit active debris removal methods. Sci China, Technol Sci. 2020;63(11):2188–210.
9. Arshad M, Bazzocchi MCF, Hussain F. Emerging strategies in close proximity operations for space debris removal: a review. Acta Astronaut. 2025;228:996–1022.
10. Machula M, Sandhoo G. Rendezvous and docking for space exploration. In: 1st space exploration conference: continuing the voyage of discovery. 2005.
11. Gaylor D, Barbee B. Algorithms for safe spacecraft proximity operations. In: American Astronautical Society - space flight mechanics 2007 - advances in the astronautical sciences. Proceedings of the AAS/AIAA space flight mechanics meeting. Advances in the astronautical sciences. 2007. p. 133–52.
12. Sellmaier F, Boge T, Spurmann J, Gully S, Rupp T, Huber F. On-orbit servicing missions: challenges and solutions for spacecraft operations. In: SpaceOps 2010 conference delivering on the dream hosted by NASA marshall space flight center and organized by AIAA. 2010.
13. Whelan DA, Adler EA, Wilson SB III, Roesler GM Jr. DARPA Orbital Express program: effecting a revolution in space-based systems. In: Horais BJ, Twiggs RJ, editors. Small payloads in space. Society of Photo-optical Instrumentation Engineers (SPIE) conference series. vol. 4136. 2000. p. 48–56.
14. Blackerby C, Okamoto A, Fujimoto K, Okada N, Forshaw JL, Auburn J. Elsa-d: an in-orbit end-of-life demonstration mission. In: Proceedings of the International Astronautical Congress, IAC. vol. 6. 2018.
15. Blackerby C, Okamoto A, Iizuka S, Kobayashi Y, Fujimoto K, Seto Y, Fujita S, Iwai T, Okada N, Forshaw J, et al. The elsa-d end-of-life debris removal mission: preparing for launch. In: Proceedings of the International Astronautical Congress, IAC. vol. 8. 2019.
16. Biesbroek R, Aziz S, Wolahan A, Cipolla S-f, Richard-Noca M, Piguet L. The clearspace-1 mission: ESA and clearspace team up to remove debris. In: Proc. 8th eur. conf. Sp. debris. 2021. p. 1–3.
17. Liou J-C, Johnson NL. A sensitivity study of the effectiveness of active debris removal in Leo. Acta Astronaut. 2009;64(2):236–43.
18. Liou J-C, Johnson NL, Hill NM. Controlling the growth of future Leo debris populations with active debris removal. Acta Astronaut. 2010;66(5):648–53.
19. Lewis HG, White AE, Crowther R, Stokes H. Synergy of debris mitigation and removal. Acta Astronaut. 2012;81(1):62–8.
20. Wijayatunga MC, Armellin R, Holt H, Pirovano L, Lidtke AA. Design and guidance of a multi-active debris removal mission. Astrodynamics. 2023;7(4):383–99.
21. Medioni L, Gary Y, Monclin M, Oosterhof C, Pierre G, Semblanet T, Comte P, Nocentini K. Trajectory optimization for multi-target active debris removal missions. Adv Space Res. 2023;72(7):2801–23. Space Environment Management and Space Sustainability.
22. Flood MM. The traveling-salesman problem. Oper Res. 1956;4(1):61–75.
23. Barbee BW, Alfano S, Piñon E, Gold K, Gaylor D. Design of spacecraft missions to remove multiple orbital debris objects. In: 2011 aerospace conference. 2011. p. 1–14.
24. Cerf M. Multiple space debris collecting mission—debris selection and trajectory optimization. J Optim Theory Appl. 2013;156(3):761–96.
25. Izzo D, Getzner I, Hennes D, Simões LF. Evolving solutions to TSP variants for active space debris removal. In: Proceedings of the 2015 annual conference on genetic and evolutionary computation. GECCO '15. New York: Association for Computing Machinery; 2015. p. 1207–14.
26. Shen H-X, Zhang T-J, Casalino L, Pastrone D. Optimization of active debris removal missions with multiple targets. J Spacecr Rockets. 2018;55(1):181–9.
27. Federici L, Zavoli A, Colasurdo G. On the use of a* search for active debris removal mission planning. J Space Saf Eng. 2021;8(3):245–55.
28. Federici L, Zavoli A, Colasurdo G. A time-dependent tsp formulation for the design of an active debris removal mission using simulated annealing. Adv Astronaut Sci. 2019:1349–1368.
29. Missel J, Mortari D. Path optimization for space sweeper with sling-sat: a method of active space debris removal. Adv Space Res. 2013;52(7):1339–48.
30. Zona D, Zavoli A, Federici L, Avanzini G. Evolutionary optimization for active debris removal mission planning. IEEE Access. 2023;11:41019–33.
31. Liu Y, Yang J, Wang Y, Pan Q, Yuan J. Multi-objective optimal preliminary planning of multi-debris active removal mission in Leo. Sci China Inf Sci. 2017;60:1–10.
32. Yang J, Hou X, Hu YH, Liu Y, Pan Q. A reinforcement learning scheme for active multi-debris removal mission planning with modified upper confidence bound tree search. IEEE Access. 2020;8:108461–73.
33. Viavattene G, Devereux E, Snelling D, Payne N, Wokes S, Ceriotti M. Design of multiple space debris removal missions using machine learning. Acta Astronaut. 2022;193:277–86.
34. Wang Y, Kim JE, Suresh K. Opportunities and challenges of quantum computing for engineering optimization. J Comput Inf Sci Eng. 2023;23(6):060817.
35. Abbas A, Ambainis A, Augustino B, Bärtschi A, Buhrman H, Coffrin C, Cortiana G, Dunjko V, Egger DJ, Elmegreen BG, et al. Challenges and opportunities in quantum optimization. Nat Rev Phys. 2024:1–18.
36. Barenco A, Bennett CH, Cleve R, DiVincenzo DP, Margolus N, Shor P, Sleator T, Smolin JA, Weinfurter H. Elementary gates for quantum computation. Phys Rev A. 1995;52(5):3457.

37. Albash T, Lidar DA. Adiabatic quantum computation. Rev Mod Phys. 2018;90(1):015002.
38. Kadowaki T, Nishimori H. Quantum annealing in the transverse Ising model. Phys Rev E. 1998;58(5):5355.
39. Johnson MW, Amin MH, Gildert S, Lanting T, Hamze F, Dickson N, Harris R, Berkley AJ, Johansson J, Bunyk P, et al. Quantum annealing with manufactured spins. Nature. 2011;473(7346):194–8.
40. Yarkoni S, Raponi E, Bäck T, Schmitt S. Quantum annealing for industry applications: introduction and review. Rep Prog Phys. 2022;85(10):104001.
41. Orús R, Mugel S, Lizaso E. Quantum computing for finance: overview and prospects. Rev Phys. 2019;4:100028.
42. Streif M, Neukart F, Leib M. Solving quantum chemistry problems with a d-wave quantum annealer. In: Quantum technology and optimization problems: first international workshop, QTOP 2019, Munich, Germany, March 18, 2019. Proceedings. vol. 1. Springer; 2019. p. 111–22.
43. Inoue D, Yoshida H. Model predictive control for finite input systems using the d-wave quantum annealer. Sci Rep. 2020;10(1):1591.
44. Novara C, Boggio M, Volpe D. A quantum optimization approach to nonlinear model predictive control. In: 2025 European Control Conference (ECC). 2025.
45. Crawford D, Levit A, Ghadermarzy N, Oberoi JS, Ronagh P. Reinforcement learning using quantum Boltzmann machines. 2016. arXiv preprint. arXiv:1612.05695.
46. Date P, Arthur D, Pusey-Nazzaro L. Qubo formulations for training machine learning models. Sci Rep. 2021;11(1):10029.
47. Biswas R, Jiang Z, Kechezhi K, Knysh S, Mandrà S, O'Gorman B, Perdomo-Ortiz A, Petukhov A, Realpe-Gómez J, Rieffel E, Venturelli D, Vasko F, Wang Z. A nasa perspective on quantum computing: opportunities and challenges. Parallel Comput. 2017;64:81–98. High-End Computing for Next-Generation Scientific Discovery.
48. Chang Z, Chen Y, Yang W, Zhou Z. Mission planning problem for optical video satellite imaging with variable image duration: a greedy algorithm based on heuristic knowledge. Adv Space Res. 2020;66(11):2597–609.
49. Liu Y, Chen Q, Li C, Wang F. Mission planning for Earth observation satellite with competitive learning strategy. Aerosp Sci Technol. 2021;118:107047.
50. Chang Z, Punnen AP, Zhou Z, Cheng S. Solving dynamic satellite image data downlink scheduling problem via an adaptive bi-objective optimization algorithm. Comput Oper Res. 2023;160:106388.
51. Wang M, Zhou Z, Chang Z, Chen E, Li R. Deep reinforcement learning for agile Earth observation satellites scheduling problem with variable image duration. Appl Soft Comput. 2025;169:112575.
52. Stollenwerk T, Michaud V, Lobe E, Picard M, Basermann A, Botter T. Agile Earth observation satellite scheduling with a quantum annealer. IEEE Trans Aerosp Electron Syst. 2021;57(5):3520–8.
53. Marchioli V, Boggio M, Volpe D, Massotti L, Novara C. Scheduling of satellite constellation operations in EO missions using quantum optimization. In: Optimization, learning algorithms and applications. Cham: Springer; 2024. p. 227–42.
54. Marchioli V, Boggio M, Volpe D, Massotti L, Novara C. Quantum optimization for closed-loop scheduling of Earth observation satellite formation. SN Comput Sci. 2025;6:6.
55. Snelling D, Devereux E, Payne N, Nuckley M, Viavattene G, Ceriotti M, Wokes S, Di Mauro G, Brettle H. Innovation in planning space debris removal missions using artificial intelligence and quantum-inspired computing. In: 8th European conference on space debris. 2021.
56. Matsubara S, Takatsu M, Miyazawa T, Shibasaki T, Watanabe Y, Takemoto K, Tamura H. Digital annealer for high-speed solving of combinatorial optimization problems and its applications. In: 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC). 2020. p. 667–72.
57. Swain T. Optimisation of active space debris removal missions with multiple targets using quantum annealing. 2023. arXiv preprint. arXiv:2311.01852.
58. Booth M, Reinhardt SP, Roy A. Partitioning optimization problems for hybrid classical/quantum execution. Technical Report, 01–09. 2017.
59. Dobarco-Otero J. Upgrades to Object Reentry Survival Analysis Tool (ORSAT) for Spacecraft and Launch Vehicle Upper Stage Applications.
60. Pontijas Fuentes I, Bonetti D, Letterio F, Vicario de Miguel G, Blanco Arnao G, Palomo P, Parigini C, Lemmens S, Lips T, Kanzler R. Upgrade of ESA's Debris Risk Assessment and Mitigation Analysis (DRAMA) tool: spacecraft entry survival analysis module. Acta Astronaut. 2019;158:148–60.
61. Klinkrad H. Space debris: models and risk analysis. Berlin: Springer; 2006.
62. Lewis HG, Yazadzhiyan V. Evaluation of low Earth orbit post-mission disposal measures. J Space Saf Eng. 2024;11(3):526–31.
63. Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science. 1983;220(4598):671–80.
64. Glover F. Tabu search: a tutorial. Interfaces. 1990;20(4):74–94.
65. Kramer O. Genetic algorithms. Cham: Springer; 2017. p. 11–9.
66. Vallado D, Crawford P. SGP4 orbit determination. In: AIAA/AAS astrodynamics specialist conference and exhibit. 2008.
67. Jain S. Solving the traveling salesman problem on the d-wave quantum computer. Front Phys. 2021;9:760783.
68. Glover F. Tabu search: a tutorial. Interfaces. 1990;20(4):74–94.
69. Palubeckis G. Multistart tabu search strategies for the unconstrained binary quadratic optimization problem. Ann Oper Res. 2004;131:259–82.
70. Mitchell M. Genetic algorithms: an overview. In: Complex. vol. 1. 1995. p. 31–9. Citeseer.
71. Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science. 1983;220(4598):671–80.
72. Morita S, Nishimori H. Mathematical foundation of quantum annealing. J Math Phys. 2008;49(12):125210.
73. Johnson MW, Amin MH, Gildert S, Lanting T, Hamze F, Dickson N, Harris R, Berkley AJ, Johansson J, Bunyk P, et al. Quantum annealing with manufactured spins. Nature. 2011;473(7346):194–8.
74. Boixo S, Ortiz G, Somma R. Fast quantum methods for optimization. Eur Phys J Spec Top. 2015;224(1):35–49.
75. Zanca T, Santoro GE. Quantum annealing speedup over simulated annealing on random Ising chains. Phys Rev B. 2016;93:224431.
76. Hauke P, Katzgraber HG, Lechner W, Nishimori H, Oliver WD. Perspectives of quantum annealing: methods and implementations. Rep Prog Phys. 2020;83(5):054401.
77. D-Wave Systems https://www.dwavesys.com/.

78. Volpe D, Orlandi G, Turvani G. Improving the solving of optimization problems: a comprehensive review of quantum approaches. Quantum Rep. 2025;7(1):3.
79. D-Wave Systems Inc. Leap Hybrid Solver. 2023. Available https://www.dwavequantum.com/resources/white-paper/d-wave-hybrid-solver-service-an-overview/.
80. Volpe D, Cirillo GA, Zamboni M, Turvani G. Integration of simulated quantum annealing in parallel tempering and population annealing for heterogeneous-profile qubo exploration. IEEE Access. 2023;11:30390–441.
81. CelesTrak. NORAD GP Element Sets. https://celestrak.org/NORAD/elements/.

Publisher's note