aerospace

MDPI

Article

A Multi-Task Dynamic Scheduling Method for Space Launch TT&C Resources Based on Priority Rules and Adaptive NSGA-II

Lisong Hao [1], Yunfeng Liang [2], Taibo Li [1] and Hongwei Liu [1,*]

[1] National Innovation Institute of Defense Technology, Chinese Academy of Military Science, Beijing 100071, China; 19118408802@163.com (L.H.); taibo_li@163.com (T.L.)
[2] PLA Unit 63729, Taiyuan 030032, China; 19148642579@163.com
* Correspondence: liuhw05@163.com; Tel.: +86-181-1007-9554

Abstract

To address the strong constraints, multiple objectives, and high dynamism of space telemetry, tracking, and command (TT&C) resource scheduling in flight-type launch scenarios, this study proposes a dynamic scheduling approach that integrates priority rules with adaptive multi-task evolution. First, a mixed-integer programming model is developed to capture fixed and mobile equipment, diverse mission requirements, and spatiotemporal coupling constraints, with the optimization objectives of minimizing the total mobile distance, the total number of deployed devices, and the number of mobile devices deployed. Second, a priority-based dynamic rescheduling mechanism is designed to support rolling insertion of emergency tasks and, when necessary, adjust conflicting tasks according to priority rules. Third, an improved NSGA-II algorithm is introduced, incorporating adaptive population adjustment, early stopping, and decoding caching to enhance multi-task search efficiency and convergence stability. Finally, a simulation experiment is constructed based on a typical commercial launch scenario, targeting three types of static task scenarios of different scales; the proposed Adaptive-NSGA-II consistently yields feasible schedules, while reducing the solution time by 70.4%, 62.7%, and 61.0%, respectively, compared with standard NSGA-II. In the dynamic emergency task insertion scenarios, the proposed priority-rule-based rolling rescheduling strategy successfully completes insertion in all three cases, achieves zero additional mobile distance in low-conflict scenarios, and remains significantly superior to the fixed-time direct insertion strategy even under high-conflict conditions. The experimental results demonstrate the effectiveness, robustness, and engineering applicability of the proposed method for TT&C resource scheduling in flight-type launch operations.

Keywords: flight-type launch; TT&C resource scheduling; emergency task insertion; NSGA-II; multi-task optimization; dynamic scheduling

Academic Editor: Vladimir S. Aslanov

Received: 15 May 2026
Revised: 26 June 2026
Accepted: 29 June 2026
Published: 30 June 2026

1. Introduction

With the rapid development of the commercial space industry and the accelerated deployment of large low-Earth-orbit constellations, launch operations are shifting from traditional low-frequency, mission-specific organization toward high-frequency, standardized, flight-type launch modes. Such flight-type launch operations emphasize dense

mission organization, rapid response, and reliable support, thereby imposing significantly higher requirements on the resource configuration efficiency and dynamic scheduling capability of ground systems. As a critical infrastructure connecting mission command, vehicle state acquisition, and process control, TT&C systems perform essential functions such as telemetry, tracking, and safety control; their scheduling performance directly affects mission safety, continuity, and overall execution efficiency.

Compared with conventional single-mission or low-density launch scenarios, TT&C resource scheduling in flight-type launch operations is substantially more complex. On the one hand, parallel launches of multiple rockets lead to denser spatiotemporal distributions of TT&C tasks, with substantial differences among task arcs in terms of time windows, equipment types, and service-quality requirements. On the other hand, TT&C resources are inherently heterogeneous, including both fixed equipment with immutable geographic locations and mobile equipment capable of repositioning but constrained by transfer time and mobility cost. In particular, under dynamic conditions such as emergency task insertion, schedule adjustments, or localized resource conflicts, the scheduling system must not only generate feasible plans rapidly but also minimize disruption to the existing schedule. Therefore, how to achieve efficient scheduling of multiple tasks, equipment types, and coupled constraints under limited resources has become a key issue in TT&C support for flight-type launch operations.

Existing studies have established a relatively rich methodological basis for TT&C resource scheduling; however, substantial gaps remain for dynamic, multi-task, and strongly constrained scenarios in flight-type launch operations. First, some studies focus mainly on static task allocation and are not well suited to dynamic scheduling demands such as emergency task insertion and conflict propagation. Second, many optimization models emphasize visibility coverage or task completion rate, while insufficiently accounting for service-quality constraints such as measurement accuracy. Third, at the algorithmic level, exact methods can provide high-quality solutions but are limited by computational efficiency in large-scale complex scenarios, whereas purely heuristic rules often fail to balance global optimization capability and dynamic responsiveness. Accordingly, it is necessary to develop a TT&C resource scheduling method tailored to flight-type launch scenarios that simultaneously accounts for coverage quality, resource economy, and dynamic adaptability.

Motivated by these challenges, this paper proposes a solution framework for the multi-task dynamic scheduling of space TT&C resources by integrating priority rules with adaptive NSGA-II. For fixed/mobile hybrid TT&C resources, the proposed model jointly considers equipment-type matching, coverage constraints, measurement accuracy constraints, spatiotemporal compatibility of mobile equipment, and dynamic adjustment constraints. In the static scheduling stage, adaptive NSGA-II is employed to obtain a baseline schedule that balances resource utilization efficiency and mobility cost. In the dynamic scheduling stage, for emergency task insertion, a priority-rule-based rolling rescheduling mechanism is designed to ensure preferential support for high-priority tasks while enabling low-disruption rapid adjustment.

The main contributions of this work are as follows:

1. A multi-task TT&C resource scheduling model is developed for flight-type launch scenarios. The model unifies the cooperation of fixed and mobile equipment and integrates coverage, measurement accuracy constraints, and spatiotemporal constraints into a single optimization framework, thereby improving engineering applicability.
2. An Adaptive-NSGA-II method is proposed for complex constrained scenarios. By introducing decoding caching, adaptive population adjustment, and early stopping,

the efficiency and stability of multi-task evolutionary optimization are improved, enabling rapid generation of high-quality non-dominated solutions.

3. A priority-rule-based dynamic rescheduling mechanism is proposed. For resource conflicts triggered by emergency task insertion, a rolling horizon search and conflict arbitration strategy is designed to guarantee high-priority tasks while minimizing disruption to the original schedule.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 formulates the space TT&C resource scheduling problem. Section 4 presents the quantitative calculations of coverage and accuracy constraints. Section 5 introduces the adaptive NSGA-II-based multi-task-scheduling method. Section 6 describes the priority-rule-based dynamic rescheduling mechanism. Section 7 reports simulation experiments and results. Section 8 concludes the paper and outlines future work.

2. Literature Review

In recent years, TT&C resource scheduling and related task planning problems have been extensively studied from the perspectives of exact optimization, heuristic and metaheuristic solution methods, and dynamic rescheduling, resulting in a well-established methodological system. The relevant advances from these three aspects are reviewed below.

2.1. Exact Optimization Methods

Early research on TT&C resource scheduling mainly adopted exact optimization approaches, typically mixed-integer linear programming (MILP), integer programming, and constraint satisfaction problems (CSPs). Gooley et al. were among the first to formulate satellite range scheduling as a solvable mathematical programming problem, laying the foundation for subsequent research [1]. Tang et al. further considered task splitting, temporal dependencies, and space–ground TT&C resource constraints, leading to a more realistic spacecraft task-scheduling model [2]. Marinelli et al. employed Lagrangian relaxation to decompose a large-scale satellite range scheduling problem, effectively alleviating the computational burden induced by resource constraints [3]. Park et al. developed an MILP model for multi-UAV mission allocation that jointly considered platform speed, fuel consumption, payload capacity, and task type, and solved it directly using an exact solver, demonstrating its advantages in optimizing task execution order and reducing resource consumption when resources are sufficient [4]. Rigo et al. proposed a branch-and-price algorithm for nanosatellite task scheduling; by applying Dantzig–Wolfe decomposition, the original problem was transformed into task-level subproblems, and the pricing subproblem was solved via dynamic programming, significantly improving efficiency on large-scale instances [5]. Xu Di-dong formulated an integer linear programming model for space–ground station network resource scheduling and designed a Lagrangian relaxation algorithm, which achieved rapid and exact solutions for scenarios involving multi-channel communication and launch–reentry constraints by relaxing cooperative target TT&C count constraints and decomposing subproblems [6].

Although exact methods provide strong global optimality guarantees for small- and medium-scale problems, their computational complexity typically increases rapidly with task scale and constraint count. For flight-type launch scenarios involving large-scale concurrent tasks, heterogeneous resources, and spatiotemporal coupling constraints, exact optimization alone often fails to satisfy engineering requirements for rapid solution generation.

https://doi.org/10.3390/aerospace13070605

2.2. Heuristic and Metaheuristic Methods

To improve solution efficiency in complex scenarios, researchers have extensively used heuristic and metaheuristic methods for TT&C resource scheduling. Rule-based constructive heuristics typically generate feasible solutions quickly using task priority, conflict relations, or resource benefit information. For example, Zhao Gaopeng et al. developed a greedy TT&C scheduling method for large constellations [7]; Li Cheng et al. proposed a multi-satellite, multi-station resource allocation method based on a priority-assisted divide-and-conquer strategy [8]; and Xin Liqiang et al. designed a conflict-avoidance scheduling algorithm for complex coupled TT&C requirements [9]. Although computationally efficient, these methods are often constrained by local decision-making and therefore cannot fully exploit the global solution space.

As problem scale and objective diversity increase, metaheuristic methods such as genetic algorithms, particle swarm optimization, simulated annealing, and multi-task evolutionary algorithms have attracted increasing attention. Lei et al. proposed MC-NSGA-II for emergency active debris removal tasks, enhancing multi-task search performance through multi-type encoding and dynamic parameter adjustment [10]. Chen et al. introduced a genetic algorithm with population perturbation and elimination strategies for multi-satellite TT&C scheduling, improving task completion and resource benefit in high-conflict scenarios. This method, referred to as GA-PE, is adopted as one of the comparative algorithms in our experimental study [11]. Zhang Junxin et al. designed a two-layer encoded genetic algorithm for multi-mission parallel planning in space launch sites, generating optimized plans for multi-rocket parallel tests within short computation time [12]. Ma Lin et al. combined simulated annealing with a genetic algorithm for large-constellation TT&C mission planning to improve global search capability and stability [13]. Yu et al. proposed TACCS, a two-stage algorithm for large-scale satellite TT&C scheduling; the first stage uses circle-combination selection to filter the feasible solution space, and the second stage applies simulated annealing or genetic algorithms for optimization, significantly improving efficiency and reducing antenna idle time while satisfying complex constraints [14].

In addition, multi-task-scheduling studies in related domains also provide useful references. Huang et al. used NSGA-II to solve a multi-task vehicle routing problem [15], Xu et al. developed an improved NSGA-II model for a capacitated green vehicle routing problem with time-varying speeds and soft time windows [16], and Ye et al. applied NSGA-II to airline crew scheduling to jointly account for cost and fatigue [17]. These studies indicate that multi-task evolutionary algorithms are highly suitable for complex constrained, multi-task trade-off problems; however, in engineering scheduling problems dominated by strong constraints and discrete decisions, problem-specific improvements to encoding, decoding, and evolutionary control are still required.

2.3. Dynamic Rescheduling Methods

Compared with static scheduling, dynamic rescheduling focuses more on rapid response and low-disruption adjustment under unexpected events. Sun et al. proposed a dynamic adjustment method based on feasible execution sequences and a greedy algorithm for scenarios involving new tasks and platform failures [18]. Tian et al. investigated dynamic flight scheduling optimization for airport runway conflicts [19]. Fang et al. studied emergency capacity scheduling and combined multi-stage decomposition with multiple optimization algorithms to achieve rapid response [20].

Although existing dynamic scheduling studies have achieved progress across different domains, research specific to TT&C resource scheduling in flight-type launch scenarios remains limited. The main deficiencies are: (i) most studies are centered on static baseline scheduling and lack fine-grained modeling of resource transfer and cascading con-

https://doi.org/10.3390/aerospace13070605

flicts triggered by emergency task insertion; (ii) dynamic adjustment objectives are often limited to task completion or response speed, with insufficient simultaneous consideration of mobile cost, disruption to the baseline schedule, and structural stability of resource allocation; and (iii) although priority rules are widely used in local conflict arbitration, the linkage between such rules and globally optimized multi-task schedules remains inadequate.

2.4. Research Gaps and Motivation

In summary, existing research on TT&C resource scheduling has evolved from exact optimization methods to heuristics, metaheuristics, and dynamic rescheduling approaches. However, comprehensive studies targeting high-frequency concurrency, multi-task trade-offs, and emergency event response under airline-style launch scenarios still have the following deficiencies:

1. Insufficient multi-objective coordination. Most existing studies focus on task completion rate, resource occupancy, or a single cost indicator, lacking unified modeling and collaborative optimization of objectives such as equipment travel distance, total number of activated equipment units, and dependence on mobile resources.
2. Inadequate modeling of quality constraints. Most methods allocate resources mainly based on visibility or coverage, without deeply embedding key TT&C quality indicators such as position accuracy and velocity accuracy into the scheduling model.
3. Underdeveloped dynamic insertion mechanisms. For emergency task insertion scenarios, existing research does not systematically address issues such as priority-based conflict resolution, local release and backfilling, and dynamic disturbance control.
4. Insufficient real-time solving capability. Although traditional multi-task evolutionary algorithms have good global search ability, they often suffer from slow convergence and high repeated decoding overhead in complex constrained scenarios, making it difficult to meet the need for rapid decision-making in highly dynamic environments.

To address the above shortcomings, this paper proposes a multi-task dynamic scheduling method for space TT&C resources that integrates priority rules with an adaptive NSGA-II. Through an overall framework of "static multi-task optimization + dynamic rolling rescheduling", it achieves high-quality baseline scheduling and fast insertion of emergency tasks under complex constraints, providing a solution that balances efficiency, quality and engineering feasibility for TT&C support in airline-style launch scenarios.

3. Mathematical Modeling of the Space TT&C Resource Scheduling Problem

3.1. Problem Description

In the context of scheduled-flight-mode (airline-mode) launch operations, the space telemetry, tracking, and command (TT&C) resource scheduling problem can be formulated as follows. Within a given planning horizon, multiple concurrent launch missions generate simultaneous demands for TT&C resources across numerous observation arcs. The core challenge is to optimally allocate fixed tracking stations and schedule mobile TT&C equipment units to each mission arc—determining the deployment sites and operational sequences of all assets—while satisfying a complex set of physical, temporal, and service-quality constraints, so as to yield a globally optimal scheduling plan.

The resource pool consists of heterogeneous TT&C equipment of multiple types, including radar, optical, telemetry, and range safety-control systems. A subset of these units are deployed at fixed tracking stations with known geographic positions and predefined

availability windows; the remaining units are mobile, capable of repositioning between deployment sites subject to non-negligible transit and setup times. Within the scheduling horizon, a set of TT&C mission arcs must be serviced, each corresponding to a distinct phase of a rocket's flight trajectory. Each arc is characterized by a prescribed time window, equipment-type requirements, minimum coverage rate threshold, and measurement accuracy requirements. Each arc is further categorized into one of two arc types—critical and general—which determine the coverage and accuracy thresholds to be satisfied. In addition, each mission is assigned a task priority from a five-level system (detailed in Section 6.4), which governs resource preemption and conflict resolution during dynamic rescheduling.

During execution, multiple constraint classes govern the assignment of equipment to mission arcs. Each equipment unit must arrive at the designated deployment site and complete its setup procedure before the start of the assigned arc. A single equipment unit can service at most one mission at any given time instant, and sufficient inter-task time must be reserved between consecutive assignments to accommodate equipment teardown and repositioning. The mobility of mobile units is bounded by inter-site distances and vehicle speed. To guarantee TT&C service quality, the realized coverage rate of each arc must not fall below the prescribed threshold, and the measurement accuracy must comply with mission-specific performance requirements.

In essence, the space TT&C resource scheduling problem is a spatiotemporally constrained, multi-resource, multi-mission combinatorial optimization problem. The decision variables encompass equipment selection and assignment, mobile equipment routing and timing, and mission execution sequencing. Given the large number of mission arcs, the diversity of equipment types, and the complexity of constraint interactions, this problem is proven to be NP-hard, requiring the search for Pareto-optimal solutions within the feasible space while simultaneously satisfying all operative constraints.

3.2. Notation and Model Formulation

To circumvent the difficulty of directly resolving the coupling among coverage rates, measurement accuracy, and equipment routing in the continuous spatiotemporal domain, this study adopts a hierarchical modeling paradigm organized as: launch mission → TT&C arc → feasible combination → equipment assignment. Specifically, prior to the online scheduling stage, candidate TT&C combinations satisfying the prescribed coverage rate and accuracy thresholds are generated offline based on ballistic geometry and measurement precision analysis. The online scheduling stage then selects among these prescreened candidate combinations and makes equipment assignment and dynamic rescheduling decisions. The principal sets, parameters, and decision variables of the model are defined as follows.

Let the set of space launch missions within the scheduling horizon be R, where $r \in R$ denotes a rocket or a launch mission, and its mission priority is denoted as P_r. The flight process of each rocket can be discretized into several TT&C arcs. Let $I_r \subseteq I$ denote the set of arcs associated with rocket r, where I is the set of all TT&C arcs. For an arc $i \in I_r$, its time window is given by $[t_i^s, t_i^e]$, and the set of required TT&C equipment types is denoted as $K_i \subseteq K$, where $K = \{Radar, Optical, Telemetry, Safety\ Control\}$ represents the complete set of equipment types. To ensure mission quality of service, arc i is respectively assigned a minimum coverage threshold η_i, a position accuracy threshold $\epsilon_i^{\mathrm{pos}}$, and a velocity accuracy threshold $\epsilon_i^{\mathrm{vel}}$.

Let the collection of all TT&C equipment be $E = E^f \cup E^m$, where E^f is the set of fixed equipment, and E^m is the set of mobile equipment. For any equipment $e \in E$, its type is given by $k_e \in K$, its initial deployment point is $p_e^0 \in P$, and its maximum continuous working duration is H_e. To characterize the preference for using certain equipment

in scheduling, a comprehensive equipment preference score s_e is defined, which can be formed by weighting factors such as equipment age, reliability, operating cost, transportation cost, and the like.

Let the set of TT&C equipment deployment points be P, and denote the distance between any two points $p, q \in P$ as d_{pq}. Let the average transfer speed of mobile equipment be v. If equipment transfers from point p to point q, the switching time is defined as

$$\theta_{pq} = \tau_{\text{strip}} + \tau_{\text{set}} + \frac{d_{pq}}{v} \tag{1}$$

where τ_{strip} is the dismantling time, and τ_{set} is the setup preparation time. When $p = q$, the equipment is considered to operate at the same point, requiring only a short setup time τ_{set}. This switching time θ_{pq} is also used in the dynamic rescheduling phase (Section 6.3) to evaluate whether a mobile equipment can reach an emergency task's target point before its start time.

To reduce the computational complexity in the online scheduling phase, this study first performs a pre-screening of deployable points for each arc under different equipment types based on coverage and accuracy requirements, and further constructs feasible combinations. For an arc i and equipment type $k \in K_i$, the coverage of the arc when equipment is deployed at point $p \in P$ is denoted as $\eta_{ikp} \in [0,1]$. For internal measurement equipment types (e.g., telemetry) and safety-control equipment, only coverage is used as the criterion for point feasibility. If $\eta_{ikp} \geq \eta_i$, point p is considered feasible for arc i under equipment type k. For external-measurement equipment types (e.g., radar, optical), accuracy requirements must also be satisfied. Let the position accuracy and velocity accuracy of point p for arc i under this type be $\epsilon_i^{\text{pos}}$and ϵ_i^{vel}, respectively. Point p is considered feasible for arc i under equipment type k only if $\eta_{ikp} \geq \eta_i, \epsilon_{ikp}^{\text{pos}} \leq \epsilon_i^{\text{pos}}$ and $\epsilon_{ikp}^{\text{vel}} \leq \epsilon_i^{\text{vel}}$ hold. Based on this, the set of feasible points for arc i under equipment type k is defined as $P_{ik}^{\text{fea}} \subseteq P$.

Considering that the same arc may have multiple alternative points for a certain type of equipment, each feasible deployment plan is treated as a candidate combination. For arc i and equipment type $k \in K_i$, the set of candidate combinations is defined as C_{ik}, where each combination $c \in C_{ik}$ corresponds to a specific deployment scheme. For each candidate combination $c \in C_{ik}$, its associated deployment point is denoted as $p_{ikc} \in P$, and its coverage and accuracy metrics are pre-computed offline according to Section 4.

Scheduling decision variables: The scheduling scheme is jointly described by three types of decision variables: arc execution, combination selection, and equipment assignment. The arc execution variable $u_i \in \{0,1\}$ indicates whether arc i is executed. The combination selection variable $z_{ikc} \in \{0,1\}$ indicates whether candidate combination c is selected for arc i under equipment type k. The equipment assignment variable $y_{ikc}^{\text{e}} \in \{0,1\}$ indicates whether equipment e is assigned to combination (i, k, c). The equipment activation variable $x_e \in \{0,1\}$ indicates whether equipment e is activated in the current scheduling round. In the context of dynamic scheduling, to represent whether an original task is adjusted or canceled due to the insertion of a high-priority urgent task, a dynamic adjustment variable $w_i \in \{0,1\}$ is defined to indicate whether arc i is released, reassigned, or canceled during the rescheduling process.

3.3. Constraints

The TT&C resource scheduling model formulated in this paper incorporates the following constraints, covering aspects such as mission integrity, combination feasibility, equipment matching, spatiotemporal compatibility, and dynamic scheduling disturbance control.

$$\sum_{c \in C_{ik}} z_{ikc} = u_i, \quad \forall i \in I, k \in K_i \tag{2}$$

$$\sum_{e \in E} y^e_{ikc} = z_{ikc}, \quad \forall i \in I, \forall k \in K_i, \forall c \in C_{ik} \tag{3}$$

$$y^e_{ikc} \le \mathbb{I}(k_e = k), \quad \forall i \in I, \forall k \in K_i, \forall c \in C_{ik}, \forall e \in E \tag{4}$$

$$y^e_{ikc} \le \mathbb{I}(p^0_e = p_{ikc}), \quad \forall i \in I, \forall k \in K_i, \forall c \in C_{ik}, \forall e \in E^f \tag{5}$$

$$y^e_{ikc} \le \mathbb{I}\left(a_e + \theta_{pq} \le t^s_i\right), \quad \forall i \in I, \forall k \in K_i, \forall c \in C_{ik}, \forall e \in E^m \tag{6}$$

$$\tau_{\text{strip}} + \frac{d_{p_j p_{j+1}}}{v} + \tau_{\text{set}} \le t^s_{j+1} - t^e_j \tag{7}$$

$$y^e_{ikc} \le x_e, \quad \forall i \in I, \forall k \in K_i, \forall c \in C_{ik}, \forall e \in E \tag{8}$$

$$\sum_{i,k,c} \Delta t_i \cdot y^e_{ikc} \le H_e, \quad \forall e \in E \tag{9}$$

$$u_i + w_i \le 1, \quad \forall i \in I \tag{10}$$

where a_e denotes the earliest time at which equipment e becomes available after completing its previous task. Equations (2)–(10) represent the constraints. Equation (2) specifies that if an arc is executed, exactly one candidate combination must be selected for each required equipment type; if the arc is not executed, no combination is selected. Equation (3) states that when a candidate combination is selected, exactly one equipment must be assigned to it; no equipment may be assigned to an unselected combination. Equation (4) ensures that equipment can only undertake TT&C tasks consistent with its type. Equation (5) requires that fixed equipment can only execute task combinations that match its fixed deployment point. Equation (6) indicates that mobile equipment must arrive at the target point and complete setup before the arc's start time to be assigned to that task. For each mobile equipment $e \in E^m$, let $\Gamma_e = \{(i_1, k_1, c_1), (i_2, k_2, c_2), \dots\}$ be its sequence of assigned tasks, ordered by start time. Equation (7) is applied to every consecutive pair in Γ_e, forming a chain of constraints that tracks the equipment's state evolution across the entire sequence. Specifically, the location after task j is p_j, which becomes the origin for the transfer to p_{j+1}; the completion time t^e_j propagates forward through the inequality to determine the earliest start of task $j+1$; and the total mobility distance f_1 in Equation (11) is the sum of all transfer distances $d_{p_j p_{j+1}}$ over Γ_e. Thus, the pairwise constraint applied successively provides a complete representation of location, time, and distance evolution along the task sequence. Equation (8) states that whenever equipment is assigned to any task, its activation status variable must be true. Equation (9) requires that the total duration of tasks assigned to an equipment does not exceed its maximum continuous working time. Equation (10) indicates that during rescheduling, an arc is either retained for execution or adjusted/canceled, but cannot be in both states simultaneously.

3.4. Objective Function

The proposed model aims to optimize the scheduling scheme of TT&C resources and is formulated as a multi-task optimization problem. Within the constraints of limited resources, the model seeks the overall system optimum by comprehensively considering the economic efficiency and mobility cost of the scheduling scheme. Three core optimization objectives are established as follows:

1. Minimize the total mobility distance. This objective aims to reduce fuel consumption, equipment wear, and transportation costs, as well as to avoid unnecessary long-distance transfers, thereby improving economic efficiency:

$$1. \quad \min f_1 = \sum_{e \in E^m} D_e$$

2. Minimize the total number of activated equipment units. This objective aims to reduce the total resource occupancy, including all mobile vehicles and fixed stations, thereby enhancing overall resource utilization:

$$3. \quad \min f_2 = \sum_{e \in E} x_e$$

3. Minimize the number of activated mobile equipment units. This objective focuses specifically on reducing the usage count of mobile equipment (e.g., tracking and control vehicles) to lower mobility costs and scheduling complexity:

$$5. \quad \min f_3 = \sum_{e \in E^m} x_e$$

Inherent conflicts exist among the above three objectives. Minimizing the total mobility distance tends to assign equipment to tasks in close proximity, whereas minimizing the total number of activated equipment units may force equipment to travel long distances—these two objectives constrain each other. Minimizing the number of activated mobile equipment units further reinforces the preference for utilizing fixed equipment, which may lead to concentrated workloads on fixed stations or impose additional mobility burdens on mobile equipment. Together, the three objectives form a vector optimization problem, whose Pareto frontier characterizes the trade-off between resource utilization and mobility cost under different scheduling strategies.

4. Quantitative Calculation of Coverage and Accuracy Constraints

To support the hierarchical modeling concept of "arc–candidate combination–equipment assignment" introduced in Section 3, this study first performs an offline evaluation of the service relationship between each TT&C arc and each candidate point before entering the online scheduling phase. The evaluation consists of two levels: (i) coverage calculation based on geometric visibility, and (ii) accuracy assessment for external-measurement equipment. Telemetry equipment and safety-control equipment require only that the coverage threshold be met to be deemed feasible, whereas radar and optical equipment must satisfy both position and velocity accuracy requirements in addition to the coverage threshold before being included in the candidate combination set.

4.1. Visibility Analysis and Coverage Calculation

Whether a TT&C equipment can serve a given flight arc depends first on whether the equipment has stable visibility to the target during the arc's time window. Therefore, based on the rocket trajectory and the geometric relationship with the TT&C point, this paper performs a visibility judgment for each candidate point and subsequently calculates the coverage.

The time window of arc i is $[t_i^s, t_i^e]$. The window is discretized into N_i sampling instants $\{t_1, t_2, \ldots, t_{N_i}\}$. Let $\mathbf{r}(t_n)$ be the position vector of the rocket at time t_n, and $\mathbf{S}_p$

the coordinate vector of TT&C point p. Then the line-of-sight vector from the point to the target is:

$$\boldsymbol{\rho}_{ip}(t_n) = \mathbf{r}(t_n) - \mathbf{S}_p \tag{14}$$

The corresponding slant range is:

$$d_{ip}(t_n) = \|\boldsymbol{\rho}_{ip}(t_n)\| \tag{15}$$

Further, denote the zenith direction unit vector at point p as $\mathbf{n}_p$. The elevation angle of the target relative to point p can be expressed as:

$$E_{ip}(t_n) = \arcsin\left(\frac{\boldsymbol{\rho}_{ip}(t_n) \cdot \mathbf{n}_p}{d_{ip}(t_n)}\right) \tag{16}$$

When the target's elevation angle is no less than the minimum allowable elevation angle of the equipment, and the target distance does not exceed the equipment's effective operating range, the equipment is considered visible to the target at that instant. The visibility indicator function for point p with respect to arc i is thus defined as:

$$g_{ip}(t_n) = \begin{cases} 1, & E_{ip}(t_n) \geq E^d_{\min} \text{ and } d_{ip}(t_n) \leq D^d_{\max} \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

where $E^d_{\min}$ and $D^d_{\max}$ denote the minimum operational elevation angle and the maximum effective range, respectively, for equipment of type d.

Based on the above, the coverage of point p with respect to arc i is defined as:

$$\eta_{ip} = \frac{1}{N_i}\sum_{n=1}^{N_i} g_{ip}(t_n) \tag{18}$$

This metric reflects the proportion of time within the arc's time window during which the equipment is in an effective observation state.

4.2. Measurement Accuracy Model

For telemetry and safety-control equipment, coverage alone sufficiently reflects their mission support capability. However, for external-measurement equipment such as radar and optical devices, satisfying the visibility constraint is still insufficient to guarantee TT&C quality; the measurement accuracy of target position and velocity must be further evaluated. To this end, this paper adopts the concept of spatial geometric positioning and error propagation for external ballistic measurements, and performs accuracy discrimination for radar-type and optical-type equipment during the candidate combination generation phase.

Radar accuracy measurement: Radar typically provides range R, azimuth A, and elevation E of the target. According to the monostatic positioning relationship, the target's coordinates in the station coordinate system can be expressed as:

$$\begin{cases} X_c = R\cos E\cos A \\ Y_c = R\sin E \\ Z_c = R\cos E\sin A \end{cases} \tag{19}$$

Assume that the standard deviations of the radar's measurements R, A, and E are σ_R, σ_A, and σ_E, respectively. Under the small-error assumption, a first-order error propagation method can be used to estimate the target positioning error. The measurement element covariance matrix is:

$$\mathbf{P}_m = \mathrm{diag}(\sigma_R^2, \sigma_A^2, \sigma_E^2) \tag{20}$$

The validity of the first-order error propagation method relies on the small-error assumption, i.e., the standard deviations of the raw measurements (range R, azimuth A, and elevation E) are sufficiently small relative to their magnitudes so that the higher-order Taylor terms can be neglected when mapping the nonlinear measurement-to-position relationship. In the typical TT&C scenarios considered in this study, radar ranging accu-

racy is on the order of meters (far smaller than the operating range) and angular accuracy is on the order of arcseconds, which well satisfies this small-error condition. The position error covariance matrix is:

$$\mathbf{P}_{\text{pos}} = \mathbf{J}\mathbf{P}_m\mathbf{J}^{\text{T}} \tag{21}$$

where $\mathbf{J}$ is the Jacobian matrix from the measurement elements to the target position. The position accuracy at that instant is defined as:

$$\varepsilon^{\text{pos}}(t) = \sqrt{\text{tr}\left(\mathbf{P}_{\text{pos}}(t)\right)} \tag{22}$$

The maximum position accuracy over all sampling instants within the arc is taken as the position accuracy evaluation metric for that point or combination:

$$\varepsilon_i^{\text{pos}} = \max_{t\in[t_i^s,t_i^e]} \varepsilon^{\text{pos}}(t) \tag{23}$$

For candidate combinations involving joint measurements by multiple radars, the same error propagation method can be applied based on the joint positioning result. If the maximum position error does not exceed the mission-specific threshold, the radar point or combination is judged feasible in terms of position accuracy.

Optical measurement accuracy: Optical equipment typically provides only azimuth and elevation angles of the target, belonging to the classical angle-only measurement system. Since a single optical device cannot independently determine the three-dimensional position of the target, two or more optical devices are usually required for intersection measurement. Suppose an optical combination yields a target position estimate at a given time through two-station or multi-station intersection. Its position accuracy can then be evaluated using the error propagation method as well.

Let the optical measurement vector be $\mathbf{z}^{\text{opt}}$ and its error covariance matrix be $\mathbf{P}_{\text{opt},m}$. The position error covariance matrix of the intersection positioning result can be expressed as:

$$\mathbf{P}_{\text{opt,pos}} = \mathbf{J}_{\text{opt}}\mathbf{P}_{\text{opt},m}\mathbf{J}_{\text{opt}}^{\text{T}} \tag{24}$$

where $\mathbf{J}_{\text{opt}}$ is the Jacobian matrix of the intersection positioning result with respect to the measurement elements. The position accuracy of the optical combination at time t is defined as:

$$\varepsilon_{\text{opt}}^{\text{pos}}(t) = \sqrt{\text{tr}\left(\mathbf{P}_{\text{opt,pos}}(t)\right)} \tag{25}$$

And the maximum value over the arc is taken as the position accuracy evaluation metric of that combination for the arc:

$$\varepsilon_{\text{opt},i}^{\text{pos}} = \max_{t\in[t_i^s,t_i^e]} \varepsilon_{\text{opt}}^{\text{pos}}(t) \tag{26}$$

Velocity accuracy evaluation: Velocity parameters are usually derived from a time-series of positions at consecutive instants. Assume that the position estimates obtained by a candidate point or combination at the sampling instants within the arc are $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_{N_t}$ with sampling interval Δt. The velocity can be approximated by central difference:

$$V(t_n) = \frac{\mathbf{P}_{n+1} - \mathbf{P}_{n-1}}{2\Delta t} \tag{27}$$

Propagating the position error yields the velocity error covariance matrix $\mathbf{P}_{\text{vel}}(t_n)$, and the velocity accuracy at that instant is defined as:

$$\varepsilon^{\text{vel}}(t) = \sqrt{\text{tr}\left(\mathbf{P}_{\text{vel}}(t_n)\right)} \tag{28}$$

The maximum velocity accuracy over all sampling instants within the arc is taken as the velocity accuracy metric for that candidate point or combination:

$$\varepsilon_i^{\text{vel}} = \max_{t_n\in[t_i^s,t_i^e]} \varepsilon^{\text{vel}}(t_n) \tag{29}$$

If it does not exceed the velocity accuracy threshold specified for the mission, the point or combination is considered feasible in terms of velocity accuracy.

Based on the above analysis, this paper adopts differentiated feasibility criteria for different equipment types: for telemetry and safety-control equipment, only coverage is used as the feasibility criterion; for radar and optical equipment, candidate points or combinations are required to simultaneously satisfy coverage, position accuracy, and velocity accuracy. That is, only when the following conditions are met:

$$\eta_{ikc} \geq \eta_i, \varepsilon_{ikc}^{\mathrm{pos}} \leq \varepsilon_i^{\mathrm{pos}}, \varepsilon_{ikc}^{\mathrm{vel}} \leq \varepsilon_i^{\mathrm{vel}} \tag{30}$$

is the candidate combination c included in the set of feasible combinations C_{ik} for arc i under equipment type k.

5. Multi-Objective Scheduling Optimization Based on Adaptive NSGA-II

The problem addressed in this study is essentially a combinatorial optimization problem characterized by strong constraints, multiple objectives, and a dominance of discrete variables. Since there exist significant conflicts among the objectives—for example, reducing the number of mobile equipment units often increases the maximum travel distance of a single mobile unit, while controlling rescheduling disturbances may suppress new global optimization opportunities—it is difficult to obtain stable solutions by using a single-objective weighted approach. NSGA-II, with its fast non-dominated sorting and crowding distance preservation strategy, can effectively handle multi-objective Pareto optimization problems. To adapt it to the specific scenario of this study, we customize and improve the standard NSGA-II.

5.1. Adaptive NSGA-II Algorithm Procedure

The procedure of the multi-objective scheduling optimization based on adaptive NSGA-II is shown in the table below (Algorithm 1). The overall concept is as follows: first, initialize the population based on the mathematical model established in Section 3 and the set of candidate combinations generated in Section 4; then decode each individual to construct a specific equipment assignment scheme and compute the three objective function values; on this basis, perform fast non-dominated sorting, crowding distance calculation, selection, crossover, and mutation operations, iteratively updating the population. During the evolution, adaptively adjust the population size according to improvements in the Pareto front, and terminate the iteration early when the population shows no significant improvement over a long period. Finally, output a set of feasible non-dominated scheduling schemes for subsequent invocation by the dynamic rescheduling module.

Algorithm 1: Adaptive NSGA-II for multi-task space TT&C resource scheduling.

Input: Task arc set I; equipment set E; candidate combination set C; maximum number of generations $G_{\max}$; initial population size N_0; crossover probability p_c; mutation probability p_m; minimum and maximum population sizes $N_{\min}$ and $N_{\max}$; hypervolume improvement thresholds θ_1 and θ_2; early stopping threshold θ_{stop}; patience parameter L

Output: Pareto-optimal scheduling solution set P^*

Initialization:

1. Initialize $g \leftarrow 0$, $N \leftarrow N_0$,and stagnation counter $cnt \leftarrow 0$
2. Generate the initial population P_g and create an LRU decoding cache Cache
3. Evaluate each individual in P_g by chromosome decoding, feasibility repair, and objective computation
4. Store newly decoded individuals in Cache
5. Perform fast non-dominated sorting and crowding-distance calculation on P_g

6. Compute the hypervolume value HV_g
7. While $g < G_{\max}$:
8. Select parent individuals from P_g by tournament selection
9. Generate an offspring population Q_g of size N by crossover and mutation
10. for each individual $y \in Q_g$ do
11. if y exists in Cache then
12. retrieve its feasibility and objective values from cache
13. else
14. decode y according to the task arc–equipment type–candidate combination structure
15. assign equipment using preference-guided selection
16. perform feasibility repair and objective evaluation
17. store the result in Cache
18. end if
19. End for
20. $R_g \leftarrow P_g \cup Q_g$
21. Perform fast non-dominated sorting and crowding-distance calculation on R_g
22. Generate P_{g+1} by environmental selection
23. Compute HV_{g+1}
24. $\Delta\mathrm{HV}_g = \left(\mathrm{HV}_{g+1} - \mathrm{HV}_g\right)/\max\left(\mathrm{HV}_g, \varepsilon\right)$
25. If $\Delta\mathrm{HV}_g > \theta_1$ then $N \leftarrow \min(N + \delta_1, N_{\max})$
26. else if $\Delta\mathrm{HV}_g < \theta_2$ then $N \leftarrow \max(N - \delta_2, N_{\min})$
27. If $\left|\Delta\mathrm{HV}_g\right| < \theta_{\mathrm{stop}}$ then $cnt \leftarrow cnt + 1$ else $cnt \leftarrow 0$
28. If $cnt \geq L$ then break
29. $P_g \leftarrow P_{g+1}$; $\mathrm{HV}_g \leftarrow \mathrm{HV}_{g+1}$; $g \leftarrow g + 1$
30. End while
31. Extract all non-dominated individuals in P_g as P^*
32. Return P^*

5.2. Chromosome Encoding and Decoding

Given the large number of equipment types, strong temporal coupling among tasks, and the large set of candidate combinations in the space TT&C resource scheduling problem, this paper adopts an individual representation that combines equipment activation encoding with rule-driven decoding. The chromosome only determines which equipment units are allowed to be used in the current scheduling round, while the specific assignment of equipment to arcs is generated by the decoding process based on heuristic rules. This reduces the encoding dimension, decreases the number of infeasible solutions, and facilitates the integration of various constraints into the decoding procedure.

The number of fixed equipment units is denoted as $|E^f|$, and the number of mobile equipment units as $|E^m|$. A chromosome is represented as a binary vector of length $\left|E^f\right| + |E^m|$:

$$X = \left\{x_1, x_2, \ldots, x_{|E^f|+|E^m|}\right\} \tag{31}$$

where each gene is defined as:

$$x_k = \begin{cases} 1, & \text{equipment is activated in the current scheduling round} \\ 0, & \text{otherwise} \end{cases} \tag{32}$$

The first $\left|E^f\right|$ genes correspond to the activation status of fixed equipment, and the remaining $|E^m|$ genes correspond to mobile equipment. Thus, an individual essentially describes a decision on which equipment units are allowed to be used, rather than directly specifying the assignment of equipment to each task arc. The specific combination selection and equipment assignment at the arc level are generated by the decoding process.

This study adopts a stepwise decoding strategy that proceeds in task order and time sequence, attempts combinations in descending order of quality, and assigns equipment according to a preference rule. During decoding, the set of activated fixed equipment and the set of activated mobile equipment are first determined from the chromosome. Then, each task is processed in ascending order of its arc start time. For each equipment type required by an arc, the corresponding candidate combinations are traversed in order, and an attempt is made to find a feasible equipment unit among the currently activated equipment that satisfies the time compatibility, point matching, and mobility reachability constraints. If a candidate combination is successfully assigned, the demand for that equipment type is considered satisfied. If all candidate combinations fail, the arc is considered infeasible, and a penalty is imposed on the individual.

This approach shifts the handling of complex constraints from the encoding layer to the decoding layer, embedding a heuristic assignment strategy. To improve decoding efficiency, the candidate combinations for an arc are not randomly attempted; instead, a composite score is constructed offline based on coverage prediction and accuracy metrics, and the combinations are sorted in descending order of this score. For a candidate combination c of arc s_i, the composite score is formulated as:

$$\mathrm{Score}(c) = \frac{\mathrm{Cov}(c)}{1 + \frac{\mathrm{Acc}(c)}{\kappa}} \tag{33}$$

where $\mathrm{Cov}(c)$ represents the coverage capability of the combination, $\mathrm{Acc}(c)$ represents the position accuracy metric, and κ is a normalization weight set to 100 in this study. For arcs that only require coverage (e.g., telemetry equipment) and for positioning equipment, the rule simplifies to sorting solely by coverage. During decoding, combinations with higher scores are attempted first, steering the algorithm toward searching in feasible regions with higher service quality.

5.3. Adaptive Population and Early Stopping Strategy

Standard NSGA-II adopts a fixed population size, which may lead to insufficient population diversity or wasteful computational resources in the later stages of evolution. To address this issue, this paper introduces an adaptive population adjustment mechanism and an early stopping strategy based on the evolution information of the hypervolume (HV), so as to dynamically control computational cost while ensuring convergence performance.

Let the population size at generation t be N_t, and the corresponding Pareto front be P_t; the hypervolume value HV_t is then computed. To measure the convergence speed, the average HV improvement rate over a sliding window of length ω (set to 5 in this paper) is defined as:

$$\Delta_t = \frac{1}{\omega} \sum_{k=t-\omega+1}^{t} \frac{\mathrm{HV}_k - \mathrm{HV}_{k-1}}{\mathrm{HV}_{k-1} + \psi} \tag{34}$$

where ψ is a very small positive number to prevent division by zero. When Δ_t drops significantly (e.g., below 50% of the value in the previous window), the population is considered to have fallen into local convergence; in this case, the population size is increased to introduce diversity. Conversely, if the convergence speed is relatively high, the population size may be moderately reduced to decrease redundant computations. The adjustment rule is as follows:

$$N_{t+1} = \begin{cases} \min(N_t \times 1.2, N_{\max}), & \Delta_t < 0.5 \cdot \Delta_{t-\omega} \\ \max(N_t \times 0.9, N_{\min}), & \Delta_t > 2.0 \cdot \Delta_{t-\omega} \\ N_t, & \text{otherwise} \end{cases} \tag{35}$$

where $N_{\min} = 20$, $N_{\max} = 100$, and the initial population size $N_0 = 30$. The adjustment is performed every 5 generations to avoid destabilizing the evolution due to frequent changes.

To automatically terminate the evolution when no significant improvement is observed, an early stopping patience parameter $p = 10$ is defined. Let HV_{best} be the best hypervolume value observed so far. If the relative improvement of HV_{best} remains below the threshold $\delta = 0.001$ for p consecutive generations, and the ratio of the standard deviation to the mean of the HV values over the most recent p generations (coefficient of variation) is less than the convergence threshold $\theta = 0.01$, then the algorithm is determined to have converged and the iteration is terminated early. This mechanism avoids unnecessary computation, especially in large-scale task scenarios, thereby significantly reducing solution time.

In summary, the key parameters of the adaptive population adjustment and early stopping mechanism are configured as follows. The sliding window length ω for HV-based adaptive adjustment is set to 5 generations. The initial population size N_0 is set to 30, with a minimum population size $N_{\min} = 20$ and a maximum population size $N_{\max} = 100$. Adjustments are performed every 5 generations to avoid destabilizing the evolution. For early stopping, the patience parameter p is set to 10 generations, the HV improvement threshold δ is set to 0.001, and the convergence threshold θ is set to 0.01. All these parameter values are consistently applied in the simulation experiments reported in Section 7.

5.4. Decoding Cache

Chromosome decoding is the most time-consuming step in NSGA-II. Since genetic operations generate many repeated or similar chromosomes, this paper introduces a decoding cache based on the least recently used (LRU) strategy to avoid redundant decoding of identical chromosomes.

The tuple form of the chromosome binary vector x is used as the key to cache its decoding results, including the equipment assignment scheme, objective function values, constraint violation values, etc. The maximum cache capacity is set to $C_{\max} = 10000$. When the cache is full, the least recently used entry is evicted. Before each decoding operation, the cache is queried; if a hit occurs, the result is returned directly; otherwise, a full decoding is performed and the result is stored in the cache.

Let the total number of decoding operations be N and the number of cache hits be H. The effective hit rate is defined as:

$$\eta = \frac{H}{N} \tag{36}$$

In the simulation experiments, this strategy significantly reduces decoding time, especially when the population size is large and the number of generations is high. Under such conditions, the cache hit rate can reach 20–40% or more, markedly improving the overall solution efficiency.

5.5. Preference-Based Equipment Selection

To enhance the rationality and engineering applicability of equipment assignment during decoding, this paper designs a comprehensive equipment preference score, which prioritizes equipment with higher scores while satisfying the constraints.

Considering factors such as equipment reliability, age, operating cost, mobility transportation cost, and maximum continuous working time, the comprehensive equipment preference score S_e is defined as:

$$S_e = \omega_1 \cdot \tilde{a}_e + \omega_2 \cdot \tilde{r}_e + \omega_3 \cdot \tilde{t}_e + \omega_4 \cdot \tilde{o}_e + \omega_5 \cdot \tilde{l}_e \tag{37}$$

where $\tilde{a}_e$ is the normalized age indicator (shorter age is better); $\tilde{r}_e$ is the normalized operating cost indicator (lower cost is better); $\tilde{t}_e$ is the normalized transportation cost indicator (lower cost is better); $\tilde{o}_e$ is the normalized maximum continuous working time indicator (longer time is better); $\tilde{l}_e$ is the reliability indicator; and the weights ω_{1-5} are the weighting coefficients, which are set to $(0.2, 0.2, 0.1, 0.2, 0.3)$ in this study, reflecting a higher emphasis on reliability and operational capability.

In the decoding phase, for each candidate combination, if multiple feasible equipment units are available, the following selection rules are applied:

Fixed equipment: Among the candidate fixed equipment units that satisfy point matching and time non-conflict requirements, the one with the highest comprehensive preference score is selected.

Mobile equipment: Among the candidate mobile equipment units that satisfy mobility reachability and mission time window constraints, the one with the highest preference score is selected.

This strategy ensures that the scheduling solution is not only feasible in terms of constraints but also tends to select equipment with better performance, lower cost, and higher reliability, thereby improving the engineering feasibility and practical execution effectiveness of the scheme.

6. Dynamic Rescheduling Mechanism Based on Priority Rules

In flight-priority launch scenarios, the insertion of urgent tasks is the norm rather than the exception. For example, the temporary addition of key tracking arcs, adjustments to launch plans, or sudden equipment failures can render the existing scheduling scheme invalid. An initial scheduling scheme obtained solely through static optimization is often insufficient to meet real-time requirements. Therefore, this study proposes a dynamic rescheduling mechanism based on priority rules, which ensures the completion of high-priority tasks while minimizing disruptions to the original schedule.

6.1. Description of Dynamic Scenarios

During the execution of flight-priority launch missions, dynamic events can be diverse. This paper focuses on the most typical scenario: the insertion of urgent tracking and control (TT&C) tasks. Suppose a static optimal scheduling scheme S_0 has been obtained at time t_0. When an urgent task arrives, the system needs to add a set of new tracking arcs ΔI to the original task set I. These arcs have specific service time windows, equipment-type requirements, and coverage/precision constraints. Unlike the static phase, the objective in the dynamic phase is no longer to construct a global schedule from scratch, but to derive an updated schedule S_1 based on the existing scheme S_0, such that the following requirements are simultaneously met:

1. Emergency tasks are given priority assurance;
2. Already scheduled high-priority tasks are disturbed as little as possible;
3. The modification to the original baseline schedule is kept as small as possible;
4. The state transitions of mobile equipment still satisfy the constraints of dismantling, transfer, and setup preparation.

Therefore, dynamic rescheduling is essentially a local combinatorial optimization problem oriented toward "minimum-disruption insertion." The difficulty lies not in refinding the global optimum, but in identifying releasable resources, controlling cascading conflicts, and completing rapid backfilling within a limited time.

6.2. Dynamic Rescheduling Algorithm Procedure

To achieve a low-disturbance and highly agile dynamic response, this section proposes a rescheduling algorithm that combines rolling horizon search with priority rules. The algorithmic procedure is shown in Algorithm 2.

Algorithm 2: Priority-rule-based rolling rescheduling for emergency task insertion.

Input: Baseline schedule S_0; Feasible combination set of original tasks C_0; Set of emergency task arcs ΔI; Feasible combination set of emergency tasks C_ε; Initial insertion time T_0; Rolling step ΔT; Maximum number of iterations $K_{\max}$

Output: Updated dynamic schedule S_1

Initialization:

1. Read the baseline schedule and construct timelines for fixed and mobile equipment
2. Initialize equipment states (current position, earliest available time, cumulative travel distance)
3. Generate the set of emergency arcs ΔI from the emergency task plan, load their feasible combinations C_ε
4. For $k = 0$ to $K_{\max}$ do
5. $T_k \leftarrow T_0 + k \cdot \Delta T$
6. Compute absolute execution time windows for each emergency arc based on T_k
7. Copy the current schedule state as a temporary scheduler for this attempt
8. Initialize a new assignment set $A_k = \emptyset$ and a adjusted task queue $Q_k = \emptyset$
9. for each emergency arc $i \in \Delta I$ in order of start time do
10. for each required equipment type $d \in D_i$ do
11. Iterate over candidate combinations in C_ε in descending order of score
12. if a directly assignable equipment exists then
13. Complete the assignment and update A_k
14. else if an equipment is available that conflicts only with lower-priority tasks then
15. Release the conflicting lower-priority tasks and their subsequent dependent tasks, add them to Q_k
16. Complete the assignment for the emergency task and update A_k
17. else: mark insertion failure for the current time T_k and break out of this attempt
18. If all emergency arcs are successfully inserted in this round then
19. for each adjusted task $q \in Q_k$ do
20. Retrieve its original feasible combinations from C_0 and attempt local reassignment
21. if reassignment fails then mark T_k as infeasible
22. if all adjusted tasks are successfully reassigned then return the current schedule as S_1
23. If all rolling time points fail, return "Dynamic rescheduling infeasible"

The rolling horizon search explores insertion time points $T_k = T_0 + k \cdot \Delta T$, where $\Delta T = 300$ s (5 min) and $K_{\max} = 12$, covering a total search window of 60 min from the initial emergency task request time. This range is selected based on typical operational practice, where emergency tasks are expected to be inserted within one hour of their announcement. In each attempt, the algorithm copies the current schedule state and tries to insert all emergency arcs sequentially; if any arc fails, the attempt is discarded and the search moves to the next time point.

6.3. Priority-Driven Insertion Strategy

When an emergency task arrives, the system first searches for resources that can be directly inserted into the current equipment timelines. For fixed equipment, it is only necessary to satisfy equipment-type matching, point consistency, and non-conflicting time windows. For mobile equipment, the spatiotemporal reachability to the target point must also be satisfied. Let the current point of mobile equipment $e \in E$ be p_e^{cur}, its earliest available time be t_e^{ava}, and the target point be p. The reachability condition is then:

$$7. \quad t_e^{\text{ava}} + \tau_e(p_e^{\text{cur}}, p) \le t_s^i$$

where $\tau_e(p_e^{\text{cur}}, p)$ includes the dismantling, transfer, and setup preparation times.

When a candidate equipment unit is already undertaking an original task during the time window of the emergency task, a resource conflict occurs. This paper adopts a priority-based conflict resolution rule: the resource is allowed to be released only if the priority of the emergency task is higher than that of the conflicting task, i.e.:

$$P_i > P_j, \forall j \in \Phi(e) \tag{39}$$

where $\Phi(e)$ denotes the set of tasks conflicting with emergency task i on equipment e. To maintain equipment state consistency, this study releases from the earliest conflict moment onward all subsequent dependent tasks on that equipment and adds them to a queue for backfilling. This strategy ensures the responsiveness of high-priority tasks while avoiding state discontinuity in the equipment timeline caused by local deletions.

6.4. Priority Task System

To support conflict resolution and backfilling decisions in dynamic rescheduling, this paper constructs a discrete task priority system. While the arc type (critical or general) defines the service-quality requirements (coverage and accuracy thresholds), the task priority system described here governs the order of resource allocation and preemption when conflicts occur during dynamic rescheduling. These two classifications serve different purposes and coexist in the overall scheduling framework. Let the priority of task i be denoted as follows:

$$P(i) \in \{1, 2, 3, 4, 5\} \tag{40}$$

A larger priority value indicates a higher task importance and a stronger right to resource preemption. The priorities of original tasks are predetermined by the mission plan, while those of emergency tasks are determined by the dynamic event input.

Under this system, resource conflict adjustments follow a strict partial order: a higher-priority task may preempt conflicting resources used by lower-priority tasks; tasks of the same priority are not forcibly replaced, and lower-priority tasks cannot adversely affect the resource allocation of higher-priority tasks. When two emergency tasks of equal (highest) priority compete for the same resource, the algorithm resolves the conflict by: (i) giving precedence to the task with the earlier start time (earliest-deadline-first); and if the start times are identical, (ii) using a tie-breaker based on the task's total coverage requirement, giving priority to the task with the higher coverage threshold. Both tasks are never forcibly preempted by each other; instead, the algorithm attempts to insert them sequentially in the order determined by the above rules, and if insertion of the second task fails, a rolling search for an alternative time slot is triggered.

This priority system provides a unified conflict resolution criterion for dynamic scheduling, endowing the resource concession process with a clear hierarchical structure, thereby contributing to the stability and interpretability of scheduling decisions in complex dynamic environments.

7. Simulation Experiments and Result Analysis

To validate the effectiveness of the proposed dynamic TT&C resource scheduling method integrating priority rules with adaptive NSGA-II, this chapter constructs simulation experiments based on a typical commercial launch scenario. First, the experimental scenario and parameter settings are introduced. Second, from the perspective of static

multi-objective optimization, the convergence performance, solution set quality, and acceleration effect of the decoding cache mechanism of the adaptive NSGA-II algorithm are evaluated. Third, for emergency task insertion scenarios, the performance of the priority-rule-based rolling rescheduling mechanism in guaranteeing high-priority tasks and controlling scheduling disturbances is verified. Finally, a comprehensive discussion is presented based on the experimental results.

7.1. Experimental Scenario and Parameter Settings

A typical low-latitude commercial launch site is selected as the simulation background, with the launch point coordinates set to (28.39° N, 80.61° W). The TT&C network is configured with reference to the actual layout of the Eastern Range of the United States, comprising a total of 28 candidate TT&C points. The longitude, latitude, and altitude of each point are shown in Table 1.

Table 1. TT&C point configuration.

Point No.	Latitude (°N)	Longitude (°W)	Altitude (m)	Point No.	Latitude (°N)	Longitude (°W)	Altitude (m)
1	28.3922	80.6077	5	15	26.5000	77.0000	5
2	28.4800	80.5700	10	16	25.0000	76.5000	3
3	28.4200	80.6500	3	17	23.5000	75.5000	2
4	28.3500	80.6800	8	18	21.5000	72.0000	0
5	27.8500	80.4800	15	19	28.8000	82.0000	30
6	27.2000	80.2500	12	20	29.6000	82.5000	40
7	26.5500	80.0500	8	21	30.5000	83.0000	50
8	25.7500	80.1800	5	22	27.5000	81.5000	35
9	29.2000	81.0000	20	23	26.8000	82.0000	25
10	30.3000	81.4000	18	24	20.0000	70.0000	0
11	31.1500	81.3500	10	25	18.0000	68.0000	0
12	28.9000	78.5000	10	26	32.0000	75.0000	0
13	29.5000	76.0000	0	27	33.0000	72.0000	0
14	30.0000	74.0000	0	28	34.0000	70.0000	0

The TT&C resources consist of two categories: fixed equipment and mobile equipment. A total of 54 equipment units are configured, including 25 fixed units and 29 mobile units, covering radar, optical, telemetry, and safety-control types. Mobile equipment is initially deployed at a preset base and can transfer between different points, with an average transfer speed of 72 km/h. The setup time for continuous operation at the same point is set to 60 s; when transferring between different points, additional dismantling, transfer, and setup preparation times are considered, where the dismantling time is 1 h and the setup preparation time is 2 h. The maximum operating range, minimum elevation angle, and quantity configuration for each equipment type are shown in Table 2.

Table 2. Equipment types and performance parameters.

Equipment Type	Quantity (Fixed/Mobile)	Operating Range	Min. Elevation Angle	Ranging Accuracy	Angle Accuracy
Class A Radar	8/7	2000 km	3°	5 m	50″
Class B Radar	3/5	1800 km	3°	10 m	150″
Class A Optical	2/5	1500 km	2°	-	10″
Class B Optical	2/3	1200 km	2°	-	30″
Telemetry	7/5	1400 km	3°	-	
Safety Control	3/4	1000 km	3°	-	

To examine the adaptability of the algorithms under different task scales, three static task scenarios are constructed, as shown in Table 3. The flight process of each rocket is

divided into five or sic TT&C arcs, each with a specific time window, equipment-type requirements, coverage threshold, accuracy threshold, and task priority. Rocket trajectories are generated using Systems Tool Kit (STK, version 12.4) with a sampling interval of 1 s, The launch point is fixed at (28.39° N, 80.61° W), covering four representative azimuths: due east (90°), due north (0°), due west (270°), and northeast (45°). Three static scenarios (S1–S3) are constructed by scaling the number of launches from four to eight to 12, with staggered launch times generated within the 5-day scheduling period. The flight process of each rocket is divided into five or six TT&C arcs, each with a specific time window, equipment-type requirements, coverage threshold, accuracy threshold, and task priority.

Candidate combinations are constructed through a two-stage offline screening. First, visibility is evaluated using Equations (14)–(18); points with coverage above 0.8 (critical arcs) or 0.7 (normal arcs) are retained. Second, for radar and optical equipment, position and velocity accuracies are evaluated using Equations (19)–(29); combinations satisfying both accuracy thresholds (50 m/10 m/s for critical arcs, 60 m/20 m/s for normal arcs) are retained. Telemetry and safety-control equipment require only coverage screening. The coverage and measurement accuracy of each point with respect to each arc are calculated according to the method described in Section 4.

Table 3. Static task scenarios.

Scenario ID	Number of Rockets	Total Arcs	Scheduling Period	Description
S1	4	23	5 days	Small-scale baseline scenario
S2	8	46	5 days	Medium-scale scenario
S3	12	69	5 days	High-density launch scenario

The launch times of the rockets are randomly generated in a stratified manner within the 5-day scheduling period. As the scenario scale increases, the average interval between tasks gradually decreases, thereby intensifying resource competition among different tasks. Arc types are divided into critical and normal categories, with corresponding coverage and accuracy thresholds given in Table 4.

Table 4. Arc coverage and accuracy thresholds.

Arc Type	Coverage Threshold	Position Accuracy Threshold	Velocity Accuracy Threshold
Critical	0.8	50m	10m/s
Normal	0.7	60m	20m/s

Based on the static scheduling schemes, emergency task insertion scenarios are further constructed to validate the effectiveness of the priority-rule-based dynamic rescheduling mechanism. Emergency tasks are assigned the highest priority with equipment requirements including radar, telemetry, optical, or combinations thereof; the coverage threshold is set to 0.90, and the position and velocity accuracy thresholds are kept consistent with those of critical arcs.

Three dynamic scenarios, denoted D1, D2, and D3, are defined as shown in Table 5. Each dynamic scenario consists of one emergency rocket mission, but the number of arcs gradually increases, reflecting an increasing complexity of the emergency task. All emergency tasks are given the highest priority to reflect the scheduling requirement of emergency response and priority assurance.

Table 5. Dynamic task scenarios.

Scenario ID	Number of Rockets	Total Arcs
D1	1	2

D2	1	4
D3	1	6

7.2. Baseline Algorithms and Evaluation Metrics

To comprehensively evaluate the performance of the proposed method in static multi-task scheduling, experiments are designed from two aspects: algorithm comparison and metric evaluation. In the static scheduling stage, the focus is on the improvement of adaptive NSGA-II over standard NSGA-II in terms of solution efficiency, convergence performance, and solution set quality. To validate the effectiveness of each component of the proposed adaptive NSGA-II described in Section 5, the following static comparison algorithms are set up.

1. Standard NSGA-II: The classical NSGA-II framework is used for solution, retaining the same chromosome encoding, candidate combination decoding, crossover probability, and mutation probability as the proposed method.
2. NSGA-II + Cache: Based on the standard NSGA-II, the decoding cache mechanism and equipment preference assignment rules are retained, but the HV-evolution-based adaptive population size adjustment and early stopping are removed; a fixed population size and fixed maximum number of generations are always used.
3. NSGA-II + Adaptive: Based on standard NSGA-II, HV-evolution-based adaptive population adjustment and early stopping are introduced, but the decoding cache is not enabled.
4. Adaptive-NSGA-II: The full method integrates decoding cache, adaptive population adjustment, early stopping, and preference-based equipment selection rules, constituting the complete static multi-objective scheduling solution method of this paper.
5. GA-PE: A single-objective genetic algorithm with population perturbation and elimination strategies [11], using weighted aggregation (0.4, 0.3, 0.3) for the three objectives.

The five methods share the same chromosome encoding, candidate combination set, and constraint handling approach to ensure fairness of comparison. Unless otherwise specified, each algorithm uses binary equipment activation encoding, with population size 30, crossover probability 0.9, mutation probability 0.1, and maximum number of generations 100. For the adaptive population methods, the initial population size is set to 30, the minimum population size to 20, the maximum to 100, the sliding window length to 5, and the early stopping patience parameter to 10 generations. To ensure statistical reliability, all experiments are repeated 30 times with different random seeds, and results are reported as mean ± standard deviation.

In the dynamic rescheduling experiments, the following baseline strategy is set:

1. Fixed-time direct insertion strategy: When an emergency task arrives, only a single direct insertion attempt is performed at its initial request time, without any further rolling search or local adjustment. This method can be regarded as the simplest online insertion baseline.
2. Non-preemptive rolling insertion strategy: This strategy performs the same rolling horizon search as the proposed method (5 min step, up to 60 min horizon) but does not release any conflicting tasks.
3. Greedy repair heuristic: Upon insertion failure, this method releases all conflicting tasks (regardless of priority) and then attempts to reassign them in a greedy manner using available slack time.
4. Priority-rule-based rolling rescheduling strategy: When insertion fails at the initial time, this method continues to roll forward along the timeline to search for subsequent feasible insertion windows; when conflicts occur, priority rules are used for conflict resolution and local adjustments are triggered when necessary.

All algorithms are compared under identical maximum generation limits (100 generations). The proposed Adaptive-NSGA-II may terminate early when no significant HV improvement is observed, which is part of its design to reduce unnecessary computation. Runtime reductions are therefore reported under equal maximum generation limits, with the actual savings reflecting the efficiency gains of the proposed method.

To evaluate the performance of the proposed method comprehensively from multiple dimensions, separate evaluation metrics are defined for static and dynamic scheduling.

Static scheduling stage: Seven metrics are selected to evaluate algorithm performance in terms of feasibility and optimality: (1) task completion rate—directly reflects the executability of the scheduling scheme; (2) total number of activated equipment units—smaller values indicate more compact resource allocation; (3) number of activated mobile equipment units—smaller is better; (4) total mobility distance—reflects the overall mobility cost; (5) maximum mobility distance—smaller values indicate lower per-unit mobility pressure, which is more favorable for engineering implementation; (6) solution time—total computational time from initialization to output of the non-dominated solution set; and (7) hypervolume (HV)—comprehensively measures the convergence and diversity of the Pareto solution set in the objective space; a larger HV indicates that the solution set is closer to the ideal front and has wider coverage. The HV metric is computed using a fixed reference point [10,000,000, 100, 100], consistently applied across all algorithms. No explicit normalization to [0,1] is applied; the fixed reference point is chosen to be sufficiently large to encompass all observed objective ranges.

Dynamic scheduling stage: Four metrics are selected to assess the agility and low-disturbance characteristics of the rescheduling mechanism: (1) emergency task insertion success rate—the proportion of emergency tasks successfully inserted in a given dynamic scenario; this directly reflects the emergency response capability of the dynamic scheduling mechanism; (2) newly added equipment assignments—the number of additional equipment assignment entries required to accommodate the emergency task, measuring the extra resource investment needed for dynamic insertion; (3) newly added total mobility distance—the sum of all additional mobility distances incurred by mobile equipment to complete the emergency task insertion, relative to the baseline scheduling scheme. This metric measures both the extra mobility cost and the disturbance to the original schedule introduced by dynamic rescheduling; and (4) rescheduling runtime—the total computational time consumed by the rolling horizon search and conflict resolution process, reflecting the real-time responsiveness of the algorithm.

7.3. Experimental Results

7.3.1. Experimental Results of Static Scenarios

To ensure statistical reliability and to account for the inherent stochasticity of the metaheuristics, all experiments are repeated 30 times with different random seeds. All numerical results are reported as mean ± standard deviation over these independent runs. This multi-run setup provides a solid basis for comparing algorithm performance and assessing the variability of the solutions.

From the experimental results, it can be seen that all four multi-objective algorithms obtain feasible scheduling schemes with a task completion rate of 100% in the three scenarios, indicating that the "candidate combination pre-screening + rule-driven decoding" framework constructed in this paper can effectively handle coverage, accuracy, and equipment spatiotemporal compatibility constraints. In contrast, the single-objective GA-PE achieves task completion rates of 87.0%, 56.5%, and 64.34% in S1, S2, and S3, respectively (with a standard deviation of 17.53% in S3). This decline is attributed to its weighted aggregation strategy, which over-emphasizes equipment minimization (combined weight 0.6) at the expense of task completeness when resource conflicts intensify. On this basis,

the main differences among the five algorithms lie in the trade-offs among solution time, HV metric, and final objective values.

Overall, the proposed Adaptive-NSGA-II exhibits higher solution efficiency in all three scenarios S1, S2, and S3. Compared with the standard NSGA-II, the proposed method significantly reduces solution time while maintaining a 100% task completion rate; compared with NSGA-II + Adaptive, it further demonstrates the acceleration effect of the decoding cache mechanism for complex constrained scheduling problems. On the other hand, the standard NSGA-II obtains better single-objective values and higher HV metrics in several scenarios, but at the cost of longer iteration processes and higher overall computational time. This indicates that the advantage of the proposed method lies mainly in its fast feasible solution capability for engineering applications, while the standard NSGA-II has a more thorough global search space when given sufficient computational budget.

Looking at different scale scenarios, as the task scale increases from S1 to S3, the solution difficulty of all algorithms rises significantly, manifested by the increased solution time, expanded mobility distances, and intensified resource competition. Nevertheless, Adaptive-NSGA-II is still able to terminate the search early with fewer actual iterations in S2 and S3, demonstrating good scalability and computational stability.

Figure 1 shows the variation in solution time of different algorithms in three static scenarios. It can be observed that as the task scale increases from S1 to S3, the solution time of each algorithm generally shows an upward trend, which is consistent with the increase in the number of arcs, the intensification of resource conflicts, and the higher complexity of decoding.

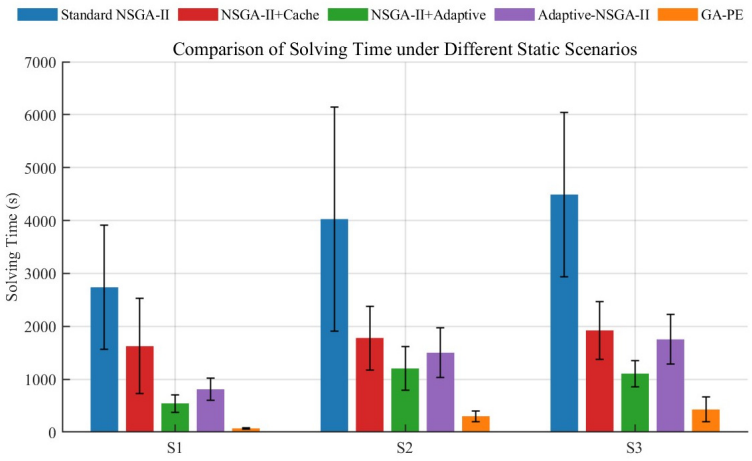

Figure 1. Comparison of solution time of different algorithms in scenarios S1, S2, and S3. Error bars represent the standard deviation over 30 independent runs.

Across all scenarios, GA-PE achieves the shortest mean computation times: 71.88 ± 12.53 s in S1, 300.57 ± 100.29 s in S2, and 427.80 ± 232.52 s in S3. However, as shown above, this extreme efficiency comes at the cost of substantially lower task completion rates, confirming a clear trade-off between computational speed and solution completeness.

In scenario S1, Adaptive-NSGA-II achieves a mean solution time of 809.94 ± 207.41 s, significantly lower than the 2738.10 ± 1171.52 s of the standard NSGA-II, a reduction of about 70.4%; in scenario S2, Adaptive-NSGA-II solves in 1501.80 ± 465.30 s, a reduction of about 62.7% from the 4026.88 ± 2114.82 s of the standard NSGA-II; in the high-density scenario S3, Adaptive-NSGA-II takes 1752.98 ± 470.75 s, while the standard NSGA-II takes 4490.20 ± 1553.01 s, a reduction of about 61.0%. These results show that the computational

efficiency advantage of the proposed method is statistically significant across all scenarios, although the variability also increases with problem scale.

Further comparing the three improved algorithms, the solution time of NSGA-II + Adaptive is higher than that of Adaptive-NSGA-II in all three scenarios, while the time performance of NSGA-II + Cache is relatively close to that of Adaptive-NSGA-II. This indicates that in the current problem, the decoding cache mechanism plays a direct and significant role in reducing solution time, while the adaptive population adjustment and early stopping strategies mainly control overall computational overhead by reducing ineffective evolution generations. The combination of the two mechanisms enables the proposed method to achieve good time efficiency across different scale scenarios.

Figure 2 presents a comparison of the HV metrics of different algorithms under scenarios S1, S2, and S3. GA-PE is not included in the HV comparison, as it is a single-objective method that does not produce a Pareto front. It should be noted that the final HV value of each algorithm is taken as the last value of its HV historical sequence for analysis. Experimental results show that the HV values obtained by all algorithms are positive and exhibit a clear convergence process, effectively reflecting the coverage and convergence quality of the non-dominated solution sets in the objective space.

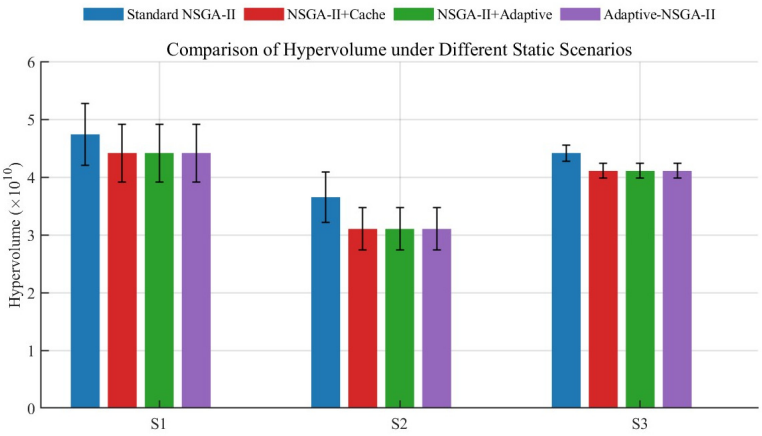

Figure 2. Comparison of HV metrics of different algorithms in scenarios S1, S2, and S3. Error bars represent the standard deviation over 30 independent runs.

From an engineering application perspective, although the HV of Adaptive-NSGA-II is slightly lower than that of the standard NSGA-II, its corresponding solution time is significantly shorter, and it still maintains a high quality of non-dominated solutions. This demonstrates that the proposed method achieves a balance between "solution set quality" and "computational timeliness" that is more suitable for dynamic scheduling scenarios. Notably, the HV of Adaptive-NSGA-II reaches approximately 93.2% (S1), 85.0% (S2), and 93.0% (S3) of that of the standard NSGA-II, while its solution time is only 29.6% (S1), 37.3% (S2), and 39.0% (S3) of the latter, indicating high overall efficiency.

Figure 3 shows the HV convergence curves of Adaptive-NSGA-II under scenarios S1, S2, and S3. It can be observed that the HV under all three scenarios follows a typical evolution pattern: rapid improvement in the early stage, slow improvement in the middle stage, and stable convergence in the later stage.

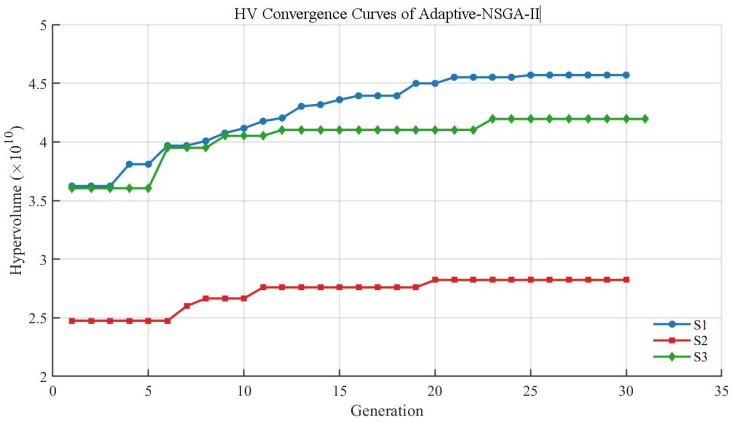

Figure 3. HV convergence curves of Adaptive-NSGA-II in scenarios S1, S2, and S3.

In scenario S1, the HV continuously increases from an initial value of 3.6234 to 4.5692, with a large overall improvement, indicating that the algorithm has strong search capability in small-scale scenarios. In scenario S2, the HV increases from 2.4745 to 2.8224, showing a relatively small but steady rise. In scenario S3, the HV increases from 3.6042 to 4.1948 and stabilizes after the 20th generation, suggesting that the algorithm can complete its main search process within a small number of generations in high-density scenarios.

These results indicate that the proposed Adaptive-NSGA-II exhibits stable convergence behavior across scenarios of different scales. Notably, in complex scenarios, it can obtain high-quality non-dominated solution sets in a relatively small number of actual evolution generations, thereby avoiding the extra overhead caused by the many redundant iterations in the later stages of the standard NSGA-II.

Figure 4 illustrates the dynamic population size adjustment process of Adaptive-NSGA-II in the three scenarios. It can be observed that the algorithm does not maintain a fixed population size; instead, it dynamically adjusts the population based on the HV improvement trend.

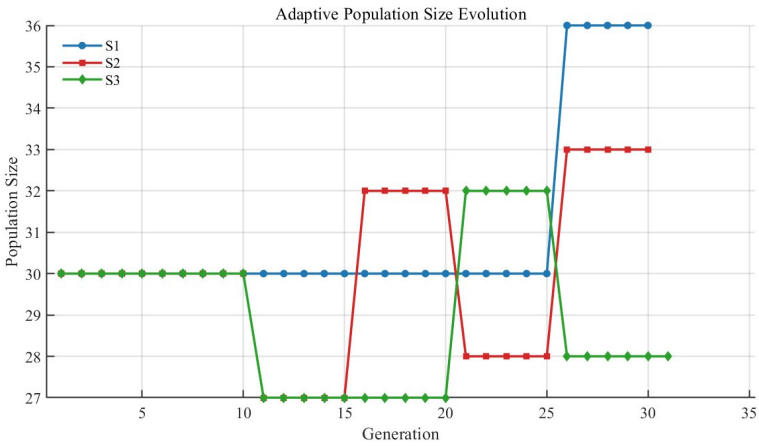

Figure 4. Dynamic population size adjustment process of Adaptive-NSGA-II in three types of scenarios.

In scenario S1, the population size remains at 30 for most iterations and expands to 36 in the later stage. In scenario S2, the population size undergoes a dynamic change: 30 → 27 → 32 → 28 → 33. In scenario S3, the population size evolves as 30 → 27 → 32 → 28. These results indicate that the algorithm can keep a small population size in the early stage to reduce computational cost, expand the population to enhance diversity when HV im-

provement slows down, then moderately compress the population after convergence stabilizes, and finally cooperate with the early stopping mechanism to terminate early.

This adaptive population adjustment improves the utilization efficiency of computational resources, enabling the algorithm to better adapt to task scenarios of different scales and complexities. Notably, in S2 and S3, the algorithm does not run to the preset 100 generations but completes convergence around the 30th to 31st generation, demonstrating good potential for online applications.

Figure 5 provides a comprehensive comparison of mobile equipment activation and mobility distance under different strategies.

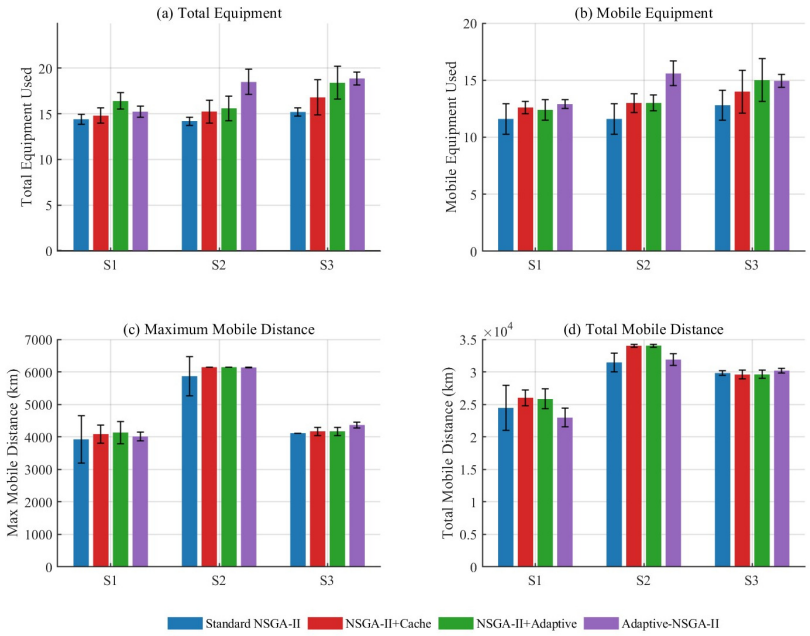

Figure 5. Comprehensive comparison of mobile equipment activation and travel distance under different strategies. Error bars represent the standard deviation over 30 independent runs.

Figure 5 provides a comprehensive comparison of equipment usage and mobility distance among the four multi-objective algorithms that achieve 100% task completion rates. GA-PE, the single-objective comparator, is excluded from this figure because its task completion rates fall well below 100% (87.0% in S1, 56.5% in S2, and 64.34% in S3). While GA-PE uses fewer equipment units (as shown in the supplementary data available upon request), this apparent resource saving is achieved at the cost of failing to schedule a significant portion of tasks—a condition that is unacceptable in practical launch scenarios where mission completeness is mandatory. Including GA-PE in this equipment comparison would be misleading, as its resource-saving behavior does not reflect a valid trade-off under the prerequisite of 100% task completion.

Among the four multi-objective algorithms, Adaptive-NSGA-II uses 15.23–18.87 total equipment units and 12.90–15.59 mobile units across the three scenarios, which are slightly higher than the standard NSGA-II in larger scenarios—a trade-off for maintaining 100% task completion while achieving faster computation. For mobility distance, Adaptive-NSGA-II achieves 4011.84–6136.07 km in maximum distance and 22,964.05–31,891.43 km in total distance, which are comparable to or lower than the standard NSGA-II across all scenarios. The small standard deviations indicate consistent performance across different random seeds. This demonstrates that the proposed method not only reduces computation time but also maintains competitive mobility costs.

Based on the experimental results of the three static scenarios for the multi-objective algorithms, the following conclusions can be drawn.

1. The proposed Adaptive-NSGA-II can stably generate feasible solutions with 100% task completion rate in all three scenarios, indicating that the established candidate combination screening and rule-driven decoding framework has good feasibility guarantee capability.
2. In terms of solution efficiency, Adaptive-NSGA-II shows a significant advantage over the standard NSGA-II: solution time reductions of 70.4%, 62.7%, and 61.0% in S1, S2, and S3, respectively (based on mean values). In particular, in the high-density scenario S3, the algorithm can terminate the search early within only 30.87 ± 0.35 generations, showing strong engineering application potential.
3. The caching mechanism has a clear acceleration effect, and the absolute time benefit becomes more significant as the scenario scale increases; the adaptive population and early stopping strategies mainly improve overall search efficiency by reducing ineffective evolution generations. Their combination yields the best overall time performance.

Therefore, it can be concluded that the proposed method does not simply pursue static global optimality but, while ensuring scheduling feasibility and relatively high solution quality, places more emphasis on fast solution generation capability and engineering real-time performance under complex constraints. This characteristic also provides a usable high-quality baseline schedule for subsequent emergency task insertion and dynamic rescheduling.

7.3.2. Experimental Results of Dynamic Tasks

To validate the proposed priority-rule-based dynamic rescheduling method, emergency tasks are inserted into the three static baseline scenarios S1, S2, and S3. The proposed method is compared against three baseline strategies: (i) fixed-time direct insertion, (ii) non-preemptive rolling search, and (iii) greedy repair heuristic. The results are shown in Table 6.

Table 6. Experimental results of emergency task insertion.

Scenario	Method	Success	Newly Added Fixed/ Mobile Equipment	Newly Added Total Mobility Distance	Rescheduling Runtime
S1	Direct	Yes	2/6	3407.34 km	2 s
S1	Non-Preemptive Rolling	Yes	7/1	0 km	194 s
S1	Greedy Repair Heuristic	Yes	7/1	0 km	235 s
S1	Proposed	Yes	6/1	0 km	223 s
S2	Direct	Yes	2/8	3958.32 km	2 s
S2	Non-Preemptive Rolling	No	-	-	-
S2	Greedy Repair Heuristic	No	-	-	-
S2	Proposed	Yes	5/4	647.53 km	288 s
S3	Direct	No	-	-	-
S3	Non-Preemptive Rolling	No	-	-	-
S3	Greedy Repair Heuristic	No	-	-	-
S3	Proposed	Yes	4/5	1530.71 km	362 s

As shown in Table 6, in the low-conflict scenario S1, all four methods succeed, but with markedly different resource efficiency. Direct insertion activates six mobile units and incurs 3407 km of mobility distance, whereas the proposed method uses six fixed and one mobile unit with zero additional distance—comparable to the non-preemptive rolling and greedy repair strategies, though with one fewer fixed unit. In S2, direct insertion still suc-

ceeds but at an unsustainable cost: eight mobile units and 3958 km. Both the non-preemptive rolling and greedy repair heuristics fail, demonstrating that neither rolling search alone nor indiscriminate release with greedy reassignment can handle cascading conflicts when the baseline schedule is saturated. The proposed method, by contrast, succeeds with only four mobile units and 647.53 km—just 16.4% of the direct insertion cost. In the high-density S3, direct insertion, non-preemptive rolling, and greedy repair all fail; only the proposed method succeeds, using four fixed and five mobile units with 1530.71 km of additional mobility.

Based on the above dynamic experimental results, the following conclusions can be drawn.

1. The proposed priority-rule-based rolling rescheduling strategy is the only method that successfully inserts emergency tasks in all test scenarios, whereas the direct insertion strategy fails in S3, and both the non-preemptive rolling and greedy repair heuristic fail in S2 and S3. This clearly demonstrates that the combination of rolling horizon search with priority-driven selective release and backfilling is essential for handling cascading conflicts in dense launch scenarios.
2. While successfully inserting emergency tasks, the proposed method greatly reduces the number of newly activated mobile equipment units and the total mobility distance compared with direct insertion. In S1, it achieves zero mobility cost with fewer fixed equipment units than the other rolling methods; in S2, it reduces the mobility distance by 83.6% compared with direct insertion (647.53 km vs. 3958.32 km), reflecting the design goal of "low disturbance, high agility".
3. From the equipment activation structure, the proposed method tends to prioritize fixed equipment and only uses mobile equipment when necessary. This characteristic is consistent with the optimization objective to "minimize the number of activated mobile equipment" in static scheduling, indicating that the dynamic rescheduling mechanism can inherit the economic preference of static scheduling.
4. In scenario S3, although more mobile equipment is required and the runtime increases to 362 s, the algorithm still finds a feasible insertion scheme while direct insertion and both baseline rolling strategies completely fail. This demonstrates that the proposed method possesses the robustness needed to cope with sudden high-density task demands in flight-like launches, at the cost of increased computational effort—a trade-off that is justified when feasibility is the primary concern.

In summary, the experimental results in Section 7.3 validate the proposed method in both static multi-task optimization and dynamic rescheduling. For static scheduling, Adaptive-NSGA-II achieves a 100% task completion rate, reduces solution time by 19.3–44.4% compared to standard NSGA-II, and maintains an HV above 93% of the latter, striking a favorable balance between solution quality and computational efficiency. For dynamic scheduling, the priority-based rolling rescheduling strategy successfully inserts emergency tasks in all scenarios, incurs zero mobility cost in low-conflict scenarios, and significantly outperforms the fixed-time direct insertion strategy as well as the non-preemptive rolling and greedy repair heuristics under high conflict. These results demonstrate the engineering applicability and performance advantages of the proposed method in flight-like launch TT&C resource scheduling scenarios.

8. Conclusions and Future Work

To address the strong constraints, multiple objectives, and high dynamism of space TT&C resource scheduling in flight-like launch scenarios, this paper proposes a multi-task dynamic scheduling method that integrates priority rules with an adaptive NSGA-II. Focusing on the static optimization of TT&C resources and the dynamic insertion of emer-

gency tasks, the study completes modeling, solution, and simulation verification. The main conclusions are as follows.

1. In dynamic scenarios, the proposed priority-based rolling rescheduling achieves 100% insertion success across all tests, whereas the fixed-time strategy fails under high conflict (S3), demonstrating superior adaptability.
2. A static multi-task optimization method based on Adaptive-NSGA-II is proposed. By introducing a decoding cache, adaptive population adjustment, and an early stopping mechanism, combined with candidate combination pre-screening and rule-driven decoding, the solution efficiency under complex constraints is improved. Experimental results show that in the three static scenarios S1, S2, and S3, the proposed method consistently yields feasible solutions with a 100% task completion rate. Compared with the standard NSGA-II, the solution time is reduced by 70.4%, 62.7%, and 61.0%, respectively, while maintaining a high hypervolume (HV) level, demonstrating a favorable balance between solution quality and computational efficiency.
3. A priority-rule-based rolling rescheduling mechanism is proposed. After an emergency task arrives, the method first attempts direct insertion. When resource conflicts occur, it locally releases and backfills lower-priority tasks according to task priorities, and completes the updated schedule through rolling horizon search. Experimental results show that in the three dynamic scenarios D1, D2, and D3, the proposed method successfully inserts emergency tasks in all cases, whereas the fixed-time direct insertion strategy fails in the high-density scenario S3, confirming the stronger dynamic adaptability of the proposed method.
4. The proposed method exhibits superior resource organization characteristics in dynamic scenarios. Compared with the direct insertion strategy, it makes fuller use of fixed equipment and effectively suppresses mobility costs: it achieves zero mobility distance insertion in scenario S1, significantly reduces the total mobility distance in scenario S2, and still accomplishes emergency task support under high-conflict conditions with controllable mobility cost in scenario S3, validating its low disturbance and high robustness.

In summary, the proposed method achieves efficient static optimization of space TT&C resources and dynamic rescheduling of emergency tasks while satisfying coverage and accuracy requirements. It is suitable for TT&C support scenarios in flight-like launches that demand fast response and engineering executability.

Although this study has obtained certain results, there remains room for further expansion. Future research can focus on the following aspects: (1) extending the unified rescheduling model to cover multiple types of dynamic events such as equipment failures, point failures, and communication interruptions; (2) introducing factors such as task value, disturbance propagation depth, and recovery cost to further improve the dynamic conflict resolution mechanism; and (3) exploring parallel computing and reinforcement learning methods to further enhance real-time solution capability in large-scale online dynamic scheduling scenarios. Specifically, we plan to formulate the dynamic rescheduling problem as a Markov Decision Process (MDP), where the state encodes the current equipment occupancy and task queue, the action corresponds to selecting which task to insert or release, and the reward balances insertion success and schedule disturbance. Deep Q-Networks (DQN) or Proximal Policy Optimization (PPO) could be trained offline using historical task scenarios to generate near-optimal insertion decisions in milliseconds, thereby substantially improving real-time responsiveness in large-scale online scheduling scenarios.

Author Contributions: Conceptualization, L.H. and Y.L.; methodology, H.L.; validation, L.H., Y.L. and H.L.; formal analysis, T.L.; investigation, Y.L.; resources, H.L.; data curation, L.H.; writing—

original draft preparation, L.H.; writing—review and editing, L.H.; visualization, T.L.; supervision, Y.L.; project administration, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 12402422.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gooley, T.D. The Satellite Range Scheduling Process. Master's Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Dayton, OH, USA, 1993.
2. Tang, J.; Hou, Y.; Wu, S.; Si, Z.; Xu, J.; Qi, C. An Approach for Spacecraft Operational Task Scheduling Considering Constrained Space–Ground TT&C Resources and Task Splitting. *Aerospace* **2025**, *12*, 1077.
3. Marinelli, F.; Nocella, S.; Rossi, F. A Lagrangian heuristic for satellite range scheduling with resource constraints. *Comput. Oper. Res.* **2011**, *38*, 1572-1583.
4. Park, J.; Noh, G.; Park, C.; Kim, J.; Kim, J.; Lee, D.; Cho, D. Development of mission allocation based on MILP for multi-UAVs with limited resources. *Aerosp. Sci. Technol.* **2025**, *166*, 110598.
5. Rigo, C.A.; Seman, L.O.; Camponogara, E.; Filho, E.M.; Bezerra, E.A.; Munari, P. A branch-and-price algorithm for nanosatellite task scheduling to improve mission quality-of-service. *Eur. J. Oper. Res.* **2022**, *303*, 168–183.
6. Xu, D. Research on Resource Scheduling Algorithm for Space-Ground Station Network Based on Mathematical Programming. Master's degree thesis, University of Electronic Science and Technology of China, Chengdu, China, 2024.
7. Zhao, G.P.; Cao, Y.F.; Ni, K.W.; Wang, Y.Z.; Zhang, X.T.; Zheng, Y.C. A TT&C resource scheduling method for large-scale constellations. Chinese Patent Application No. 202410653499[P], 13 December 2024.
8. Li, C.; Liu, P.; Liu, C.; Zhang, H.P.; Chen, M.; Wu, L. A priority-assisted divide-and-conquer strategy for multi-satellite multi-station TT&C resource allocation. *Spacecr. Eng.* **2024**, *33*, 29–36.
9. Xin, L.Q.; Zhang, C.; Zhao, L.Z.; Liu, J.P. Conflict avoidance scheduling algorithm for complex associated TT&C requirements. *Syst. Eng. Electron.* **2022**, *44*, 1581–1588.
10. Lei, H.; Zhang, X.; Liao, W.; Wei, G.; Fan, S. Task Allocation Method for Emergency Active Debris Removal Based on the Fast Elitist Non-Dominated Sorting Genetic Algorithm. *Aerospace* **2025**, *12*, 405.
11. Chen, M.; Wen, J.; Song, Y.J.; Xing, L.N.; Chen, Y.W. A population perturbation and elimination strategy based genetic algorithm for multi-satellite TT&C scheduling problem. *Swarm Evol. Comput.* **2021**, *65*, 100912.
12. Zhang, J.X.; Hu, M.; Zhong, W.A.; Sun, L.Y.; Hu, P.; Ye, X.; Yan, Z. Multi-mission parallel planning method for space launch site based on genetic algorithm. *J. Electron. Meas. Instrum.* **2025**, *39*, 117–124.
13. Ma, L.; Qin, Y.; Qin, J.H.; Xu, M. TT&C mission planning for large constellations using hybrid simulated annealing genetic algorithm. *J. Astronaut.* **2023**, *44*, 1757–1766.
14. Yu, T.; Chen, K.; Li, S.; Yu, Q.; Jia, P.; Li, L. An improved method for complex constraints on TT&C scheduling problem. In Proceedings of the 42nd Chinese Control Conference, Tianjin, China, 24–26 July 2023.
15. Huang, F.; Xiao, B.; Jin, Y. Multi-task Vehicle Routing Problem with Unpaired Pickup and Delivery Demands using NSGA-II Algorithm. In Proceedings of the 2024 15th International Conference on Reliability, Maintenance and Safety (ICRMS), Guilin, China, 31 July 2024–3 August 2024.
16. Xu, Z.; Elomri, A.; Pokharel, S.; Mutlu, F. A model for capacitated green vehicle routing problem with the time-varying vehicle speed and soft time windows. *Comput. Ind. Eng.* **2019**, *137*, 106011.
17. Ye, B.; Qin, F.; Zhang, J. Research on Airline Crew Scheduling Model for Fatigue Management. *Aerospace* **2025**, *12*, 1116.
18. Sun, P.; Chen, G.Y.; Zhang, J.Y.; Wu, J.S. Dynamic adjustment model and algorithm for mission planning based on emergencies. *Control Decis.* **2020**, *35*, 1052–1062.

19. Tian, C.S.; Zhang, C.Y.; Wang, W.X.; Tian, Q.C. Dynamic optimization algorithm for airport flight scheduling. *Mod. Electron. Tech.* **2019**, *42*, 33–40.
20. Fang, J. Research on Theory and Methods of Emergency Transportation Resource Scheduling Under Unexpected Incidents. Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2015.