aerospace

MDPI

Article

Threshold-Gated Conflict-Aware Arc Selection for Satellite Task Scheduling: Evidence from Starlink Mega-Constellation Simulations

Xiaoxuan Li, Yuanyuan Jiao and Xiaogang Pan *

National Key Laboratory of Information Systems Engineering, National University of Defense Technology, Changsha 410003, China; lixiaoxuan2024@nudt.edu.cn (X.L.); jyjy0103@163.com (Y.J.)
* Correspondence: panxiaogang_nudt@163.com; Tel.: +86-137-8711-5398

Abstract

Greedy schedulers are widely used for low-Earth-orbit satellite task scheduling for their computational efficiency and determinism. However, they select candidate arcs by primary criteria alone, so near-equivalent alternatives are separated arbitrarily, potentially reducing scheduling flexibility for later tasks under contention. We present FAMAS-G, a lightweight conflict-aware extension of greedy scheduling rather than a new optimization framework. Its threshold-gated conflict-aware arc selection activates only when primary scores cannot clearly distinguish among candidates, applying a bounded conflict adjustment that favors lower-conflict arcs without displacing clearly superior ones. It relies only on pre-computed arc-level conflict degrees, adding no message passing, backtracking, or iterative search. We evaluate FAMAS-G against Greedy-Central, Base-CNP, and the original FAMAS in Starlink mega-constellation simulations at 100, 300, and 500 tasks per 6 h period across four TLE epochs. Hierarchical bootstrap analysis shows statistically supported improvements over Greedy-Central at all scales (ΔWFR $= +0.0111$, $+0.0051$, and $+0.0024$), with the advantage attenuating as contention rises. Ablation isolates the conflict-aware tiebreak as the primary positive contributor; urgent priority shows no measurable effect, whereas the risk tiebreak shows a small but significant negative effect. Lightweight conflict awareness thus improves greedy scheduling under contention while retaining single-pass, sub-second execution. Significance at the largest scale is sensitive to epoch inclusion; results are specific to the evaluated Starlink scenarios and require further validation.

Keywords: satellite task scheduling; conflict-aware heuristic; greedy algorithm; Starlink constellation; ground station network; mega-constellation

Academic Editors: Andrew W. H. Ip, Zhuming Bi and Kai Leung Yung

Received: 24 August 2026
Revised: 7 September 2026
Accepted: 8 September 2026
Published: 11 September 2026

1. Introduction

The deployment of low-Earth-orbit (LEO) mega-constellations—led by SpaceX's Starlink, whose mid-2026 TLE snapshots list more than 10,000 satellites—has transformed satellite task scheduling from a manageable resource allocation problem into a large-scale combinatorial optimization challenge. Each satellite requires periodic communication windows with ground stations for telemetry, tracking, and command (TT&C) operations. With thousands of satellites competing for access to a limited number of ground stations, each offering narrow visibility windows of a few minutes per pass, the scheduling problem must reconcile competing demands under tight temporal constraints [1,2].

The satellite range scheduling problem (SRSP) has been studied extensively since the 1990s, primarily in the context of the U.S. Air Force Satellite Control Network (AFSCN).

We organize the literature along a methodological spectrum from centralized optimization through heuristics to distributed approaches, identifying a specific gap at the intersection of greedy efficiency and conflict awareness.

Centralized optimization formulates scheduling as a mixed-integer or constraint program. Foundational AFSCN work combined MIP with heuristic insertion, achieving approximately 91% scheduling rates [3]; Lagrangian fix-and-relax solved GALILEO instances via maximum-weight independent set on interval graphs [4]; and computational tractability was characterized for single-satellite, multi-station formulations [5,6]. Although recent branch-and-cut-and-price methods solve related agile Earth observation instances at moderate scale [7], exact optimization of large TT&C instances remains computationally demanding, motivating efficient heuristic and distributed alternatives.

Heuristic and metaheuristic approaches trade optimality guarantees for computational speed. Greedy construction has long served as a practical AFSCN baseline [1,8] and remains competitive in recent practice [9]. Genetic algorithms have been widely applied to single-station and massive-constellation instances [10–12]; HSAGA reports >99% completion on a substantially different benchmark (180 satellites, 32 antennas), a completion rate that is not directly comparable with the WFR values reported here. Conflict information has been exploited mainly at the window level within iterative or multi-stage frameworks: priority-based conflict-avoidance heuristics [13], post hoc arc replacement [14], variable neighborhood search with Metropolis acceptance [15], and conflict-priority tabu search whose indicators guide neighborhood moves [16]. These methods achieve strong performance but introduce computational overhead from iterative, population-based, or multi-stage search. Knowledge-assisted adaptive large neighborhood search [17] and recent hybrids of metaheuristics with reinforcement-learning controllers [18–20] extend the same trade-off: better solutions at higher computational cost.

Distributed and market-based approaches address scheduling through decentralized negotiation based on the Contract Net Protocol (CNP) [21], with variants for task allocation in satellite swarms [22], circular contracts [23], onboard coordination under uncertainty [24], and disturbance propagation in dynamic TT&C scheduling [25]. These methods excel at handling dynamic disturbances, but under high contention communication overhead increases and bid informativeness decreases [26].

Remaining gap. Despite this extensive body of work, a specific gap exists at the intersection of greedy efficiency and arc-level conflict awareness. Greedy schedulers offer millisecond-scale execution and deterministic reproducibility but select arcs using only primary criteria (typically earliest start time); when two arcs have nearly identical start times but different conflict implications for subsequent tasks, the choice is arbitrary and may inadvertently consume an arc critical for other tasks. Metaheuristic and multi-stage methods [13,14] incorporate conflict information but abandon the single-pass greedy structure, introducing substantially higher computational overhead associated with iterative, population-based, or multi-stage search. Distributed CNP-based methods address resource contention through negotiation but require message-passing infrastructure and face scalability challenges under high load. Among the representative methods reviewed here, we did not identify an approach that combines arc-level conflict scoring, near-tie threshold activation, bounded adjustment, and single-pass greedy selection with mechanism-level ablation within the same framework. Our novelty claim concerns the integration of these design elements within a lightweight single-pass greedy scheduler and their mechanism-level empirical isolation; we do not claim that conflict scoring, thresholding, bounded adjustment, or greedy scheduling is individually novel in isolation. This gap is practically important when schedules must be re-computed frequently, deterministic behavior is desirable for operational verification, and the scheduling node has limited computational budget. The

present evidence is limited to static Starlink-derived scenarios with five ground stations and fixed task-priority distributions; dynamic arrivals, stochastic disruptions, heterogeneous constellations, and onboard implementation are not evaluated here.

We introduce FAMAS-G (FAMAS–Greedy), a greedy-backbone scheduler augmented with threshold-gated conflict-aware arc selection. The mechanism is simple: when a task has multiple feasible arcs with near-equivalent primary scores (determined by start time), the algorithm prefers the arc with the lowest pre-computed conflict degree—the number of other arcs with which it temporally overlaps at the same station. This bias is threshold-gated—it activates only when primary scores cannot clearly discriminate among candidates, and its magnitude is bounded to prevent pathological deviations from the primary objective. It is lightweight—it requires only pre-computed arc-level conflict degrees and adds no message-passing, backtracking, or iterative optimization to the greedy framework. Unlike the original decentralized FAMAS bidding architecture, which used an unbounded linear conflict penalty that distorted the primary scheduling objective, FAMAS-G employs soft, bounded, threshold-gated adjustments that respect the earliest-arc heuristic when primary score differences are clear.

Accordingly, this study makes three linked contributions. First, it isolates the effect of conflict-aware near-tie selection through mechanism-level ablation: the conflict tiebreak is the dominant positive contributor to the performance advantage, whereas the urgent-priority bonus shows no measurable effect and the risk tiebreak exerts a small but statistically significant negative effect at the tested scales. Second, it characterizes how the marginal benefit varies with task density, documenting contention-induced convergence: the WFR advantage over Greedy-Central narrows from $+0.0111$ to $+0.0024$ as tasks per 6 h period grow from 100 to 500. Third, it evaluates the mechanism across four independent Starlink TLE epochs spanning 16 days and two distinct orbital shells, using a pre-specified hierarchical bootstrap framework ($B = 10{,}000$, 3-level cluster-robust) with Holm–Bonferroni correction and leave-one-epoch-out sensitivity analysis.

The remainder of this paper is organized as follows. Section 2 formalizes the scheduling problem. Section 3 describes the baseline schedulers and FAMAS-G. Section 4 presents the experimental setup and statistical framework, including the small-scale exact benchmark and the parameter-sensitivity analyses. Section 5 reports the results, Section 6 discusses the findings and limitations, and Section 7 concludes the paper.

2. Problem Formulation

2.1. System Entities and Candidate Arcs

We consider a satellite task scheduling problem with the following entities.

Ground Stations. A set $\mathcal{S} = \{S_1, \ldots, S_M\}$ of ground stations, where $M = 5$ in our experiments. Each station S_j has a device transition time τ_j (fixed at 30 s for all stations), representing the minimum reconfiguration interval between consecutive tasks.

Satellites. A constellation of N_{sat} satellites in LEO, with orbits specified by two-line element (TLE) sets at specific reference epochs. Satellite positions are propagated using the SGP4 model.

Tasks. A set of task requests $\mathcal{T} = \{T_1, \ldots, T_N\}$. Each task T_i is characterized by the following:

- $w_i \in \{1, 3\}$: Priority weight (80% weight-1, 20% weight-3 in our generated instances).
- $d_i \in \{120, 180, 240, 300\}$ s: Required tracking duration, drawn uniformly.
- $\mathcal{S}_i \subseteq \mathcal{S}$: Set of compatible ground stations.
- $[r_i, \delta_i]$: Release time and deadline within a 6 h (21,600 s) planning period.

The weight used by the scheduler can be interpreted more generally as a cycle-specific effective priority rather than as a permanently fixed mission attribute. For a dynamic mission, one may define an effective weight

$$w_i^{\text{eff}}(t) = w_i^0 \, f_i^{\text{completion}}(t) \, f_i^{\text{urgency}}(t) \, f_i^{\text{revisit}}(t), \tag{1}$$

where the factors respectively encode the value remaining after previous observations, deadline-related urgency, and the residual demand for repeated observations. FAMAS-G is agnostic to the mechanism that generates w_i^{eff} (Equation (1)); updated priorities can be supplied before each replanning cycle without changing the conflict-aware arc-selection rule. The present experiments use a fixed two-level weight distribution and therefore do not validate performance under dynamically evolving priorities.

Candidate Arcs. For each task T_i, a set of candidate arcs $\mathcal{P}_i = \{p_{i,1}, \ldots, p_{i,k_i}\}$ is generated by discretizing satellite-to-station visibility windows at $\Delta = 30$ s resolution. A visibility window exists when the satellite is above a 5° minimum elevation angle relative to the station, as determined by SGP4 propagation. Each arc $p = (i, j, t_{\text{start}}, t_{\text{end}})$ encodes the task i, station j, start time $t_{\text{start}}(p)$, end time $t_{\text{end}}(p) = t_{\text{start}}(p) + d_i$, and a conflict degree $c(p) \in \mathbb{N}_0$ (defined below).

2.2. Conflict Graph and Optimization Model

Conflict Graph. Two arcs p and q are in conflict, denoted by $(p, q) \in \mathcal{E}$, if and only if they are assigned to the same station and their temporal intervals overlap, including the transition time:

$$\begin{aligned}(p,q) \in \mathcal{E} \iff \text{station}(p) = \text{station}(q) \wedge [t_{\text{start}}(p),\ t_{\text{end}}(p) + \tau) \cap \\ [t_{\text{start}}(q),\ t_{\text{end}}(q) + \tau) \neq \emptyset.\end{aligned} \tag{2}$$

The conflict degree $c(p) = |\{q : (p, q) \in \mathcal{E}\}|$ counts the number of other arcs that conflict with p (Equation (2)). Conflict degrees are pre-computed once per problem instance via a sweep-line algorithm over time-sorted arcs per station, requiring $O(|\mathcal{P}| \log |\mathcal{P}| + |\mathcal{E}|)$ time, where the $|\mathcal{E}|$ term reflects explicit enumeration of conflict edges (worst-case quadratic in $|\mathcal{P}|$ per station, though physical constraints limit the realized density).

Decision Variables and Formal Model. We formalize the scheduling problem as a combinatorial optimization model. For each candidate arc $p \in \mathcal{P} = \bigcup_{T_i \in \mathcal{T}} \mathcal{P}_i$, define a binary decision variable:

$$x_p = \begin{cases} 1, & \text{if arc } p \text{ is selected for scheduling,} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The objective is to maximize the total weight of scheduled tasks:

$$\max_x \sum_{T_i \in \mathcal{T}} w_i \sum_{p \in \mathcal{P}_i} x_p. \tag{4}$$

Since the total task weight $\sum_{T_i \in \mathcal{T}} w_i$ is constant for a given instance, maximizing (4) is equivalent to maximizing the Weighted Fulfillment Rate (WFR), defined as (5):

$$\text{WFR}(x) = \frac{\sum_{T_i \in \mathcal{T}} w_i \sum_{p \in \mathcal{P}_i} x_p}{\sum_{T_i \in \mathcal{T}} w_i}, \tag{5}$$

which ranges from 0 (no tasks scheduled) to 1 (all tasks scheduled).

The assignment is subject to two core constraints. First, each task may be assigned at most one arc (one-arc-per-task):

$$\sum_{p \in \mathcal{P}_i} x_p \leq 1, \qquad \forall T_i \in \mathcal{T}. \tag{6}$$

Second, conflicting arcs cannot be simultaneously selected:

$$x_p + x_q \leq 1, \qquad \forall (p, q) \in \mathcal{E}. \tag{7}$$

The complete problem is a Maximum-Weight Independent Set on the conflict graph $(\mathcal{P}, \mathcal{E})$ with partition constraints (6), which is NP-hard in general.

Remark on constraint encoding. Station compatibility ($\mathcal{S}_i$), visibility windows (5° minimum elevation, SGP4 propagation), tracking duration (d_i), release time (r_i), and deadline (δ_i) are embedded in the construction of candidate arcs $\mathcal{P}_i$ from satellite-to-station visibility windows at $\Delta = 30$ s discretization, rather than introduced as additional decision constraints in the model. An arc $p \in \mathcal{P}_i$ exists only if $\text{station}(p) \in \mathcal{S}_i$, $t_{\text{end}}(p) - t_{\text{start}}(p) = d_i$, $r_i \leq t_{\text{start}}(p)$, and $t_{\text{end}}(p) \leq \delta_i$. This encoding reduces the size of the constraint system and concentrates the combinatorial difficulty in the conflict relation $\mathcal{E}$.

Optimality Boundary. Equations (4)–(7) constitute a formal combinatorial optimization representation of the scheduling problem. FAMAS-G does not remove the absence of a global optimality guarantee inherent to single-pass greedy scheduling. Its purpose is narrower: it mitigates one specific form of greedy myopia that arises when multiple feasible arcs have nearly equivalent primary scores but substantially different downstream conflict implications. Because task order remains fixed and no backtracking or global search is performed, earlier commitments can still preclude a globally superior schedule. We therefore interpret FAMAS-G as a lightweight heuristic that improves a particular local decision mechanism rather than as a method that solves the global optimality problem. In the primary experiment (Sections 3 and 4), this problem is solved approximately by the greedy heuristics; no mixed-integer programming (MIP) solver is invoked in the primary pipeline. To quantify the residual suboptimality empirically, a supplementary small-scale exact benchmark additionally solves instances with $n \in \{20, 30, 50\}$ to certified optimality using a MIP solver (Section 4.5); the resulting empirical optimality gaps are reported in Section 5.

Implementation Architecture Assumption. In this study, the term "centralized" describes the information scope of the scheduler: the scheduling process has instance-level access to the task set and pre-computed candidate arcs. It does not imply that a designated master satellite propagates the orbits of the entire constellation onboard. TLE propagation with SGP4 and visibility-window generation are performed during instance construction, after which FAMAS-G operates on the pre-computed arc representation. The reported scheduling runtime therefore excludes real-time orbit determination and orbit-update communication. Operational onboard/distributed communication requirements are outside the present scope and should be evaluated separately.

3. Scheduling Methods

3.1. Baseline Methods

We compare FAMAS-G against three baselines spanning the design space from centralized greedy to decentralized coordination.

Greedy-Central (GC). A deterministic centralized greedy scheduler with access to the global set of tasks and candidate arcs for the evaluated planning instance, serving as the primary baseline. Tasks are sorted by a scarcity-adjusted priority measure:

$$\pi(T_i) = \frac{w_i}{1 + |\mathcal{S}_i|}, \tag{8}$$

with the sort key $(-\pi(T_i),\ \delta_i,\ \text{task\_id}_i)$, assigning higher priority to high-weight tasks with few compatible stations. This lexicographic ordering—scarcity-adjusted weight, then deadline, then task ID—provides a deterministic total order that both GC and FAMAS-G share. For each task in order, GC selects the earliest feasible arc (by t_{start}, then station ID, then arc ID). GC has no inter-agent communication, no conflict awareness, and no backtracking. It is a deterministic centralized single-pass greedy baseline without explicit conflict awareness. GC makes decisions using only the candidate arcs and their start times; it does not have access to information about future tasks beyond the fixed processing order.

Base-CNP. A message-driven Contract Net Protocol. A Task Broker Agent (TBA) announces tasks sequentially; each Ground Station Agent (GSA) bids based on task weight, earliest feasible start time, and local station load. Bids use task weight as the utility component without conflict degree, arc scarcity, or risk information. Re-auction on rejection is permitted up to a maximum round limit. Base-CNP represents standard decentralized coordination without conflict awareness.

FAMAS (original). The predecessor to FAMAS-G (internal, unpublished). It uses a linear bid formula:

$$B_{ij} = \eta_1 U_i - \eta_2 C_{ij} - \eta_3 L_j - \eta_4 R_{ij}, \tag{9}$$

where U_i is the task utility (weight), C_{ij} is the conflict degree, L_j is the station load, R_{ij} is the arc-level risk, and the fixed weights are $\boldsymbol{\eta} = (0.4, 0.3, 0.15, 0.15)$. FAMAS (original) also employs hard filtering of non-positive bids, scarcity-first task ordering, and affected-neighborhood repair. FAMAS (original) represents an earlier conflict-aware approach whose design choices FAMAS-G improves upon.

Table 1 summarizes the structural differences among the four methods.

Table 1. Structural comparison of the four scheduling methods.

Property	GC	Base-CNP	FAMAS (Orig.)	FAMAS-G
Architecture	Centralized	Decentralized	Decentralized	Centralized
Coordination	None	Message passing	Message passing	None
Task ordering	Weight/scarcity	Sequential	Scarcity-first	Weight/scarcity
Arc selection	Earliest feasible	Bid-based	Linear bid	5-stage layered
Conflict awareness	None	None	Linear penalty	Threshold-gated
Bid rejection	N/A	No hard filter	Yes	N/A (score-based selection)

Table 1 highlights an operational trade-off rather than a universal ranking. Centralized single-pass methods minimize coordination overhead and provide deterministic low latency but require instance-level information, whereas CNP-based approaches provide decentralized coordination at the cost of message exchange and auction latency. Conflict-aware methods use additional structural information to preserve future flexibility, while hard bid filtering can discard otherwise feasible opportunities. The relative importance of latency, communication, autonomy, and solution quality is mission-dependent; we therefore avoid assigning universal numerical weights to these attributes.

VNS-, ALNS-, and reinforcement-learning-based schedulers were not included as executable baselines because a common implementation and directly compatible benchmark representation were not available for the present task model. We therefore restrict quantitative claims to methods executed on identical instances and do not claim superiority over optimization- or learning-based schedulers on the basis of cross-paper results.

3.2. Proposed Method: FAMAS-G

FAMAS-G is designed as a greedy-backbone scheduler with threshold-gated conflict awareness. It inherits Greedy-Central's task ordering, feasibility filter, and earliest-arc heuristic, then adds three soft decision layers that operate only when candidate arcs are difficult to discriminate by primary score alone. Each layer is bounded in its maximum adjustment to prevent deviations from the greedy baseline that could degrade performance, as occurred with FAMAS (original)'s unbounded linear penalty.

The central insight is that conflict awareness can be injected into a greedy framework without inter-agent message passing or iterative optimization, provided that (a) conflict information is pre-computed at the arc level and (b) the conflict-based adjustment is threshold-gated to activate only when primary scores are ambiguous. FAMAS-G processes tasks sequentially in a fixed deterministic order. For each task, it applies a five-stage arc selection pipeline (Algorithm 1):

1. Feasibility Filter. Remove arcs that are unavailable, incompatible, or conflict with already-scheduled tasks on the same station.
2. Primary Scoring. Score each feasible arc by $s_{\text{prim}}(p) = 1 - t_{\text{start}}(p)/T_{\text{period}}$, preferring earlier start times.
3. Urgent Priority. For tasks with $w_i \geq 3$, add a bounded bonus to the primary score.
4. Conflict Tiebreak. [Core mechanism] Among arcs within a threshold of the best primary score, penalize higher-conflict arcs.
5. Risk Tiebreak. Among arcs within the same threshold, apply a secondary positional/scarcity adjustment.

The selected arc is committed to the station schedule before processing the next task. There is no backtracking, no message passing, and no inter-station coordination. Figure 1 summarizes the pipeline; the complete procedure is given in Algorithm 1 (Section 3.5).

Primary Arc Scoring. The primary score follows the Greedy-Central heuristic: among feasible arcs, earlier start times are preferred.

$$s_{\text{prim}}(p) = 1 - \frac{t_{\text{start}}(p)}{T_{\text{period}}} \in [0, 1], \tag{10}$$

where $T_{\text{period}} = 21{,}600$ s is the planning horizon. This encodes the operational principle that earlier execution preserves more schedule capacity for subsequent tasks. When all awareness mechanisms are disabled ($\beta_{\text{urgent}} = 0$, conflict and risk tiebreaks off), FAMAS-G reduces exactly to Greedy-Central—a design property verified by the built-in self-test in the implementation.

3.3. Threshold-Gated Conflict-Aware Arc Selection

This is the core algorithmic contribution of FAMAS-G.

Motivation. When multiple feasible arcs have nearly identical primary scores (similar start times on different stations), Greedy-Central chooses arbitrarily based on secondary sort keys (station ID, arc ID). Under high contention, this arbitrary choice can inadvertently block subsequent tasks. Conflict-aware selection uses pre-computed arc-level conflict degree to prefer arcs that leave more scheduling flexibility for future tasks.

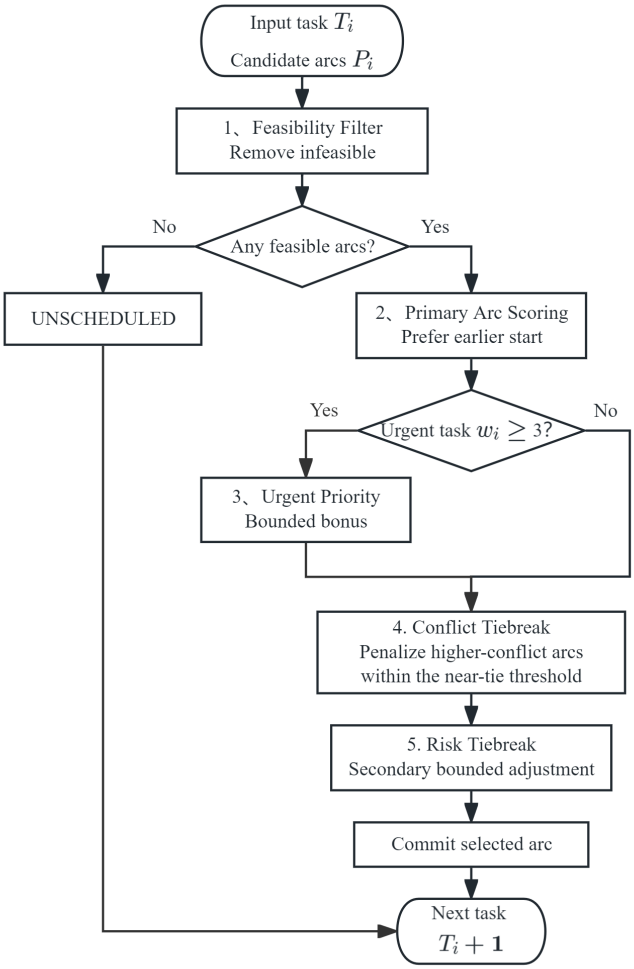

Figure 1. Five-stage FAMAS-G arc-selection pipeline for each task. Candidate arcs are first filtered for feasibility, scored by the primary earliest-start criterion, and then processed by bounded auxiliary adjustments; after these five decision stages, the selected arc is committed to the station schedule. The conflict-aware tiebreak is the core mechanism identified by the ablation analysis.

Threshold Gate. The mechanism activates only when at least two candidate arcs have primary scores within a proximity threshold θ of the best primary score s^*_{prim}:

$$\theta = \tau_{\text{threshold}} \cdot \max(|s^*_{\text{prim}}|,\ \varepsilon), \tag{11}$$

where $\tau_{\text{threshold}} = 0.05$ and $\varepsilon = 0.01$ in the frozen configuration.

Conflict Penalty. For each candidate arc p within the threshold window, a conflict penalty is applied:

$$\delta_{\text{conflict}}(p) = -\frac{c(p)}{\max(1,\ D_{\max}(\text{station}(p)))} \cdot \alpha, \tag{12}$$

where $c(p)$ is the pre-computed conflict degree of arc p, $D_{\max}(j) = \max_{q:\text{station}(q)=j} c(q)$ is the maximum conflict degree on station j, and $\alpha = \tau_{\text{max\_adj}} \cdot \max(|s^*_{\text{prim}}|,\ \varepsilon)$ with $\tau_{\text{max\_adj}} = 0.10$.

The penalty is proportional to the normalized conflict degree: arcs with more temporal overlaps receive larger penalties. For example, when $s^*_{\text{prim}} = 0.8$, the frozen settings $\tau_{\text{threshold}} = 0.05$ and $\tau_{\text{max\_adj}} = 0.10$ give $\theta = 0.04$ and $\alpha = 0.08$. Thus, only arcs within 0.04 of the best primary score are eligible for conflict-based re-ranking, and the maximum

conflict penalty is capped at 0.08—at most 10% of the primary score range, ensuring the mechanism cannot override a substantially better primary arc.

Effect. The mechanism re-ranks arcs within the tiebreak window, promoting lower-conflict arcs over higher-conflict ones while preserving the primary preference for earlier start times. When no arcs fall within the threshold, or when all candidates have identical conflict degrees, the ranking is unchanged.

Relationship to FAMAS (original). In FAMAS (original), conflict degree entered a linear bid formula with a fixed coefficient and hard negative-bid filtering, causing approximately 70% bid rejection and WFR collapse. FAMAS-G instead uses conflict as a soft, bounded, threshold-gated tiebreaker: it never rejects a feasible arc, it only adjusts relative ranking when primary scores cannot clearly discriminate.

3.4. Auxiliary Mechanisms

The conflict-aware tiebreak is complemented by two auxiliary mechanisms, neither of which is a primary contributor at the tested scales.

3.4.1. Urgent Priority

For tasks with weight $w_i \geq 3$, a bounded bonus is added prior to conflict tiebreaking:

$$s_{\text{urgent}}(p) = s_{\text{prim}}(p) + \beta_{\text{urgent}} \cdot \frac{w_i - 1}{4} \cdot \left(1 - \frac{t_{\text{start}}(p)}{T_{\text{period}}}\right), \tag{13}$$

where $\beta_{\text{urgent}} = 0.05$. The weight factor $\frac{w_i - 1}{4}$ equals 0.5 for the modeled urgent weight $w_i = 3$; the normalization would reach 1.0 only at $w_i = 5$, which lies outside the modeled weight set. The time factor ensures the bonus is larger for earlier arcs. With our task distribution (80% $w = 1$, 20% $w = 3$), urgent priority applies to approximately 20% of tasks. The maximum bonus for a weight-3 task at the earliest feasible start time is approximately 0.025, which is small relative to typical primary score gaps between candidate arcs.

Empirical contribution. As reported in Section 5.6, ablation analysis finds that urgent priority contributes negligibly to overall WFR improvement in the current experimental setting (ΔWFR $= -0.0001$ at $n = 500$, $p = 0.39$; similarly null at $n = 100$ and $n = 300$, $p = 0.98$ and $p = 0.76$). It is retained in the algorithm description for completeness but is not a primary contributor.

3.4.2. Risk Tiebreak

A secondary tiebreak mechanism penalizes arcs based on two positional risk factors, each contributing 0–0.25 and summed to a total $R(p)$ capped at 1.0.

Temporal edge risk. For arcs belonging to the same task and station, let $L = |\{q \in \mathcal{P}_i : \text{station}(q) = \text{station}(p)\}|$ be the number of such arcs, and let $\text{idx}(p)$ be the 0-based rank of $t_{\text{start}}(p)$ among their distinct start times. When $L > 1$:

$$R_{\text{edge}}(p) = 0.25\left(1 - \frac{\min(\text{idx}(p),\ L - 1 - \text{idx}(p))}{L - 1}\right), \tag{14}$$

penalizing arcs at the temporal extremes of the same-task same-station arc set.

Scarcity risk. Let $n(p) = |\{q \in \mathcal{P}_i : \text{station}(q) = \text{station}(p)\}|$ be the number of candidate arcs available for task T_i on the same station. When $n(p) \leq 5$,

$$R_{\text{scarcity}}(p) = 0.25\left(1 - \frac{n(p)}{6}\right), \tag{15}$$

penalizing arcs belonging to tasks with few alternatives on a given station; $R_{\text{scarcity}}(p) = 0$ when $n(p) > 5$.

The risk adjustment is $\delta_{\text{risk}}(p) = -R(p) \cdot \beta$, where $R(p) = \min(R_{\text{edge}}(p) + R_{\text{scarcity}}(p),\ 1.0)$ and $\beta = \tau_{\text{risk\_max\_adj}} \cdot \max(|s^*_{\text{prim}}|,\ \varepsilon)$, with $\tau_{\text{risk\_max\_adj}} = 0.10$.

Empirical contribution. Ablation analysis (Section 5.6) finds that the risk tiebreak exerts a small but statistically significant negative effect on WFR under the frozen configuration: $\Delta\text{WFR} = +0.0042$ at $n = 500$ ($p < 0.0001$, $d_z = -0.55$), favoring the no-risk variant, with similar effects at $n = 100$ ($+0.0066$) and $n = 300$ ($+0.0053$). The risk signal is anti-correlated with scheduling quality in the current setting. It is retained for completeness but is not a primary contributor.

3.5. Unified Scoring and Algorithm

The arc selection process can be expressed as a unified additive scoring function. Define the threshold-gating indicator:

$$\mathbb{I}_\theta(p) = \begin{cases} 1, & \text{if } |s_{\text{prim}}(p) - s^\star_{\text{prim}}| < \theta, \\ 0, & \text{otherwise,} \end{cases} \tag{16}$$

where $s^\star_{\text{prim}} = \max_{p \in \mathcal{C}_i} s_{\text{prim}}(p)$ and θ is defined in Equation (11) with $\varepsilon = 0.01$.

The complete FAMAS-G score for arc $p \in \mathcal{C}_i$ is:

$$s_{\text{FG}}(p) = s_{\text{prim}}(p) + \delta_{\text{urgent}}(p) + \delta_{\text{conflict}}(p) + \delta_{\text{risk}}(p), \tag{17}$$

where

$$\delta_{\text{urgent}}(p) = \mathbb{1}[w_i \geq 3] \cdot \beta_{\text{urgent}} \cdot \frac{w_i - 1}{4} \cdot \left(1 - \frac{t_{\text{start}}(p)}{T_{\text{period}}}\right), \tag{18}$$

$$\delta_{\text{conflict}}(p) = -\mathbb{I}_\theta(p) \cdot \frac{c(p)}{\max(1,\ D_{\max}(\text{station}(p)))} \cdot \alpha, \tag{19}$$

$$\delta_{\text{risk}}(p) = -\mathbb{I}_{\theta_r}(p) \cdot R(p) \cdot \beta, \tag{20}$$

with $\alpha = \tau_{\text{max\_adj}} \cdot \max(|s^\star_{\text{prim}}|,\ \varepsilon)$, $\beta = \tau_{\text{risk\_max\_adj}} \cdot \max(|s^\star_{\text{prim}}|,\ \varepsilon)$, and θ_r computed analogously to θ using the same $\tau_{\text{threshold}} = 0.05$ and $\varepsilon = 0.01$.

The selected arc for task T_i is then as follows:

$$p^\star_i = \arg\max_{p \in \mathcal{C}_i} s_{\text{FG}}(p), \tag{21}$$

with ties broken deterministically by $(t_{\text{start}}(p),\ \text{station\_id},\ \text{arc\_id})$.

The indicator $\mathbb{I}_\theta(p)$ in Equation (16) formally captures the threshold-gated property: among candidates whose primary scores are within the proximity threshold θ of the best primary score, the conflict penalty δ_{conflict} may alter the ranking; candidates farther than θ from the best primary score are unaffected by the conflict adjustment. When no alternative candidate lies within the proximity threshold of the best primary score ($\mathbb{I}_\theta(p) = 1$ only for the current-best arc p^*_{prim}), the conflict adjustment applies uniformly to that single arc, leaving the ranking unchanged. When all candidates within the threshold have identical conflict degrees, the penalty is uniform and does not change the relative ordering. Under either condition the conflict tiebreak cannot change the selected arc. When, additionally, the urgent bonus and risk tiebreak are disabled or inactive, FAMAS-G reduces to Greedy-Central—a condition verified by the code-level self-test (FAMAS-G with all awareness

off ≡ GC). Boundedness Property. For any arc $p \in \mathcal{C}_i$, the conflict adjustment satisfies the following:

$$|\delta_{\text{conflict}}(p)| \leq \alpha \equiv \tau_{\text{max\_adj}} \cdot \max(|s^{\star}_{\text{prim}}|,\ \varepsilon) \leq 0.10 \cdot \max(|s^{\star}_{\text{prim}}|,\ 0.01). \tag{22}$$

This follows directly from $c(p) \leq D_{\max}(\text{station}(p))$ (by definition of $D_{\max}$), which bounds the normalized conflict degree $c(p)/\max(1, D_{\max}(\text{station}(p))) \leq 1$. The boundedness property ensures the conflict term cannot override a substantially better primary arc: at $s^{\star}_{\text{prim}} = 0.8$, the maximum penalty is 0.08, which is at most 10% of the primary score range. This contrasts with FAMAS (original)'s unbounded linear penalty, whose magnitude scaled with the raw conflict degree and caused approximately 70% bid rejection.

Algorithm 1 summarizes the complete pipeline.

Algorithm 1: FAMAS-G static scheduling.

Input: Tasks $\mathcal{T}$, Arcs $\mathcal{P}$, Stations $\mathcal{S}$, transition times τ, config C
Output: Assignments σ, task statuses

for each station $j \in \mathcal{S}$: schedule$[j] \leftarrow \emptyset$
$\sigma \leftarrow \emptyset$
Pre-compute $D_{\max}[j] \leftarrow \max_{p:\text{station}(p)=j} c(p)$ for each station j
Sort $\mathcal{T}$ by $(-w_i/(1+|\mathcal{S}_i|),\ \delta_i,\ \text{task\_id}_i)$ ▷ GC ordering
for each task T_i in sorted order:
  $\mathcal{C}_i \leftarrow \{p \in \mathcal{P}_i : \text{available}(p) \wedge \text{feasible}(p, \text{schedule})\}$
  if $\mathcal{C}_i = \emptyset$: mark UNSCHEDULED; **continue**
  for each $p \in \mathcal{C}_i$: $s_{\text{prim}}(p) \leftarrow 1 - t_{\text{start}}(p)/T_{\text{period}}$
  if $w_i \geq 3$: ▷ Urgent Priority
    $s_{\text{total}}(p) \leftarrow s_{\text{prim}}(p) + \beta_{\text{urgent}} \cdot \frac{w_i - 1}{4} \cdot (1 - t_{\text{start}}(p)/T_{\text{period}})$
  else: $s_{\text{total}}(p) \leftarrow s_{\text{prim}}(p)$
  Sort $\mathcal{C}_i$ by s_{total} descending; $p^* \leftarrow \mathcal{C}_i[0]$
  if C.enable_conflict_tiebreak $\wedge\ |\mathcal{C}_i| \geq 2$: ▷ Conflict Tiebreak
    $\theta \leftarrow C.\tau_{\text{threshold}} \cdot \max(|s_{\text{prim}}(p^*)|,\ 0.01)$
    $\alpha \leftarrow C.\tau_{\text{max\_adj}} \cdot \max(|s_{\text{prim}}(p^*)|,\ 0.01)$
    for each $p \in \mathcal{C}_i$ where $|s_{\text{prim}}(p) - s_{\text{prim}}(p^*)| < \theta$:
      $\delta_c(p) \leftarrow -(c(p)/\max(1, D_{\max}[\text{station}(p)])) \cdot \alpha$
      $s_{\text{total}}(p) \leftarrow s_{\text{total}}(p) + \delta_c(p)$
    Re-sort $\mathcal{C}_i$; $p^* \leftarrow \mathcal{C}_i[0]$
  if C.enable_risk_tiebreak $\wedge\ |\mathcal{C}_i| \geq 2$: ▷ Risk Tiebreak
    for each $p \in \mathcal{C}_i$ where $|s_{\text{prim}}(p) - s_{\text{prim}}(p^*)| < \theta_r$:
      $s_{\text{total}}(p) \leftarrow s_{\text{total}}(p) - R(p) \cdot \beta$
    Re-sort $\mathcal{C}_i$; $p^* \leftarrow \mathcal{C}_i[0]$
  $\sigma(T_i) \leftarrow p^*$; schedule[station$(p^*)$].append$(p^*)$
return σ

3.6. Computational Complexity

The complexity of FAMAS-G decomposes into a one-time preprocessing phase and a per-task scheduling phase.

Preprocessing. Conflict graph construction uses a sweep-line algorithm per station, sorting arcs by start time and scanning forward to detect temporal overlaps. With $|\mathcal{P}|$ candidate arcs distributed across M stations, this requires $O(|\mathcal{P}| \log |\mathcal{P}| + |\mathcal{E}|)$ time, where the $|\mathcal{E}|$ term accounts for explicit edge enumeration. Station-level maximum conflict degrees $D_{\max}(j)$ are computed in $O(|\mathcal{P}|)$ via a single pass over all arcs.

Per-task scheduling. For each task T_i, the feasibility filter checks $|\mathcal{P}_i|$ candidate arcs against all committed arcs on the relevant station: $O(|\mathcal{P}_i| \cdot |\text{committed}|)$. Primary scoring, urgent bonus, and tiebreak adjustments each require $O(|\mathcal{P}_i|)$ operations. In the worst case, $|\text{committed}| = O(N)$ per station, and with $K = \max_i |\mathcal{P}_i|$ denoting the maximum number of candidate arcs per task, the per-task feasibility check contributes $O(N \cdot K \cdot |\text{committed}|)$. The total worst-case complexity is $O(|\mathcal{P}| \log |\mathcal{P}| + N \log N + N \cdot K \cdot |\text{committed}|)$. Since $|\text{committed}| \leq N$ in the worst case and K is typically 8–15, the pessimistic bound simplifies to $O(|\mathcal{P}| \log |\mathcal{P}| + N \log N + N^2 K)$. In practice, the number of committed arcs per station is bounded by the station's scheduling capacity (a 6 h period accommodates at most a few hundred arcs per station at 30 s transition time), and the average $|\mathcal{P}_i|$ is approximately 6–7 across task scales, with per-task maxima up to approximately 15. A distribution-independent average-case bound would require a probabilistic model for arc availability and station occupancy, which is not assumed here. We therefore provide a practical-case characterization instead. Let $\bar{K}$ denote the mean number of candidate arcs per task and $\bar{C}$ the mean number of committed station arcs inspected during feasibility checking. The dominant scheduling work is approximately proportional to $N\bar{K}\bar{C}$. In the evaluated instances, $\bar{K} \approx$ 6–7 (maximum approximately 15), station occupancy is constrained by the finite 6 h horizon, and the observed scheduling runtime grows approximately linearly over $n =$ 100–500. This is not presented as a formal average-case proof.

Empirical cost. Runtime is measured around the scheduling call only and therefore excludes conflict graph precomputation ($O(|\mathcal{P}| \log |\mathcal{P}| + |\mathcal{E}|)$), which is performed once per instance during instance construction (Section 4.2). At $n = 500$, FAMAS-G runs in approximately 0.105 s versus 0.008 s for GC (a factor of approximately 13×). Conflict-graph construction is performed once per problem instance and the resulting conflict degrees are reused throughout the scheduling pass; its cost is therefore amortized over the task-level decisions. The reported 13× ratio is a scheduling-call ratio, not an end-to-end ratio including instance construction. Because preprocessing lies outside the measured interval, this difference reflects the per-task scoring and tiebreak layers that FAMAS-G adds to the greedy backbone rather than the cost of constructing the conflict graph. All methods complete well under the 6 h planning horizon, and the observed runtime is unlikely to constitute a computational bottleneck in the evaluated static scheduling setting. Relative to Greedy-Central, FAMAS-G adds one-time conflict-graph construction and bounded per-candidate scoring while retaining a single forward scheduling pass. The CNP baselines additionally incur repeated bidding and coordination rounds. Iterative VNS, ALNS, and metaheuristic approaches incur repeated neighborhood or population operations whose cost depends on stopping criteria and implementation. Because the cited methods use different formulations and platforms, we compare algorithmic structure rather than claim a hardware-independent runtime ordering.

4. Experimental Setup

4.1. TLE Data and Orbital Scenarios

Four independent TLE snapshots of the Starlink constellation were obtained (Table S2). E1–E3 are standard 3-line TLE format from Space-Track, covering the primary Starlink shells (Starlink-1xxx/2xxx/3xxx, orbital altitudes approximately 340–570 km). E4 is a SpaceX supplemental CSV covering a distinct orbital shell (Starlink-38xxx), providing orbital diversity.

For each TLE epoch, three planning window offsets were applied (start times: epoch date 12:00, 15:00, and 18:00 UTC), yielding 12 orbital states (4 epochs × 3 offsets). The hierarchical structure is as follows: TLE epoch → planning offset → orbital state → seed. The 4 TLE epochs constitute the highest level of statistical independence; offsets within

the same epoch share the same underlying satellite positions and are correlated. Full TLE epoch and offset details are provided in Tables S2 and S3 (Supplementary Materials).

4.2. Task Generation and Ground Stations

Three task scales were evaluated: $n \in \{100, 300, 500\}$ (where n denotes the task count throughout the experimental sections; N denotes sample size). Task weights follow a fixed distribution: 80% weight-1 (normal priority) and 20% weight-3 (urgent). Required tracking durations are drawn uniformly from $\{120, 180, 240, 300\}$ s. For each (orbital state, seed) pair, n satellites are selected uniformly without replacement, task parameters are assigned per the distributions above, visibility windows are computed via SGP4 propagation (5° minimum elevation), candidate arcs are generated at $\Delta = 30$ s discretization, and the conflict graph is pre-computed over all arcs. All four methods receive identical (tasks, arcs, conflict graph) per instance within a given experimental run, verified via `task_set_hash` fingerprinting, enforcing strict paired comparison. The fingerprint is computed within a single process and is used solely to confirm that all methods in that run operate on one shared instance; it is not a portable identifier and is not intended for cross-run or cross-platform comparison.

Ground Stations. Five ground stations with distinct geographic coordinates were used (Jiamusi, Beijing, Lhasa, Qingdao, Neimenggu), all with identical device transition time $\tau = 30$ s. Full station details are provided in Table S4 (Supplementary Materials).

Seeds. Ten pre-specified seeds (800–809) were used. These seeds were never used in any pilot, calibration, or diagnostic experiment. No seed was replaced or excluded post hoc.

4.3. Statistical Framework

Hierarchical Bootstrap. Statistical inference accounts for the nested experimental design using a three-level hierarchical bootstrap: (1) resample TLE epochs (4) with replacement; (2) within each resampled epoch, resample planning offsets (3) with replacement; (3) within each resampled (epoch, offset), resample seeds (10) with replacement. The procedure is repeated for $B = 10{,}000$ resamples with fixed bootstrap seed 42.

Confidence Intervals. 95% percentile confidence intervals are computed as $[\hat{q}_{0.025}, \hat{q}_{0.975}]$, where $\hat{q}_\alpha$ is the α-quantile of the bootstrap distribution of the mean ΔWFR.

Effect Size. Cohen's d_z for paired designs is computed at the seed level:

$$d_z = \frac{\overline{\Delta}}{\sigma_\Delta}, \tag{23}$$

where $\overline{\Delta}$ is the mean seed-level ΔWFR (FAMAS-G minus baseline) and σ_Δ is the standard deviation of seed-level differences (using $n - 1$ denominator).

p-values. Bootstrap p-values are computed as the fraction of centered bootstrap resamples whose absolute value equals or exceeds the observed point estimate.

Multiple Comparison Correction. Holm–Bonferroni correction is applied across the three task scales for the primary comparison (FAMAS-G vs. Greedy-Central). Let $p_{(1)} \le p_{(2)} \le \cdots \le p_{(m)}$ be the $m = 3$ ordered raw p-values. The step-down adjusted p-values are as follows:

$$p^*_{(k)} = \max_{j \le k} \min\bigl(1,\ (m - j + 1) \cdot p_{(j)}\bigr), \qquad k = 1, \ldots, m, \tag{24}$$

where the outer maximum enforces monotonicity of the adjusted p-values. Holm–Bonferroni correction is applied only to the pre-specified family of three primary FAMAS-G versus Greedy-Central comparisons across task scales. Secondary baseline comparisons and mechanism-ablation analyses are treated as exploratory; their p-values are reported nominally without an additional multiplicity correction and should be interpreted accordingly.

Sensitivity Analyses. We perform per-TLE-epoch disaggregation ($N = 30$ per cell), leave-one-epoch-out analysis ($N = 90$ per fold), and per-orbital-state disaggregation ($N = 10$ per cell).

4.4. Ablation Design and Reproducibility

4.4.1. Ablation Experiment Design

To isolate the contribution of each mechanism, five FAMAS-G configurations were evaluated on identical instances using the same 4 epochs × 3 offsets × 10 seeds × 3 scales design. The configurations form a leave-one-out ablation: FAMAS-G (full, all three mechanisms enabled), FG-NoConflict (conflict tiebreak disabled), FG-NoUrgent (urgent priority disabled), FG-NoRisk (risk tiebreak disabled), and FG-NoTiebreak (both conflict and risk tiebreaks disabled). Full configuration details are provided in Table S5 (Supplementary Materials).

4.4.2. Configuration

The frozen FAMAS-G configuration used in all experiments is provided in Table S1 (Supplementary Materials). Key parameters include the follows: primary score proximity threshold $\tau_{\text{threshold}} = 0.05$, maximum conflict adjustment $\tau_{\text{max\_adj}} = 0.10$, urgent bonus maximum $\beta_{\text{urgent}} = 0.05$, risk adjustment maximum $\tau_{\text{risk\_max\_adj}} = 0.10$, planning period $T_{\text{period}} = 6$ h, and station transition time $\tau = 30$ s. This configuration was locked before the primary confirmatory experiment and was never modified based on results; its fingerprint is `8710d0c4c977c567`, the first 16 hexadecimal characters of the SHA-256 digest of the configuration's canonical JSON serialization.

4.4.3. Reproducibility

All algorithm implementations, experimental data (SHA256: `08f2f7b701e4728a fee805a181a5ca71a0faeb3096ea18ca1abd17c16c7b9743e`), statistical analysis code, and the frozen FAMAS-G configuration (fingerprint `8710d0c4c977c567`, the first 16 hexadecimal characters of its SHA-256 digest) are available as described in the Data Availability Statement. The experimental environment is Python 3.12.4, NumPy 1.26.4, SciPy 1.13.1, on Windows 11 AMD64.

4.5. Exact Optimality Benchmark Design

To quantify the empirical optimality gap of the greedy heuristics, a supplementary small-scale benchmark was pre-specified (design locked before execution) and run after the primary experiment. Instances with $n \in \{20, 30, 50\}$ tasks (satellite count equal to task count, as in the primary pipeline) were generated for all 12 orbital states with four pre-specified seeds from the primary range (800, 803, 806, 809). On each instance, Greedy-Central and FAMAS-G (frozen configuration) were executed exactly as in the primary pipeline, and the exact optimum of the conflict-graph MWIS model (4)–(7) was computed with a MIP formulation (binary arc variables, one-arc-per-task and conflict-edge constraints) solved by the open-source CBC solver [27] through the PuLP interface, with a per-instance time limit of 120 s; optimality was certified by the solver status `Optimal`. Per the pre-specified reportability rule, a scale is summarized only if all 48 of its instances (12 states × 4 seeds) are generated successfully and solved to certified optimality. The $n = 20$ scale was not summarized because four E4_O1 instances could not be generated under the pre-specified satellite-selection rule: the selected satellites had no visibility windows to the five ground stations in the planning interval—a data characteristic, not a MIP solver failure. We therefore did not alter the sampling rule or replace satellites post hoc, and no $n = 20$ summary is reported. All 48 instances at each of $n = 30$ and $n = 50$ were generated and solved to certified optimality (maximum MIP runtime 19.6 s). Per-scale summaries are reported in Section 5.9, and instance-level results are provided as Supplementary Table S8.

4.6. Parameter Sensitivity Analysis

Following the primary experiment, a post hoc exploratory sensitivity analysis examined the two gating parameters around their frozen values: $\tau_{\text{threshold}} \in \{0.02, 0.05, 0.08\}$ and $\tau_{\text{max\_adj}} \in \{0.05, 0.10, 0.15\}$, a 3×3 grid whose center is the frozen configuration. All nine configurations were run on $n = 300$ instances across all 12 orbital states and all 10 primary seeds (120 instances), with Greedy-Central run once per instance and the nine FAMAS-G variants applied to the identical (tasks, arcs, conflict graph) triple. For each configuration, the mean ΔWFR (FAMAS-G variant minus Greedy-Central) and its 95% confidence interval were computed with the same 3-level hierarchical bootstrap as the primary analysis ($B = 10{,}000$, seed 42). As a sanity check, the frozen center configuration reproduced the primary $n = 300$ result exactly (mean $\Delta\text{WFR} = +0.0051$, 95% CI $[+0.0026, +0.0077]$). This analysis is exploratory: no parameter was re-tuned on its basis and the frozen configuration narrative is unchanged. Results are reported in Section 5.10 and Supplementary Table S7.

4.7. Evaluation Metrics

Primary Metric. Weighted Fulfillment Rate (WFR), as defined in (5), computed identically across all methods.

Secondary Metric. Wall-clock runtime measured via `time.perf_counter()`, excluding data loading and instance construction. Reported in seconds.

5. Results

5.1. Overall Scheduling Performance

Table 2 summarizes the mean WFR and runtime for all method $\times$ scale combinations ($N = 120$ per cell: 4 TLE epochs $\times$ 3 offsets $\times$ 10 seeds). Figure 2 visualizes the same comparison with ± 1 SD error bars across the $N = 120$ instances per method–scale cell. FAMAS-G achieves the highest mean WFR at every scale (0.8462, 0.6921, 0.6085), followed by Greedy-Central (0.8351, 0.6870, 0.6061). The absolute WFR difference between FG and GC narrows from 0.0111 at $n = 100$ to 0.0024 at $n = 500$. Base-CNP and FAMAS (original) achieve substantially lower WFR at all scales. All 1440 runs completed with 0 errors, 0 validator failures, and 0 paired hash mismatches.

Table 2. Mean WFR (standard deviation) and runtime at $n = 500$ for all methods and task scales.

Method	$n = 100$ WFR	$n = 300$ WFR	$n = 500$ WFR	Runtime $n = 500$ (s)
FAMAS-G	0.8462 (0.0549)	0.6921 (0.0240)	0.6085 (0.0175)	0.105 (0.008)
Greedy-Central	0.8351 (0.0519)	0.6870 (0.0230)	0.6061 (0.0166)	0.008 (0.001)
Base-CNP	0.7715 (0.0688)	0.5859 (0.0320)	0.4954 (0.0227)	0.116 (0.010)
FAMAS (original)	0.4362 (0.0572)	0.3853 (0.0269)	0.3386 (0.0211)	0.188 (0.012)

5.2. Primary Comparison: FAMAS-G vs. Greedy-Central

Table 3 reports the primary comparison using hierarchical bootstrap ($B = 10{,}000$, three-level cluster-robust). Figure 3 shows the point estimates and hierarchical-bootstrap 95% confidence intervals at all three scales. FAMAS-G outperforms Greedy-Central with Holm-corrected statistical significance at all three scales.

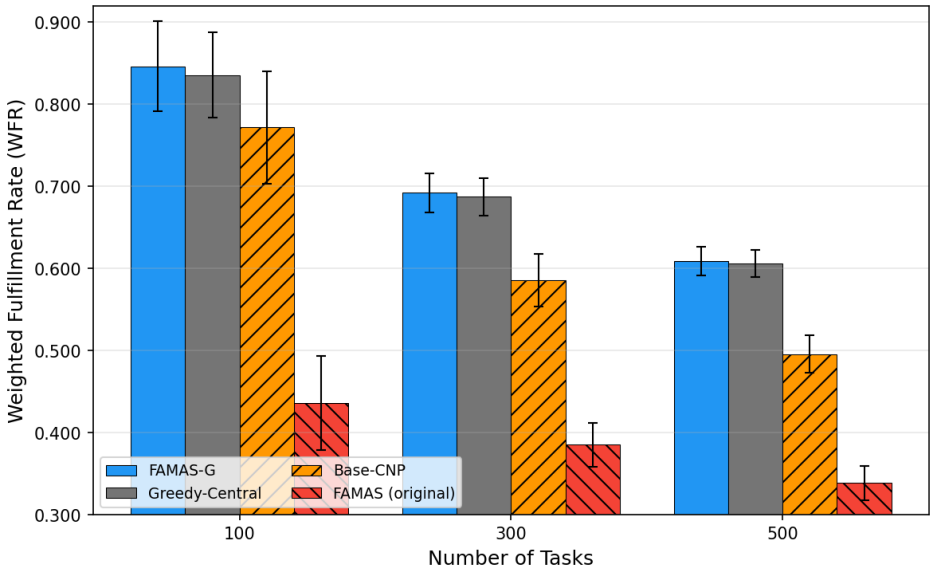

Figure 2. Weighted Fulfillment Rate (WFR) by task scale for all four methods. Error bars denote ± 1 standard deviation across $N = 120$ scheduling instances per method–scale cell.

Table 3. Hierarchical bootstrap results for the primary comparison (FAMAS-G − Greedy-Central).

Scale	ΔWFR	95% CI	Cohen's d_z	Holm-adj p	Win Rate
100	+0.0111	[+0.0069, +0.0151]	0.70	<0.0001	70.8%
300	+0.0051	[+0.0026, +0.0077]	0.49	<0.0001	66.7%
500	+0.0024	[+0.0006, +0.0042]	0.29	0.0111	60.0%

At $n = 100$, $\Delta\text{WFR} = +0.0111$, 95% CI $[+0.0069, +0.0151]$, Cohen's $d_z = 0.70$ (medium-to-large effect), Holm-adjusted $p < 0.0001$. FAMAS-G wins in 85 of 120 seed-level pairs (70.8%), loses in 17 (14.2%), and ties in 18 (15.0%).

At $n = 300$, $\Delta\text{WFR} = +0.0051$, 95% CI $[+0.0026, +0.0077]$, $d_z = 0.49$ (medium effect), $p < 0.0001$, win rate 66.7% (80/120).

At $n = 500$, $\Delta\text{WFR} = +0.0024$ (+0.24 percentage points), 95% CI $[+0.0006, +0.0042]$, $d_z = 0.29$ (small effect), Holm-adjusted $p = 0.0111$. The point estimate is positive and the 95% CI does not cross zero, providing statistical evidence of an aggregate advantage. However, the effect magnitude is small: the mean improvement of +0.0024 WFR corresponds to approximately 1–2 additional scheduled weight-units per instance (total task weight ≈ 700 across 500 tasks with 80% $w = 1$, 20% $w = 3$). The 95% CI lower bound (+0.0006) is close to zero, and the win rate of 60.0% (72/120 wins) is accompanied by a 35.8% loss rate (43/120) and 4.2% ties (5/120), indicating that FAMAS-G does not outperform Greedy-Central in every instance. Per-epoch disaggregation (Table 4) shows positive mean ΔWFR in all four epochs at $n = 500$, ranging from +0.0019 (E4) to +0.0034 (E2). Figure 4 displays the per-epoch point estimates with ± 1 SE error bars. At the orbital-state level (Figure S1), FG outperforms GC in 10 of 12 states; two states show near-zero or slightly negative mean differences (E1_O3: −0.0011, E3_O1: −0.0002), each based on 10 seed pairs. The advantage at $n = 500$ is thus best characterized as a small but consistently positive aggregate improvement within the evaluated conditions, with greater sensitivity to orbital epoch and individual state geometry than at lower task scales.

Table 4. Mean ΔWFR (FAMAS-G − Greedy-Central) within each TLE epoch.

Epoch	Orbital Shell	$n = 100$	$n = 300$	$n = 500$
E1 (Jul 7)	Starlink-1xxx/2xxx/3xxx	+0.0127	+0.0057	+0.0022
E2 (Jul 14)	Starlink-1xxx/2xxx/3xxx	+0.0115	+0.0055	+0.0034
E3 (Jul 21)	Starlink-1xxx/2xxx/3xxx	+0.0129	+0.0029	+0.0021
E4 (Jul 23)	Starlink-38xxx	+0.0072	+0.0063	+0.0019

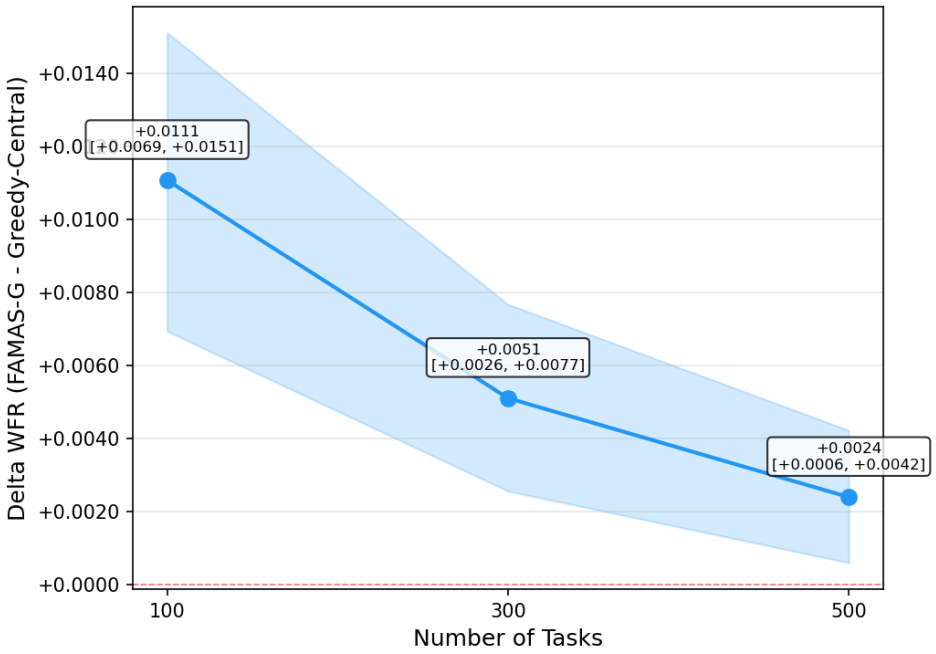

Figure 3. Point estimates and 95% hierarchical bootstrap confidence intervals for ΔWFR (FAMAS-G − Greedy-Central) across task scales.

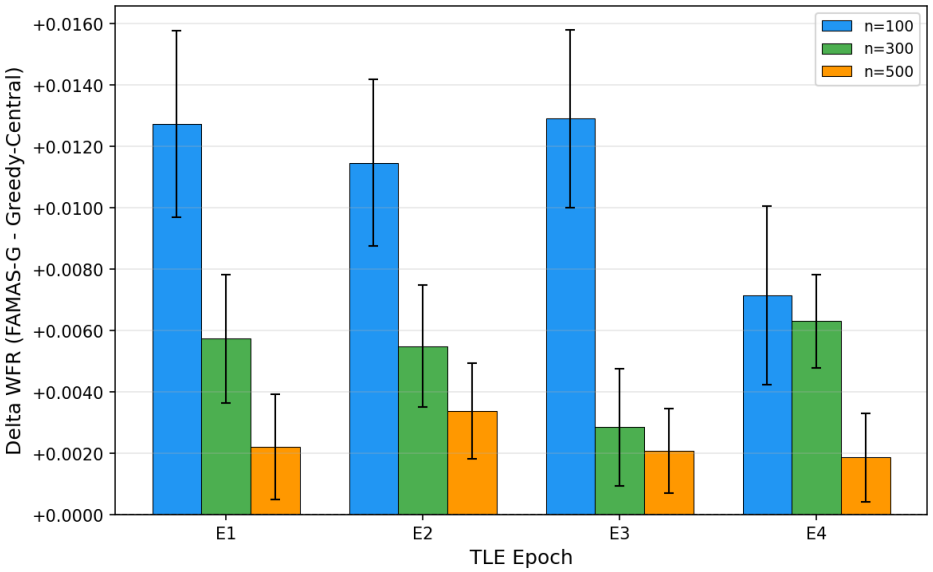

Figure 4. Per-epoch ΔWFR (FAMAS-G − Greedy-Central) across four Starlink TLE epochs. Error bars denote ±1 standard error.

5.3. Secondary Comparisons

Table 5 reports the secondary comparisons. All six comparisons yield $p < 0.0001$. The FG vs. Base-CNP advantage widens from +0.0747 ($n = 100$) to +0.1131 ($n = 500$), with Cohen's d_z increasing from 2.07 to 7.32, reflecting Base-CNP's increasing difficulty in

coordinating under higher task density without conflict awareness. The FG vs. FAMAS (original) advantage narrows from +0.4099 to +0.2698, but d_z increases from 6.26 to 12.76 due to decreasing variance. FAMAS (original) fails to schedule approximately 56–66% of weighted tasks, confirming that its unbounded linear conflict penalty degrades rather than improves performance.

Table 5. Secondary comparisons: FAMAS-G vs. Base-CNP and FAMAS-G vs. FAMAS (original).

Comparison	Scale	ΔWFR	Cohen's d_z	p
FG vs. Base-CNP	100	+0.0747	2.07	<0.0001
FG vs. Base-CNP	300	+0.1062	4.35	<0.0001
FG vs. Base-CNP	500	+0.1131	7.32	<0.0001
FG vs. FAMAS (orig.)	100	+0.4099	6.26	<0.0001
FG vs. FAMAS (orig.)	300	+0.3068	10.21	<0.0001
FG vs. FAMAS (orig.)	500	+0.2698	12.76	<0.0001

5.4. Scale-Dependent Attenuation

The ΔWFR between FAMAS-G and Greedy-Central narrows systematically with an increase in task scale: $+0.0111 \rightarrow +0.0051 \rightarrow +0.0024$. The cross-scale slope is approximately -0.0022 ΔWFR per 100 additional tasks. The 95% CIs overlap only marginally between $n = 100$ and $n = 300$ (shared interval $[+0.0069, +0.0077]$) but overlap substantially between $n = 300$ and $n = 500$ (shared interval $[+0.0026, +0.0042]$); the cross-scale point estimates should therefore not be read as formally separated. The standard deviation of seed-level differences decreases from 0.0158 ($n = 100$) to 0.0082 ($n = 500$), indicating the reduced mean difference is not an artifact of increased variance.

At $n = 500$, the attenuation is accompanied by increased instance-level heterogeneity. Per-epoch analysis (Table 4) shows that E2 contributes the largest mean advantage (+0.0034, 22/30 wins) while E4 contributes the smallest (+0.0019, 18/30 wins). At the orbital-state level, the FG advantage ranges from -0.0011 (E1_O3, 4 wins vs. 6 losses) to +0.0053 (E1_O1, 6 wins vs. 4 losses), a spread of 0.0064 WFR units that is larger than the aggregate ΔWFR of +0.0024. The leave-one-epoch-out analysis (Table 6) confirms sensitivity to epoch composition: the without-E2 fold yields $p = 0.0538$, indicating that statistical significance at $n = 500$ is not fully robust to removal of the strongest-advantage epoch.

Table 6. Leave-one-epoch-out sensitivity analysis.

Held-Out	$n = 100$	p	$n = 300$	p	$n = 500$	p
Without E1	+0.0105	<0.0001	+0.0049	0.0015	+0.0024	0.0126
Without E2	+0.0109	<0.0001	+0.0050	0.0015	+0.0021	0.0538
Without E3	+0.0105	<0.0001	+0.0058	<0.0001	+0.0025	0.0266
Without E4	+0.0124	<0.0001	+0.0047	0.0022	+0.0026	0.0204

5.5. Cross-Epoch Robustness

Table 4 reports per-epoch disaggregation ($N = 30$ per cell: 3 offsets × 10 seeds). All 12 epoch × scale cells show positive ΔWFR. The weakest cells are E4 at $n = 100$ (+0.0072) and E3 at $n = 300$ (+0.0029). At $n = 500$, per-epoch ΔWFR values range from +0.0019 (E4) to +0.0034 (E2), all within a narrow band. E4, representing the distinct Starlink-38xxx orbital shell, shows the smallest advantage at $n = 100$ and $n = 500$ but the largest at $n = 300$.

Table 6 reports the leave-one-epoch-out (LOO) sensitivity analysis ($N = 90$ per fold: 3 remaining epochs × 3 offsets × 10 seeds). At $n = 100$ and $n = 300$, all 8 LOO folds yield $p < 0.01$, indicating robustness at these scales. At $n = 500$, 3 of 4 folds are

significant at $\alpha = 0.05$; one fold (without E2) is not significant (ΔWFR $= +0.0021$, 95% CI $[-0.0001, +0.0041]$, $p = 0.0538$). The point estimates are consistent across folds (range: $+0.0021$ to $+0.0026$), indicating that the marginal significance is driven by reduced statistical power with 3 epochs rather than by any single epoch dominating the result.

5.6. Ablation Study: Mechanism Decomposition

Table 7 reports the ablation results at $n = 500$ (paired comparison, $N = 120$ per configuration, $B = 10{,}000$ bootstrap resamples). Figure 5 visualizes the mechanism-contribution decomposition (full configuration minus ablated variant). The conflict tiebreak mechanism is the dominant positive contributor to FAMAS-G's advantage over Greedy-Central:

- Conflict Tiebreak (FG-NoConflict vs. Full): ΔWFR $= -0.0097$, 95% CI $[-0.0113, -0.0081]$, $p < 0.0001$, $d_z = 1.48$. The full configuration outperforms the no-conflict variant in 105 of 120 pairs (87.5%). This is the largest single-mechanism effect observed.
- Urgent Priority (FG-NoUrgent vs. Full): ΔWFR $= -0.0001$, $p = 0.39$. The distributions are nearly identical: 106 of 120 pairs (88.3%) are ties (9 wins, 5 losses for the full configuration).
- Risk Tiebreak (FG-NoRisk vs. Full): ΔWFR $= +0.0042$, 95% CI $[+0.0026, +0.0057]$, $p < 0.0001$, $d_z = -0.55$, significantly favoring the no-risk variant. The full configuration outperforms the no-risk variant in only 29 of 120 pairs (24.2%).
- Combined Tiebreak (FG-NoTiebreak vs. Full): ΔWFR $= -0.0019$, $p = 0.032$, $d_z = 0.23$. FG-NoTiebreak yields WFR within 0.0005 of Greedy-Central (0.6066 vs. 0.6061 at $n = 500$), confirming that when both tiebreaking mechanisms are disabled, FAMAS-G's primary scoring and feasibility filter are functionally equivalent to GC.

Table 7. Ablation results at $n = 500$ ($N = 120$ per configuration).

Configuration	WFR	Δvs Full	Cohen's d_z	p
FAMAS-G (full)	0.6085	—	—	—
FG-NoConflict	0.5988	−0.0097	1.48	<0.0001
FG-NoUrgent	0.6084	−0.0001	0.10	0.39
FG-NoRisk	0.6127	+0.0042	−0.55	<0.0001
FG-NoTiebreak	0.6066	−0.0019	0.23	0.032

The magnitude of the conflict tiebreak effect also narrows with scale: Δ(FG − FG-NoConflict) $= +0.0192$ ($n = 100$) $\rightarrow +0.0129$ ($n = 300$) $\rightarrow +0.0097$ ($n = 500$), mirroring the primary FG–GC narrowing trend. At $n = 100$, urgent priority again shows a null effect ($p = 0.98$), whereas the risk tiebreak shows a significant effect, ΔWFR $= +0.0066$ in favor of the no-risk variant ($p < 0.0001$). This risk-tiebreak effect persists at $n = 300$ ($+0.0053$, $p < 0.0001$) and $n = 500$ ($+0.0042$, $p < 0.0001$).

5.7. Contention-Induced Convergence

Table 8 reports contention metrics. Figure 6 illustrates the resulting convergence pattern: as contention increases, the FAMAS-G advantage over Greedy-Central narrows while the residual gap of Greedy-Central shrinks. We quantify contention using the mean conflict degree derived from the conflict graph $(\mathcal{P}, \mathcal{E})$:

$$\bar{c} = \frac{1}{|\mathcal{P}_{\text{avail}}|} \sum_{p \in \mathcal{P}_{\text{avail}}} c(p), \tag{25}$$

where $\mathcal{P}_{\text{avail}} \subseteq \mathcal{P}$ is the set of available candidate arcs (arcs that pass the availability check and have at least one compatible station) and $c(p)$ is the conflict degree of arc p.

The mean conflict degree per arc increases by a factor of approximately 2.7 from $n = 100$ to $n = 500$ ($\bar{c} = 9.5 \rightarrow 25.9$), reflecting the combinatorial growth of temporal overlaps in denser task sets. The FG win rate decreases from 70.8% to 60.0%. This pattern—increasing contention, decreasing WFR, and diminishing marginal advantage of conflict-aware selection—is referred to as contention-induced convergence. As task density increases and more arcs compete for the same station time, the feasible scheduling space contracts, and the conflict tiebreak has less room to discriminate among alternatives.

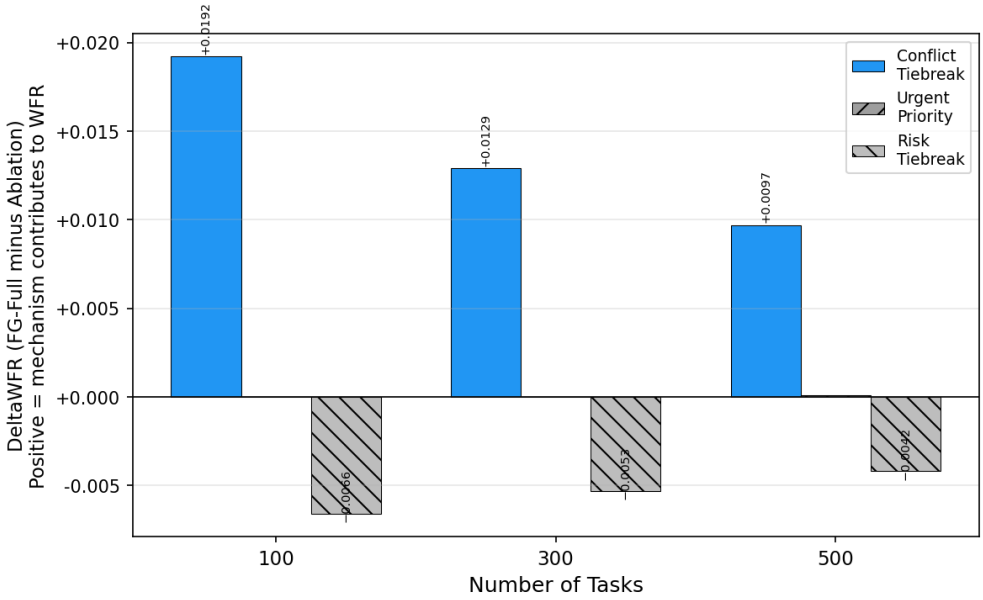

Figure 5. Mechanism contribution decomposition: ΔWFR (FAMAS-G full configuration minus ablated variant). Positive values indicate the mechanism contributes to WFR.

Table 8. Contention metrics across task scales.

Metric	$n = 100$	$n = 300$	$n = 500$
Mean conflict degree per arc	9.5	18.8	25.9
GC WFR	0.8351	0.6870	0.6061
ΔWFR (FG − GC)	+0.0111	+0.0051	+0.0024
FG win rate (seed-level)	70.8%	66.7%	60.0%

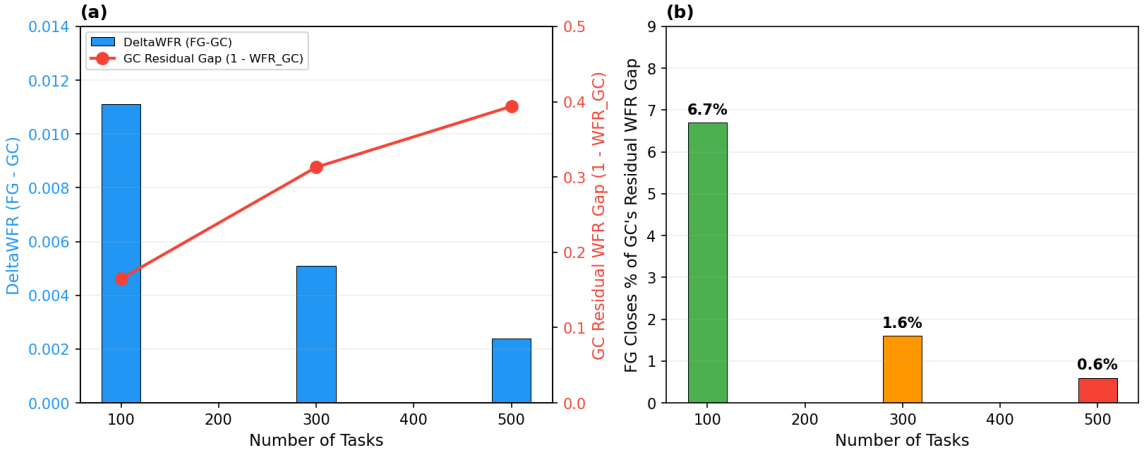

Figure 6. Contention-induced convergence. (**a**) ΔWFR (FAMAS-G − Greedy-Central, left axis) and GC residual WFR gap ($1 - \text{WFR}_{\text{GC}}$, right axis) across task scales. (**b**) Percentage of GC's residual gap closed by FAMAS-G.

5.8. Computational Cost

Table 9 reports the mean wall-clock runtime (SD in parentheses). Figure 7 plots the runtime-versus-scale trend for all four methods, with a zoomed inset for FAMAS-G and Greedy-Central. FAMAS-G completes a 500-task instance in 0.105 s on average—approximately 13× slower than Greedy-Central (0.008 s) but faster than both Base-CNP (0.116 s) and FAMAS original (0.188 s). The absolute runtime of approximately 0.1 s represents roughly 0.0005% of the 6 h planning horizon. Runtime scaling is approximately linear in the number of tasks for FAMAS-G and GC. Measurements were obtained on Windows 11, Python 3.12.4, NumPy 1.26.4; relative comparisons are more informative than absolute values. Thus, although FAMAS-G is approximately 13.2× slower than Greedy-Central for the scheduling call at $n = 500$, its mean absolute scheduling latency remains about 0.1 s in the evaluated static instances. The practical acceptability of this additional latency is discussed together with the corresponding WFR gain in Section 6.

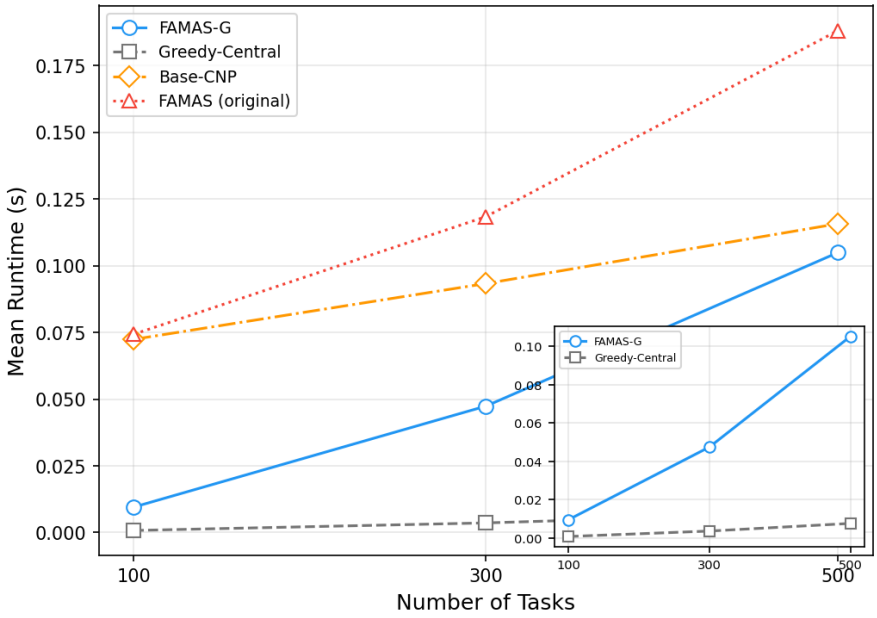

Figure 7. Mean runtime versus task scale for all four methods. The inset provides a zoomed view of FAMAS-G vs. Greedy-Central.

Table 9. Mean runtime (seconds) by method and task scale.

Method	$n = 100$	$n = 300$	$n = 500$	**Ratio vs. GC, Scheduling Call ($n = 500$)**
FAMAS-G	0.010 (0.002)	0.047 (0.005)	0.105 (0.008)	13.2×
Greedy-Central	0.001 (0.000)	0.004 (0.000)	0.008 (0.001)	1.0×
Base-CNP	0.072 (0.010)	0.093 (0.010)	0.116 (0.010)	14.5×
FAMAS (original)	0.074 (0.010)	0.118 (0.011)	0.188 (0.012)	23.5×

5.9. Exact Optimality Benchmark

Table 10 summarizes the empirical optimality gaps of the two greedy heuristics relative to the certified MIP optimum on the reportable small scales (Section 4.5).

Table 10. Small-scale exact benchmark: mean empirical optimality gap of FAMAS-G and Greedy-Central relative to the certified MIP optimum, and mean ΔWFR (FAMAS-G − Greedy-Central). All 48 instances per scale generated and certified optimal; maximum MIP runtime 19.6 s.

Scale	Mean Gap FAMAS-G	Mean Gap Greedy-Central	Mean ΔWFR	Certified
$n = 30$	1.98%	3.45%	+0.0127	48/48
$n = 50$	3.77%	4.18%	+0.0047	48/48

At both tractable scales, FAMAS-G shows a smaller empirical optimality gap than Greedy-Central (1.98% vs. 3.45% at $n = 30$; 3.77% vs. 4.18% at $n = 50$). Both gaps grow with scale, and the FAMAS-G–GC ΔWFR advantage narrows ($+0.0127 \rightarrow +0.0047$), consistent with the contention-induced convergence observed in the primary experiment. These results do not establish near-optimality; rather, they show that the conflict-aware tiebreak can improve the empirical quality of greedy decisions on small instances where the global optimum is computable. The $n = 20$ scale is not summarized for the reason stated in Section 4.5; instance-level results are provided in Supplementary Table S8.

5.10. Parameter Sensitivity

Supplementary Table S7 reports the mean ΔWFR and its 95% CI for each of the nine grid configurations (Section 4.6). The qualitative FAMAS-G advantage over Greedy-Central persists in 8 of 9 configurations; the only configuration with a negative mean ΔWFR is the most restrictive combination ($\tau_{\text{threshold}} = 0.02$, $\tau_{\text{max\_adj}} = 0.05$; mean -0.0028, 95% CI $[-0.0054, -0.0001]$). One-at-a-time perturbations around the frozen configuration keep a positive direction: the mean ΔWFR ranges from $+0.0007$ to $+0.0076$ when a single parameter is varied at a time. The mean advantage increases monotonically with $\tau_{\text{max\_adj}}$ at every $\tau_{\text{threshold}}$ level, while its dependence on $\tau_{\text{threshold}}$ is weaker at the two higher adjustment caps. These results support local robustness around the frozen configuration; they do not establish insensitivity to arbitrary parameter settings, since very small adjustment caps ($\tau_{\text{max\_adj}} = 0.05$) may eliminate the advantage and the only negative mean occurs when both parameters take lower values simultaneously.

6. Discussion

6.1. Why Conflict-Aware Selection Works

The ablation results establish that the conflict tiebreak mechanism is the dominant positive contributor within the tested configuration. Its effectiveness is consistent with the structural properties of the scheduling problem.

When candidate arcs have nearly identical primary scores (similar start times on different stations), Greedy-Central selects arbitrarily based on deterministic secondary keys. These arbitrary choices can inadvertently consume arcs that are critical for subsequent tasks, creating avoidable conflicts. The conflict tiebreak replaces this arbitrary choice with a principled one: among arcs whose primary scores lie within approximately 5% of the best primary score magnitude, prefer arcs with lower conflict degree.

Lower-conflict arcs, by definition, overlap temporally with fewer other candidate arcs on the same station. Selecting them preserves more scheduling flexibility for subsequent tasks. The threshold gate ensures the mechanism intervenes only when primary scores are genuinely ambiguous, and the bounded adjustment (magnitude capped at 10% of the reference primary score; $|\delta_{\text{conflict}}| \leq 0.10 \cdot |s^{\star}_{\text{prim}}|$) prevents pathological deviations from the greedy baseline.

This interpretation is consistent with the following: (1) the conflict graph quantifying pairwise arc incompatibility; (2) the conflict degree aggregating this information per arc;

(3) the threshold gate ensuring conflict awareness activates only when primary scores cannot clearly discriminate; and (4) the ablation confirming that disabling the conflict tiebreak produces the largest performance degradation among the tested mechanism removals. We note that this is a mechanistic interpretation, not a causally proven explanation—the experimental design does not directly manipulate scheduling flexibility.

Regarding the global-optimality limitation discussed in Section 2.2, the small-scale exact benchmark quantifies rather than eliminates the residual suboptimality of the greedy backbone. At $n = 30$ and $n = 50$, FAMAS-G exhibited smaller mean optimality gaps than Greedy-Central (1.98% vs. 3.45% and 3.77% vs. 4.18%, respectively; Table 10). These results do not establish near-optimality; they show that the conflict-aware tiebreak can improve the empirical quality of greedy decisions on small instances where the global optimum is computable, and that the gap remains strictly positive for both methods. Exact solving is not feasible at the primary scales ($n = 100$–500), so the small-scale results do not extrapolate; they characterize the evaluated regime only.

6.2. Why Urgent Priority and the Risk Tiebreak Do Not Help

The ablation results distinguish the two auxiliary mechanisms: urgent priority is indistinguishable from zero ($\Delta\text{WFR} = -0.0001$ at $n = 500$, $p = 0.39$; similarly null at $n = 100$ and $n = 300$, $p = 0.98$ and $p = 0.76$), whereas the risk tiebreak shows a small but statistically significant negative effect, favoring the no-risk variant at every scale ($\Delta\text{WFR} = +0.0066$, $+0.0053$, and $+0.0042$ at $n = 100$, 300, and 500; all $p < 0.0001$). Both patterns have identifiable structural explanations.

Urgent Priority. Three factors may explain its null effect: (1) only 20% of tasks have weight ≥ 3, and the bonus magnitude (maximum ≈ 0.025) is small relative to typical primary score gaps between candidate arcs; (2) Greedy-Central's task ordering already prioritizes high-weight tasks, so urgent priority only affects arc selection within a task, not task ordering; (3) urgent tasks typically have few candidate arcs (1–3 stations), so primary score differences between them tend to exceed the bonus magnitude. In settings with different weight distributions (e.g., more weight-5 critical tasks), the contribution could differ, but this remains untested.

Risk Tiebreak. The risk tiebreak's consistent negative effect suggests the risk heuristic is not merely uninformative but mildly misdirected in this problem setting. Temporal edge risk penalizes arcs at edge positions, but the greedy ordering systematically selects the earliest feasible arc, so edge arcs are rarely the best-primary-score candidates. Scarcity risk penalizes arcs for tasks with few alternatives, but these scarce arcs are often the only feasible option for those tasks on that station; penalizing them steers selection away from arcs that serve tasks with no other options. The effect is small in absolute terms—at most 0.0066 WFR units at $n = 100$—but consistent in direction and statistically supported at all three scales.

These mechanisms are retained in the algorithm description for completeness and transparency, but their contributions in the evaluated setting are clearly secondary: urgent priority provides no measurable benefit, and the risk tiebreak mildly degrades WFR under the frozen configuration, while the primary contribution of FAMAS-G comes from the conflict-aware tiebreak. They should not be presented as co-equal contributions alongside the conflict tiebreak.

6.3. Why the Advantage Narrows at Higher Task Scales

The ΔWFR between FAMAS-G and Greedy-Central narrows systematically as task count increases from 100 to 500: $+0.0111 \rightarrow +0.0051 \rightarrow +0.0024$. This is consistent with a pattern we describe as contention-induced convergence:

1. Average conflict degree rises from 9.5 ($n = 100$) to 18.8 ($n = 300$) to 25.9 ($n = 500$). More arcs overlap temporally, increasing the density of the conflict graph.
2. The feasible scheduling space contracts. WFR drops from approximately 0.85 to 0.61 for all methods, reducing the "degrees of freedom" available for heuristic discrimination.
3. Conflict-aware selection loses discrimination power. When nearly all feasible arcs have high and similar conflict degrees, the conflict penalty $c(p)/D_{\max}(\text{station}(p))$ becomes similar across candidates, reducing the mechanism's ability to change rankings.
4. The greedy ordering constraint dominates. Since FAMAS-G processes tasks in the same fixed order as GC, earlier decisions lock in arc commitments regardless of conflict awareness. At high task density, so many arcs are in conflict that no early choice can substantially preserve future flexibility.

We emphasize that this association is observational: contention increases with task scale, and ΔWFR decreases with task scale, but the experimental design does not manipulate contention independently of scale. Alternative explanations—such as changes in the distribution of feasible arcs, task composition effects, or interactions between greedy ordering and instance difficulty—cannot be ruled out. We deliberately avoid the term "contention ceiling" because no valid upper bound on achievable WFR has been established. A per-station dynamic-programming (DP) relaxation was considered, but because it optimizes each station independently, it discards the cross-station assignment flexibility available to a coordinated multi-station schedule; it therefore underestimates rather than bounds the achievable WFR and does not constitute a valid upper bound. Obtaining a valid and tight bound remains an open analytical problem.

6.4. Computational Acceptability

FAMAS-G's approximately 13× runtime increase relative to Greedy-Central may appear concerning when presented as a ratio. However, three considerations make it operationally acceptable: (1) absolute runtime is sub-second (0.105 s at $n = 500$), representing roughly 0.0005% of the 6 h planning horizon; (2) FAMAS-G is faster than both decentralized alternatives (Base-CNP: 0.116 s; FAMAS original: 0.188 s); and (3) the marginal computational cost (∼0.1 s) for the marginal WFR benefit (+0.0024 at $n = 500$) should be evaluated against operational priorities. For applications where every additional scheduled task has high value, the marginal cost is negligible.

Runtime measurements were obtained on a single machine and are primarily informative for relative comparisons across methods.

The engineering relevance of the observed gain is mission-dependent. At $n = 500$, the mean improvement of +0.0024 WFR corresponds to roughly one to two additional scheduled weight-units per instance, so it should not be interpreted as a large capacity increase. Such a marginal gain can nevertheless be operationally relevant when a missed contact carries disproportionate cost, whereas for low-value bulk scheduling the same gain may not justify additional implementation complexity.

6.5. Comparison with Prior Work

Table 11 situates FAMAS-G within the methodological landscape of satellite scheduling research. The table compares representative methods across five dimensions that jointly characterize the design space: conflict granularity, threshold gating, bounded adjustment, mechanism-level ablation, and single-pass structure.

Table 11. Methodological comparison of FAMAS-G with representative prior work across five design dimensions. ✓ = explicitly present; – = not explicitly reported in the reviewed paper; Implicit = conflict handled through constraints (MIP) without explicit conflict granularity. Different hardware, implementation languages, and problem scales preclude direct runtime comparison across methods. The HSAGA completion rate originates from a different benchmark with different objective functions and instance settings; cross-method entries are therefore illustrative rather than quantitative head-to-head comparisons.

Method	Conflict Granularity	Threshold Gating	Bounded Adjustment	Mechanism Ablation	Single-Pass
GC (this work)	–	–	–	–	✓
FAMAS-G (this work)	Arc	✓	✓	✓	✓
Gooley et al. (1996) [3]	Implicit	–	–	–	–
Barbulescu et al. (2004) [1]	–	–	–	–	✓
Marinelli et al. (2011) [4]	Implicit	–	–	–	–
Chen et al. (2018) [13]	Window	–	–	–	–
Wang et al. (2024) [15]	Window	–	–	–	–
Chen et al. (2024) CPVTS [16]	Window	–	–	–	–
Liu et al. (2024) [17]	Window	–	–	–	–
Cao et al. (2025) [14]	Post hoc	–	–	–	–

Two distinctions merit emphasis. First, among methods that incorporate conflict information, FAMAS-G is distinguished by operating at the arc level within a single-pass greedy framework. Chen et al. [13] compute conflict indicators at the window (mission-resource pair) level and use them to guide a differential evolution improvement phase; Chen et al. [16] similarly employ conflict-priority indicators within a variable neighborhood tabu search, where conflict awareness guides neighborhood moves rather than single-pass arc selection; Cao et al. [14] resolve conflicts through post hoc arc replacement after an initial schedule is constructed; and Wang et al. [15] employ variable neighborhood search with adaptive operator selection. These approaches separate conflict handling from the primary scheduling pass, whereas FAMAS-G integrates conflict awareness directly into the arc selection decision.

Second, the threshold gate and bounded adjustment properties, in combination, are distinctive to FAMAS-G among the methods reviewed here. The threshold gate ($\tau_{\text{threshold}} = 0.05$) ensures that conflict awareness activates only when primary scores cannot discriminate among candidate arcs—preserving the greedy heuristic's primary objective when score differences are clear. The bounded adjustment ($|\delta_{\text{conflict}}| \leq 0.10 \cdot |s^{\star}_{\text{prim}}|$) prevents the conflict term from overriding a substantially better primary arc. Prior conflict-aware methods apply conflict considerations uniformly or through unbounded metaheuristic moves, without explicit proximity-based activation gating or adjustment magnitude caps.

These design properties are not claimed to produce superior WFR compared to metaheuristic or MIP-based methods—indeed, Ma et al. [12] report >99% task completion at larger scale with HSAGA, and recent exact methods solve agile-EOS variants to optimality at moderate scale [7]. The reported HSAGA completion rate is included only as context from a different benchmark and problem formulation; it is not directly comparable with the WFR values reported in the present Starlink scheduling instances. Rather, FAMAS-G occupies a specific design point: conflict awareness at greedy speed, trading the solution quality of iterative optimization for the deterministic reproducibility and sub-second latency of a single-pass heuristic. This point is practically relevant when scheduling must be recomputed frequently (e.g., every planning cycle), when deterministic behavior is required for operational validation, or when computational resources at the scheduling node are constrained.

6.6. Conditions of Applicability and Limitations

The current findings establish the effectiveness of threshold-gated conflict-aware arc selection under the following conditions: Starlink LEO mega-constellation (∼10,700–10,860 satellites across multiple orbital shells at ∼340–570 km altitude); five ground stations with 30 s device transition time; 100–500 tasks per 6 h planning period with 80% normal-priority ($w = 1$) and 20% urgent ($w = 3$); and four TLE epochs spanning 16 days (7–23 July 2026), each with three planning offsets; the exact parameter configuration specified in Table S1 (Supplementary Materials).

We identify the following limitations, which define the boundary conditions for interpreting these findings:

TLE Data. Four independent TLE epochs provide orbital diversity but constitute a small sample. The 16-day window does not capture seasonal variations or long-term orbital precession. All experiments use Starlink TLE data; results may not transfer to other LEO constellations (e.g., OneWeb, Kuiper) or to MEO/GEO systems with different visibility dynamics. The evaluated mid-2026 Starlink TLE snapshots contain approximately 10,700–10,860 satellites, depending on epoch; "more than 10,000 satellites" is used only as a rounded description of this evaluated constellation scale, not as a claim about the number of operational spacecraft.

Experimental Design. The three planning offsets within each TLE epoch share the same satellite positions; the effective number of independent orbital configurations is closer to 4 (the TLE epochs) than 12 (the orbital states). With 10 seeds per orbital state, per-state estimates have limited precision and should not be individually interpreted. The largest evaluated scale is 500 tasks; behavior at 1000 tasks or beyond—whether the gap continues to narrow, stabilizes, or reverses—is unknown. Only static (one-shot) scheduling is evaluated; FAMAS-G's dynamic recovery module is not tested in this study. Dynamic recovery under disruption is a well-studied paradigm in the scheduling literature—through affected-operations rescheduling [28] and matchup scheduling strategies [29]—but its integration with threshold-gated conflict awareness remains an open question.

Methodological Scope. No cross-constellation validation was performed. The FAMAS-G parameters ($\tau_{\text{threshold}} = 0.05$, $\tau_{\text{max\_adj}} = 0.10$) were chosen based on pre-experiment diagnosis and locked before the primary experiment. A post hoc parameter-sensitivity analysis on a compact 3×3 grid around the frozen configuration (Section 4.6) supports local robustness but does not establish insensitivity to arbitrary parameter settings; the frozen configuration itself was never re-tuned on the basis of these results. An exact optimization-based comparison is available only at small scales: the supplementary MIP benchmark (Section 4.5) certifies optimal solutions for $n \in \{30, 50\}$, while exact solving at the primary scales ($n = 100$–500) remains computationally infeasible; no comparison to CP or metaheuristic implementations was performed. Contextualizing absolute WFR values relative to optimal solutions at the primary scales therefore remains an open task.

Generalization. The mechanism is not intrinsically tied to Starlink: any scheduling problem representable by candidate resource–time arcs and pairwise conflicts can in principle use the same conflict-aware tiebreak. However, heterogeneous constellations, Earth-observation missions, dynamic task arrivals, stochastic disruptions, and larger or heterogeneous ground-station networks require separate validation and may require parameter re-calibration. For dynamically arriving tasks, the conflict graph must be recomputed incrementally and a rolling-horizon variant is straightforward; for stochastic disruptions, nominal-window scheduling plus re-scheduling on perturbation mirrors the dynamic-recovery extension; and for larger ground-station networks, reduced contention shrinks but does not eliminate the tiebreak's scope.

Statistical Caveats. At $n = 500$, the effect magnitude is small ($d_z = 0.29$), corresponding to approximately 1–2 additional scheduled weight-units out of approximately 700 total. The without-E2 LOO fold at $n = 500$ yields $p = 0.0538$, indicating that statistical significance at the highest scale is not fully robust to removal of any single epoch. Secondary and ablation comparisons are not corrected for multiplicity beyond the three-scale Holm correction for the primary comparison.

These limitations do not invalidate the core findings but define their scope: the current study establishes that threshold-gated conflict-aware arc selection provides a statistically significant, positive, but modest WFR improvement over Greedy-Central scheduling for Starlink LEO constellation task scheduling with 100–500 tasks per 6 h period, and that the direction of this advantage is consistently positive across the four evaluated TLE epochs. The study does not establish universal orbital generalization, optimality of the specific parameterization, or applicability to fundamentally different scheduling scenarios.

7. Conclusions

We introduced FAMAS-G, a greedy-backbone scheduler augmented with threshold-gated conflict-aware arc selection, and evaluated it against three baselines across 100–500 task scales using four independent Starlink TLE epochs.

The key findings are as follows: (1) FAMAS-G achieves a statistically significant improvement over Greedy-Central at all three task scales (ΔWFR $= +0.0111, +0.0051, +0.0024$ at $n = 100, 300, 500$; Holm-adjusted $p < 0.05$), with the direction of advantage positive across all four evaluated TLE epochs, though statistical significance at $n = 500$ shows sensitivity to epoch inclusion. (2) Ablation analysis identifies the threshold-gated conflict tiebreak as the dominant positive contributor; urgent priority shows no measurable effect, whereas the risk tiebreak exerts a small but statistically significant negative effect at the tested scales. (3) The advantage attenuates with increasing task density, consistent with contention-induced convergence—as conflict density rises, the room for conflict-aware heuristics to discriminate among alternatives diminishes. (4) FAMAS-G completes scheduling in approximately 0.1 s at $n = 500$, demonstrating low computational latency in the evaluated static scheduling setting.

The primary contribution is the demonstration that a lightweight, threshold-gated conflict-awareness mechanism—requiring only pre-computed arc-level conflict degrees and a single threshold parameter—can provide a small but consistently positive aggregate improvement in greedy scheduling performance for satellite task scheduling within the evaluated Starlink conditions, without the overhead of message passing, backtracking, or iterative optimization.

The principal strengths of FAMAS-G are its deterministic single-pass structure, explicit arc-level conflict information, bounded intervention, and low absolute scheduling latency, together with a controlled ablation-based evaluation. Its principal limitations are equally important: it provides no global optimality guarantee, has been validated only in static Starlink-derived scenarios with fixed task-weight distributions, and has not yet been tested for parameter robustness beyond the compact grid of Section 4.6, heterogeneous constellations, dynamic arrivals, or onboard communication/propagation constraints.

Future work should extend the evaluation to additional TLE epochs from diverse constellations, broaden the parameter-sensitivity grid beyond the two gating parameters explored here, evaluate performance at larger task scales, incorporate dynamic task-weight policies (e.g., observation-history-dependent decay and revisit requirements) through the effective-weight formulation of Equation (1), test the dynamic recovery module under station-failure scenarios, empirically compare against VNS/ALNS/RL schedulers with

compatible implementations, and extend exact-optimality benchmarking toward the primary scales when computationally feasible.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/aerospace13090830/s1.

Author Contributions: Conceptualization, methodology, and supervision, X.P.; software, formal analysis, and investigation, X.L. and Y.J.; writing—original draft preparation, X.L.; writing—review and editing, X.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 62503487, and in part by the National Key Laboratory of Space Intelligent Control Technology under Grant 2024-CXPT-GF-JJ-012-16.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The experimental data and analysis codes generated during the current study are available from the corresponding author upon reasonable request.

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

Abbreviations

The following abbreviations are used in this manuscript:

WFR	Weighted Fulfillment Rate
FAMAS	Fast Adaptive Multi-Agent Scheduling
FAMAS-G	FAMAS–Greedy, with threshold-gated conflict-aware arc selection
GC	Greedy-Central
CNP	Contract Net Protocol
TBA	Task Broker Agent
GSA	Ground Station Agent
TLE	Two-Line Element
LEO	Low Earth Orbit
SGP4	Simplified General Perturbations 4
CI	Confidence Interval
LOO	Leave-One-Epoch-Out
DP	Dynamic Programming

References

1. Barbulescu, L.; Watson, J.P.; Whitley, L.D.; Howe, A.E. Scheduling Space–Ground Communications for the Air Force Satellite Control Network. *J. Sched.* **2004**, *7*, 7–34. [CrossRef]
2. Li, S.; Yu, Q.; Ding, H. Reviews and Prospects in Satellite Range Scheduling Problem. *Auton. Intell. Syst.* **2023**, *3*, 9. [CrossRef]
3. Gooley, T.D.; Borsi, J.J.; Moore, J.T. Automating Air Force Satellite Control Network (AFSCN) Scheduling. *Math. Comput. Model.* **1996**, *24*, 91–101. [CrossRef]
4. Marinelli, F.; Nocella, S.; Rossi, F.; Smriglio, S. A Lagrangian Heuristic for Satellite Range Scheduling with Resource Constraints. *Comput. Oper. Res.* **2011**, *38*, 1572–1583. [CrossRef]
5. Spangelo, S.; Cutler, J.; Gilson, K.; Cohn, A. Optimization-Based Scheduling for the Single-Satellite, Multi-Ground Station Communication Problem. *Comput. Oper. Res.* **2015**, *57*, 1–16. [CrossRef]
6. Vázquez, A.J.; Erwin, R.S. On the Tractability of Satellite Range Scheduling. *Optim. Lett.* **2015**, *9*, 311–327. [CrossRef]
7. Peng, G.; Wang, J.; Song, G.; Gunawan, A.; Xing, L.; Vansteenwegen, P. Branch-and-Cut-and-Price for Agile Earth Observation Satellite Scheduling. *Eur. J. Oper. Res.* **2025**, *326*, 427–438. [CrossRef]
8. Barbulescu, L.; Howe, A.E.; Whitley, L.D. AFSCN Scheduling: How the Problem and Solution Have Evolved. *Math. Comput. Model.* **2006**, *43*, 1023–1037. [CrossRef]
9. Kandepi, R.; Saini, H.; George, R.K.; Konduri, S.; Karidhal, R. Agile Earth Observation Satellite Constellations Scheduling for Large Area Target Imaging Using Heuristic Search. *Acta Astronaut.* **2024**, *219*, 670–677. [CrossRef]

10. Song, Y.; Ou, J.; Wu, J.; Wu, Y.; Xing, L.; Chen, Y. A Cluster-Based Genetic Optimization Method for Satellite Range Scheduling System. *Swarm Evol. Comput.* **2023**, *79*, 101316. [CrossRef]
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. [CrossRef]
12. Ma, L.; Qin, Y.; Qin, J.; Xu, M. Massive Constellation Measurement and Control Scheduling Based on Hybrid Simulated Annealing Genetic Algorithm. *J. Astronaut.* **2023**, *44*, 1757–1766.
13. Chen, X.; Reinelt, G.; Dai, G.; Wang, M. Priority-Based and Conflict-Avoidance Heuristics for Multi-Satellite Scheduling. *Appl. Soft Comput.* **2018**, *69*, 177–191. [CrossRef]
14. Cao, S.; Liu, X.; He, L. Three-Stage Hybrid Scheduling Method Based on Link Adjustment and Arc Replacement for Large-Scale Satellite Ground Station Resource Scheduling Considering Idleness Rate. *Results Eng.* **2025**, *27*, 105981. [CrossRef]
15. Wang, T.; Gu, Y.; Wang, H.; Wu, G. Adaptive Variable Neighborhood Search Algorithm with Metropolis Rule and Tabu List for Satellite Range Scheduling Problem. *Comput. Oper. Res.* **2024**, *170*, 106757. [CrossRef]
16. Chen, X.; Gao, Q.; Peng, S.; Song, S.; Liu, Y.; Dai, G.; Wang, M.; Zhang, C. A Conflict-Priority-Based Variable Neighborhood Tabu Search Method for Multi-Satellite Scheduling. *Adv. Astronaut. Sci. Technol.* **2024**, *7*, 163–176. [CrossRef]
17. Liu, Z.; Liu, J.; Liu, X.; Yang, W.; Wu, J.; Chen, Y. Knowledge-Assisted Adaptive Large Neighbourhood Search Algorithm for the Satellite–Ground Link Scheduling Problem. *Comput. Ind. Eng.* **2024**, *192*, 110219. [CrossRef]
18. Li, B.; Chen, M.; Wei, L.; Chen, Y.; Chen, Y. Solving the Time-Dependent Agile Earth Observation Satellite Scheduling Problem Using Adaptive Large Neighborhood Search with Reinforcement Learning Controller. *Appl. Soft Comput.* **2026**, *196*, 115078. [CrossRef]
19. Song, Y.; Suganthan, P.N.; Pedrycz, W.; Yan, R.; Fan, D.; Zhang, Y. Energy-Efficient Satellite Range Scheduling Using a Reinforcement Learning-Based Memetic Algorithm. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *60*, 4073–4087. [CrossRef]
20. Ma, C.; Zhou, D.; Sheng, M.; Li, H.; Li, J. Dynamic TT&C Mission Scheduling for Mega-Satellite Networks: A Deep Reinforcement Learning Approach. In *Proceedings of the IEEE Global Communications Conference (GLOBECOM 2023)*; IEEE: New York, NY, USA, 2023; pp. 7163–7168. [CrossRef]
21. Smith, R.G. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Trans. Comput.* **1980**, *C-29*, 1104–1113. [CrossRef]
22. Jiang, Y.; Gao, Y.; Yu, L.; Yu, J.; Li, Y.; Gao, H. Self-Organizing Method on Mission-Level Task Allocation of Large-Scale Remote Sensing Satellite Swarm. *Int. J. Aerosp. Eng.* **2022**, *2022*, 1–14. [CrossRef]
23. Jin, P.; Li, K. Distributed Satellite Resource Scheduling Based on Improved Contract Network Protocol. *Syst. Eng. Electron.* **2022**, *44*, 3164–3173. [CrossRef]
24. Yang, W.; He, L.; Liu, X.; Chen, Y. Onboard Coordination and Scheduling of Multiple Autonomous Satellites in an Uncertain Environment. *Adv. Space Res.* **2021**, *68*, 4505–4524. [CrossRef]
25. Xiang, Z.; Gu, Y.; Wang, X.; Wu, G. Hierarchical Disturbance Propagation Mechanism and Improved CNP for Satellite TT&C Resource Dynamic Scheduling. *Complex Syst. Model. Simul.* **2024**, *4*, 166–183. [CrossRef]
26. Krigman, S.; Grinshpoun, T.; Dery, L. Scheduling of Earth Observing Satellites Using Distributed Constraint Optimization. *J. Sched.* **2024**, *27*, 507–524. [CrossRef]
27. Forrest, J.; Lougee-Heimer, R. CBC User Guide. In *Emerging Theory, Methods, and Applications*; INFORMS TutORials in Operations Research; INFORMS: Catonsville, MD, USA, 2005; pp. 257–277. [CrossRef]
28. Abumaizar, R.J.; Svestka, J.A. Rescheduling Job Shops Under Random Disruptions. *Int. J. Prod. Res.* **1997**, *35*, 2065–2082. [CrossRef]
29. Bean, J.C.; Birge, J.R.; Mittenthal, J.; Noon, C.E. Matchup Scheduling with Multiple Resources, Release Dates and Disruptions. *Oper. Res.* **1991**, *39*, 470–483. [CrossRef]