drones

MDPI

Article

Execution-Consistent Event-Responsive Truck–UAV Coordination Under Simulated Progressive Road Disruption

Peng Lu [1,2,3], Daming Wu [2,3,*], Zixiao Li [2], Beinan Guo [2], Shaofei Lan [2,3] and Aoxin Yang [2,3]

[1] School of Emergency Management Science and Engineering, University of Chinese Academy of Sciences, Beijing 100049, China
[2] National Institute of Natural Hazards, Ministry of Emergency Management of China, Beijing 100085, China; yangaoxin24@mails.ucas.ac.cn (A.Y.)
[3] Key Laboratory of Compound and Chained Natural Hazards Dynamics, Ministry of Emergency Management of China, Beijing 100085, China
* Correspondence: damingwu@ninhm.ac.cn

Highlights

What are the main findings?

- Suffix-local repair reduced task-ownership changes, execution rebindings, and repeated reassignment relative to global re-auction, indicating greater stability of the ongoing plan.
- In the principal hard-LB comparison, ER-HLNS-PR increased time-critical completion and reduced priority-weighted censored completion time relative to a mission-priority-matched baseline, while reducing routine completion.

What are the implications of the main findings?

- Execution-aware replanning incorporates newly revealed road information while retaining service and recovery commitments already in progress.
- Truck–UAV planners should assess urgent and routine service, delivered mass, and plan stability together to make the consequences of task prioritization explicit.

Abstract

Progressively revealed road disruptions can invalidate truck–unmanned aerial vehicle (UAV) relief plans while deliveries and recovery operations are already under way. We develop ER-HLNS-PR, an event-responsive framework that reconciles realized task, vehicle, and material states, preserves active service and recovery commitments, and repairs affected unfinished routes through bounded deterministic search. The evaluation combines five scenario families, controlled comparisons, sensitivity analyses, and execution-invariant audits. In the principal comparison of 20 independent task–road instances with three repetitions per method, ER-HLNS-PR increased time-critical completion by 16.5 percentage points (descriptive 95% paired bootstrap interval: 8.2–25.0) and reduced priority-weighted censored completion time (PWCT) by 2444.2 s relative to a mission-priority-matched adaptive large-neighborhood search baseline; routine completion decreased by 11.5 percentage points. In a separate repair-scope control, suffix-local repair reduced ownership changes, execution rebindings, and repeated reassignment relative to global re-auction, while the completion and PWCT differences spanned zero. Relative to a full-route rolling-horizon comparator, weighted-service, PWCT, and runtime differences were small and their intervals included zero. Overall, the results distinguish two effects: stronger urgent-service performance relative to the mission-priority-matched baseline and lower plan disruption under local repair, alongside an explicit urgent-versus-routine service trade-off.

Check for updates

Academic Editors: Andreas Schwung, Stefan Lier, João Paulo Javidi da Costa, Edison Pignaton De Freitas and Diego González-Aguilera

Received: 2 September 2026
Revised: 9 October 2026
Accepted: 12 October 2026
Published: 16 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Keywords: truck–UAV routing; post-disaster relief; event-responsive planning; progressive road disruption; execution-state reconciliation; service contracts; bounded remove–reinsert search

1. Introduction

1.1. Operational Motivation

Post-disaster relief distribution must meet demands that differ in urgency, mass, and accessibility. Trucks carry bulk supplies but depend on usable roads, whereas unmanned aerial vehicles (UAVs) can deliver lightweight, time-critical supplies across local road breaks, subject to tighter payload, endurance, weather, launch, and recovery constraints. Humanitarian truck–UAV studies have examined disrupted infrastructure, uncertain access and demand, priority-sensitive service, communication-supported demand discovery, and uncertain travel times [1–5]. These settings require more than short routes: material allocation, vehicle state, launch permission, service eligibility, and recovery must remain consistent throughout execution.

Truck–UAV synchronization provides the routing foundation for this study. The flying-sidekick traveling-salesman problem couples a truck tour with UAV launch, delivery, and recovery; integrated formulations show that fixing the truck route before assigning aerial sorties can exclude useful coordinated solutions [6,7]. Extensions include multiple trucks or aircraft, flexible docking, multi-visit aerial tours, and synchronization during truck movement [8–13]. Surveys document a broad range of cooperation modes, constraints, objectives, and exact, heuristic, metaheuristic, and learning-based methods [14–17]. Building on these foundations, we examine how cooperative plans can be maintained as road information and ongoing operations evolve.

We consider a fixed-demand setting in which all delivery tasks exist at the start of an episode, while persistent road blockages are progressively revealed through vehicle movement and shared observations. Truck and UAV locations, docking relations, batteries, inventories, task status, route cursors, and recovery commitments evolve during the same period. Time-dependent and robust routing models address changing or uncertain travel conditions [18–21], and event-driven truck–UAV studies consider newly revealed emergency areas or randomly released requests [22,23]. Our focus is the interaction between newly available road evidence and the changing execution state of an already installed cooperative plan.

1.2. Related Work and Research Gap

Disaster-oriented truck–UAV research spans several complementary decision problems. Two-echelon humanitarian systems combine communication and delivery UAVs to reveal and serve needs after infrastructure failure [1]. Earthquake-relief formulations address uncertain road access and demand [2], while reinforcement-learning policies adapt distribution to uncertain travel times [3]. Other studies incorporate service priorities, time-varying weather, multiple depots, fairness, shortages, and combined delivery and surveillance [4,5,24–27]. Within this literature, our specific question is how replanning should treat decisions that have already become binding through execution.

Dong et al. [28] study flood-rescue truck–UAV routing with mobile launch and recovery, multi-point sorties, priority-sensitive response, payload-dependent energy, and disrupted roads. Their model and adaptive large-neighborhood search provide a tactical routing foundation. Building on that stage, we ask which unfinished decisions may still be revised when newly observed road conditions conflict with current task, vehicle, and recovery commitments.

Zhang and Li [29] combine predicted flood depth, depth-dependent truck travel, wind and rainfall, and flood-aware ALNS neighborhoods. Their predictive routing complements the present execution-stage focus, which reconciles an installed plan with the realized execution state as new road evidence becomes available.

Otto et al. [26] study online delivery and surveillance policies for a truck–drone tandem as impassable edges are revealed during a mission. This online setting is particularly close to ours; the present formulation additionally represents multiple service agents, finite inventory and energy, deadlines, and continuing mobile-recovery commitments within the execution-state update.

Li, Song and Li [30] address information acquisition through Bayesian active verification, allocating UAV capacity between road inspection and delivery under partial observability. Their framework determines whether and where to acquire road information before routing decisions. Our framework begins once such evidence is available and determines which task, vehicle, material, and recovery commitments remain binding and which unresolved route segments may be revised. The two approaches therefore address complementary stages of information acquisition and executable-plan updating.

Recent studies further broaden dynamic synchronization, uncertainty management, and execution-time adaptation. Xing et al. [31] formulate reliable truck–drone routing with dynamic synchronization, and Zhao et al. [32] allow drones to change take-off and landing trucks through a matheuristic scheduling framework. Yang et al. [33] and Xu et al. [34] address post-disaster stochastic scheduling and distributionally robust humanitarian network design, respectively, while Sun et al. [35] develop robust routing for delivery and surveillance under travel-time uncertainty. Li et al. [36] formulate progressively recovering access as a Bayesian belief-to-action problem, and Li et al. [37] study multi-stage cooperative delivery under progressive road recovery. Together, these studies show that dynamic truck–UAV coordination spans distinct decision stages and feasible decision spaces; Supplementary Table S1a,b organize the corresponding differences in service rules, capacity, launch/recovery coupling, information assumptions, and ongoing commitments.

The broader operating environment is partially observed because some road failures remain unknown until they are observed or encountered. POMDP and reinforcement-learning formulations are natural when the objective is to optimize a policy over modeled future uncertainty. This study addresses a different decision layer: once new road evidence is available, it updates the installed cooperative plan while preserving commitments already created by execution. We therefore formulate the problem as observation-conditioned constrained repair; predictive beliefs or learned policies can be coupled to this interface in future work.

Adaptive large-neighborhood search (ALNS) selects destroy and repair heuristics according to their search performance [38], and truck–UAV applications include coordinated-route, sortie, and dynamic-replanning variants [22,23,39–45]. We use this search literature as the computational foundation for an execution-focused update that identifies resolved or physically committed work and exposes only eligible unfinished decisions to repair.

Execution-aware planning also requires outcome measures that retain missed work and task-class differences. Mean completion time among delivered tasks omits unfinished tasks, while an aggregate completion rate can mask a shift from bulk routine delivery toward lightweight time-critical service. We therefore report overall and class-specific completion, together with priority-weighted censored completion time (PWCT), which assigns unfinished tasks the scenario horizon. Episodes provide measurements; for the principal comparison, algorithm repetitions are averaged within each independent task–road instance before paired inference. This design makes both missed work and service reallocation visible.

Accordingly, the research problem is to maintain an executable cooperative plan after new evidence arrives. The controller must reconcile realized road, task, vehicle, payload, and inventory states; advance route cursors past resolved work; preserve active commitments; and identify unfinished decisions that remain eligible for revision. This execution-state boundary is the central methodological focus of the study.

1.3. Proposed Approach and Research Questions

We propose ER-HLNS-PR, an event-responsive framework for maintaining executable truck–UAV plans during ongoing operations. After admitting a state update, the framework reconciles the installed plan with current road, task, vehicle, payload, and recovery states. It protects service and recovery commitments already in progress and repairs eligible unfinished route segments. A persistent service contract records task ownership and the responsible truck, UAV, material source, and recovery link until the commitment is resolved. The planning layer maintains route sequences and contracts, while the execution layer checks material, energy, weather, road, and recovery feasibility before issuing the next action. A deterministic remove–reinsert search uses four iterations, at most two eligible removals per iteration, and strict lexicographic acceptance.

The evaluation addresses four research questions. RQ1 examines overall, routine, and time-critical (TC) completion and PWCT across five scenario families. RQ2 tests the hard-LB service pattern under mission-priority matching, removal of targeted isolation, an objective-evaluation-budget control, and shared TC service eligibility. RQ3 examines repair locality, retention of recorded contract signatures, and changes in task ownership and execution bindings relative to global pending-task re-auction. RQ4 evaluates sensitivity to physical inputs, policy thresholds, lifeline decay, blockage severity, and topology.

This study makes three contributions. First, it formulates event-indexed execution-consistent replanning as a constrained update problem that partitions ongoing decisions into resolved, execution-protected, and eligible sets. Second, it develops a state–contract interface that preserves binding service and recovery commitments, restricts repair to eligible unfinished decisions, and installs route and contract updates atomically. A bounded remove–reinsert procedure provides a transparent computational realization of this interface; the methodological contribution lies in the execution-state partition and contract-preserving update. Third, the study evaluates the resulting service–stability trade-offs through a 20-instance repeated comparison, mission-priority and computational controls, a direct repair-scope experiment, sensitivity analyses, and execution-invariant audits.

The evaluation is organized to separate service allocation from plan stability, linking each claim to a direct comparison.

Figure 1 summarizes the study logic from the operational setting and execution-consistency gap to the proposed response and evaluation design.

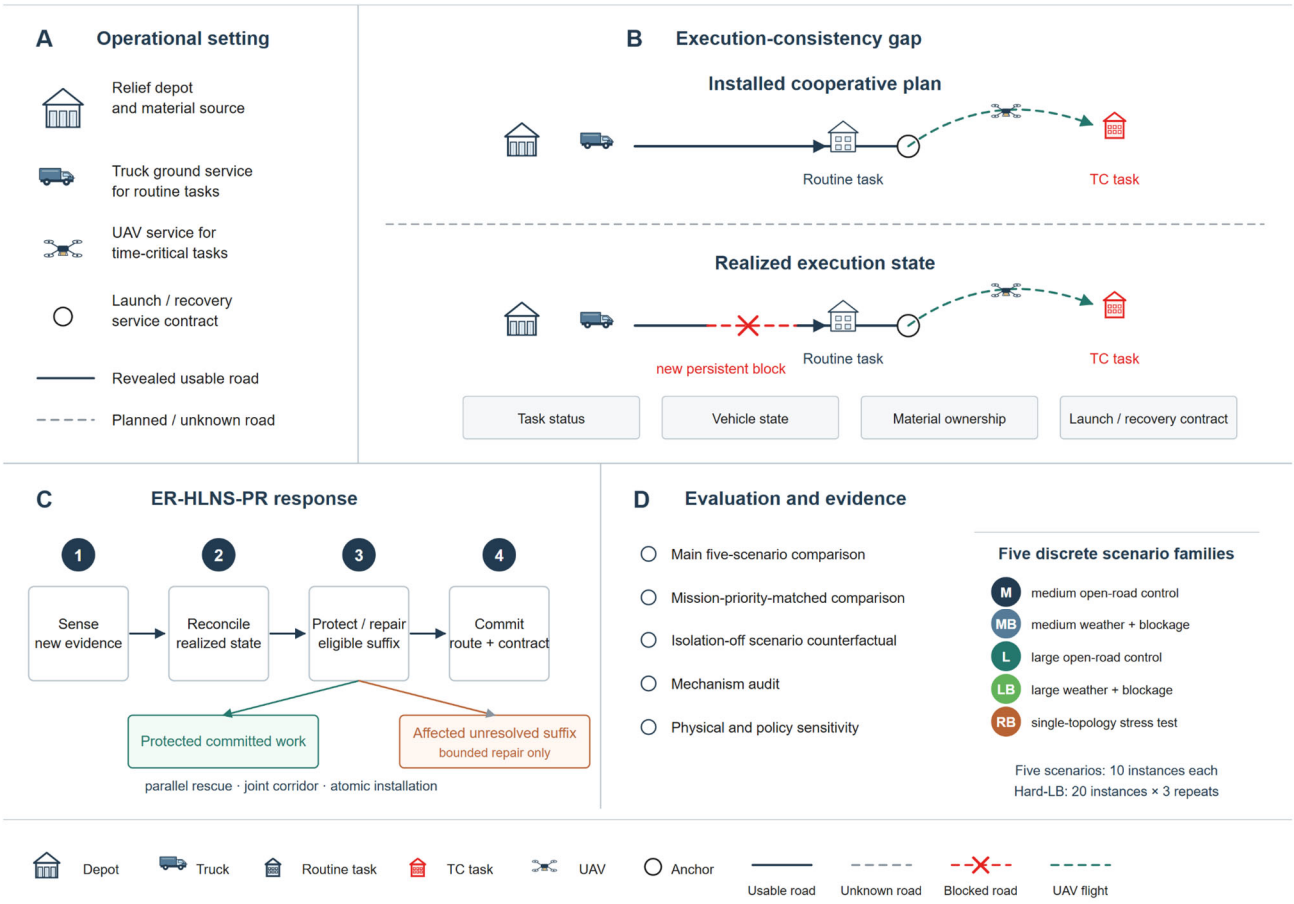


Figure 1. Study framework. (**A**) Truck–UAV relief setting; (**B**) conflict between an installed plan and newly revealed road and execution states; (**C**) state reconciliation, protection of committed work, and repair of eligible unfinished routes; and (**D**) comparison and audit design. M, MB, L, and LB use seed-specific synthetic networks, whereas RB uses one fixed Dujiangyan-derived map. The five-scenario display uses ten seed realizations per scenario; the principal hard-LB comparison uses 20 independent task–road instances with three algorithm repetitions per method and instance.

2. Materials and Methods

2.1. Operational System, Tasks, and Service Eligibility

The relief system operates on a road graph $G_t = (V, E_t)$, where V contains a depot, road junctions, and demand locations, and E_t is the physically traversable edge set at decision step t. Trucks travel only on road edges. Each truck is associated with one or more initially docked UAVs, forming a mobile service cluster. A UAV can leave its carrier only after a launch check, deliver its assigned package, and then recover to a feasible truck or depot anchor before recharging and reloading. The simulator advances in 20 s steps. Routine replanning uses a nominal five-step interval (100 s); event-triggered updates follow the admission rules in Section 2.4.1.

All demand tasks exist at episode initialization; no new requests arrive during an episode. Let $\mathcal{N} = \mathcal{N}^R \cup \mathcal{N}^C$ denote the task set. A routine-bulk task $i \in \mathcal{N}^R$ requires 800 kg, has urgency in [0.2, 0.6], and can be served only by a truck. A time-critical lightweight task $i \in \mathcal{N}^C$ requires one 50 kg package, has urgency in [0.7, 1.0], and is served by a loaded UAV agent in the main collaborative configuration. UAV bulk relay is disabled. A loaded UAV may complete a TC task either after an airborne sortie or by releasing its payload while co-located with its carrier truck; both count as TC completion, whereas the launch counter records airborne sorties only. In the E6 sensitivity, a road-reachable stocked truck may also serve a pending TC task when its shortest-path distance is at most 900 m or its

remaining deadline slack is at most 18 steps; under high pressure, the corresponding limits are 1400 m and 28 steps. This shared fallback does not interrupt airborne or in-service work and assigns at most one TC task per truck. These task classes and eligibility rules define the evaluated service policy. Delivery mode is recorded from the actual service event; a completed time-critical task is not assumed to represent an airborne sortie.

Each truck has a 3000 kg material-load capacity and initially carries 2400 kg of routine inventory and 200 kg of time-critical inventory, equivalent to three complete routine deliveries and four UAV packages. This operational payload limit is referenced to a medium 4×4 relief-truck class: an official FUSO Canter variant (Mitsubishi Fuso Truck and Bus Corporation, Kawasaki, Japan) lists 3625 kg maximum cargo [46]. The difference allows for mounting, packaging, personnel, and vehicle-body variation omitted from the model. Each UAV carries one 50 kg package, with records for payload mass, package type, source truck, recharge state, and reload status. Delivery consumes the package and reduces the payload to zero before the recovery leg; another service requires reloading from a truck. Material transferred to a UAV is deducted from its carrier's inventory. Depot replenishment remains enabled even though aggregate initial stock covers total demand. Routine tasks require the full 800 kg transfer because partial routine service is disabled.

Task state is one of pending, claimed/in service, delivered, failed, or unfinished at the episode horizon. A task is completed only after the execution layer confirms vehicle arrival, eligibility, inventory transfer, and the service transition. Each task has deadline d_i and lifeline ℓ_i. At each 20 s step, $\ell_i \leftarrow \max\{0, \ell_i - \beta_k(0.6 + u_i)[1 + 0.35(r_i/40 + w_i/15)]\}$, with $\beta_R = 0.08$, $\beta_C = 0.22$, rainfall r_i in mm h^{-1} and wind w_i in m s^{-1}. Lifeline exhaustion and deadline expiry are separate terminal mechanisms. Consequently, "not completed" includes both terminal failure and work unresolved at the horizon; the task ledger records these two outcomes separately.

Truck state comprises position, road-transit state, velocity, inventory, and the current command. UAV state additionally includes battery charge, payload, docking or follow target, sortie distance, reload status, and recovery commitment. Trucks move continuously along road edges, while UAVs move in the plane between launch, delivery, and recovery anchors. All methods use the same execution layer for motion, service, material transfer, energy use, launch, weather, recovery, and task exclusivity. These shared rules provide the common operational basis for comparison.

Initial fleet inventory covers the total modeled demand. In M/MB, two trucks hold 4800 kg of routine stock for 4800 kg of demand and 400 kg of time-critical stock for 300 kg of demand. In L/LB/RB, four trucks hold 9600 kg of routine stock for 6400 kg of demand and 800 kg of time-critical stock for 600 kg of demand. Stock can nevertheless become operationally inaccessible when it is held by a different service cluster or its carrier loses road access. The planner therefore tracks distributed material ownership rather than pooling all fleet inventory. UAV recharge and package reload follow the execution contract; depot replenishment is available but is not needed to make aggregate initial supply sufficient.

Each synthetic seed generates a road network at the M or L scale. For a given seed, M and MB share the same medium-scale topology, tasks, initial configuration, and common random-number schedule; L and LB share the corresponding large-scale realization. Seeds 100–109 thus provide ten independently generated networks at each synthetic scale, evaluated under paired open-road and disruption conditions. Routine tasks are sampled without replacement from non-depot road nodes. TC tasks are sampled from the remaining nodes using the configured mixture of uniform sampling and an epicenter-centered Gaussian bias. RB uses one fixed Dujiangyan-derived topology and facility-role task manifest; its seeds vary persistent disruption and other episode-level exogenous draws. All methods receive

the same realization within a scenario–seed block. The principal hard-LB comparison extends the large-scale set to 20 independent instances (seeds 100–119), with three algorithm repetitions per method and instance; the five-scenario display retains ten seed realizations per scenario.

2.2. Scenario Construction, Disruption, Information, and Physical Proxy

2.2.1. Scenario Families

Five scenarios were evaluated. M and L are zero-hazard controls on seed-generated medium- and large-scale synthetic networks. MB and LB use the corresponding topology, fleet, tasks, demands, deadlines, and horizon, with weather and progressive road disruption activated. RB tests execution on one fixed Dujiangyan-derived operational road graph and task manifest; its ten seeds vary disruption and other episode-level exogenous realizations. Tables 1 and 2 summarize scenario parameters.

Table 1. Scenario scale and fleet/task configuration. Each simulation step is 20 s; routine replanning is nominally every five steps (100 s).

Alias	Condition	Road Graph	Nominal Extent	Trucks/UAVs	Routine/TC Tasks	Horizon
M	M-A	80 nodes, 140 edges	5 km	2/3	6/6	560 steps
MB	M-B	80 nodes, 140 edges	5 km	2/3	6/6	560 steps
L	L-A	320 nodes, 460 edges	15 km	4/6	8/12	1440 steps
LB	L-B	320 nodes, 460 edges	15 km	4/6	8/12	1440 steps
RB	R-B	462 nodes, 709 edges	20 km	4/6	fixed 8/12	2400 steps

Table 2. Temporal and disruption configuration. Deadlines are expressed as start + interval × task index.

Alias	Routine Deadlines	TC Deadlines	Weather	Road Process	Controlled Isolation
M	$420 + 20i$	$240 + 10i$	zero-weather	open-road control	none
MB	$420 + 20i$	$240 + 10i$	12 mm h^{-1}, 6 m s^{-1}	persistent blockage	two candidates
L	$960 + 60i$	$600 + 30i$	zero-weather	open-road control	none
LB	$960 + 60i$	$600 + 30i$	12 mm h^{-1}, 6 m s^{-1}	persistent blockage	three candidates
RB	$1680 + 80i$	$960 + 40i$	wind/rain inactive	earthquake-related persistent-disruption proxy	none

Controlled-isolation metadata are retained in both members of each synthetic pair to preserve task identity. No isolation cuts are applied in the open-road controls. In MB and LB, incident-edge cuts isolate selected TC candidates as a controlled serviceability intervention, and these cuts count toward total disruption. The baseline blockage target is

$$B(t) = B_\infty(1 - e^{-t/\tau}),$$

with $B_\infty = 0.25$, $\tau = 110$ steps at M scale and $\tau = 130$ at L scale. A capped stochastic controller selects vulnerable edges toward this trajectory. Edge scores use topographic, infrastructure, rainfall, earthquake, and vulnerability proxies as configured; these scores are study-defined vulnerability indicators. Once blocked, an edge remains blocked because repair and reopening probabilities are zero. The robustness experiment changes the LB asymptote to 0.15 or 0.35 while leaving the other physical and algorithmic inputs fixed.

For an unblocked edge, the macro vulnerability score is $p_e^{\text{macro}} = \sigma(z_e)$, where $z_e = \beta_0 + \beta_s s_e + \beta_r r_e + \beta_q q_e + \beta_{sr} s_e r_e + \beta_{rq} r_e q_e + \beta_b b_e + \beta_i i_e + \beta_l l_e + \beta_{lr} l_e r_e + \beta_v v_e$. The terms are normalized slope, rain, earthquake, building, infrastructure, length, and base-vulnerability proxies. The step probability is $p_{e,t} = \text{clip}(p_e^{\text{macro}} g_t, 0, p_{\max})$, where the feedback gate g_t depends on the gap between the realized blocked fraction and $B(t)$; a hard count cap prevents exceeding the target trajectory. The configured coefficients and gate parameters are reported in Supplementary Table S6. This synthetic ranking-and-capping mechanism provides reproducible disruption realizations; its coefficients are not fitted to observed damage.

RB uses a fixed Dujiangyan-derived graph with 462 runtime nodes, 709 undirected operational edges, 113 candidate points of interest (POIs), and locations for 8 routine and 12 TC tasks. Its normalized earthquake-related spatial proxy excludes rainfall from the road-damage score. The R-scale factor reduces the base blockage asymptote from 0.25 to 0.1875; targeted isolation is disabled, and base wind and rainfall are zero. RB therefore tests execution on a larger, irregular road topology under spatially heterogeneous persistent disruption. The 20 task sites comprise 16 schools, three medical sites, and one park or open-space site selected from the facility-role candidates. These facility roles are consistent with potential shelter resources under Chinese emergency-shelter guidance [47]; the selected POIs are used as map-based task sites rather than as claims of official shelter designation. The road abstraction is map-derived, while demands, deadlines, and disruption are simulated.

Parameters are organized by decision role rather than treated as a single tuning set. Structural framework elements define the resolved/protected/eligible partition, exclusive ownership, and atomic route–contract installation. Physical and service inputs describe vehicle, payload, energy, weather, and task characteristics. Operational-policy settings control event admission, reserves, reassignment, and stalled-queue intervention; search-budget settings bound iterations, removals, and anchor candidates; and scenario-generator settings define synthetic road and disruption processes. Table 3 summarizes evidence status, while Supplementary Table S18 records each class, its source, tuning status, sensitivity evidence, and deployment interpretation. This organization separates structural choices from calibratable quantities: physical and policy inputs would require local calibration for deployment, whereas the structural execution-consistency rules are independent of their numerical values.

Table 3. Evidence status of the physical environment.

Element	Status in This Study	Permitted Interpretation
RB graph, POI file, fixed task file	prepared operational assets	single-topology stress-test input
FC100 and Canter published specifications	manufacturer scale anchors	bounds and platform context
payload/wind/rain relevance	peer-reviewed qualitative support [48–52]	sign and model structure
M/L geometry, vulnerability, blockage curve	model proxy	controlled synthetic stress
controlled-isolation cuts and deadlines	scenario assumption	serviceability stress, not observed damage
FC100 coefficients, reserve, weather gates	engineering assumption	common operational policy

2.2.2. Progressive Road Knowledge

Physical blockage and planner knowledge are different states. Let E_t be the physical road set and $\hat{E}_t^a$ the blocked-edge set known to agent a. Under shared awareness, planners learn failures through the same observation process. Truck and UAV scouts observe nearby edge segments; a blocked edge can also be learned when an attempted movement encounters it. A new observation is published to the shared map, increments a road version, invalidates affected shortest-path caches, and becomes available to the next planning decision. Thus a planned path may remain apparently feasible until the relevant failure is observed. This observation process progressively reveals the changing road state to the planners.

All algorithms use the same shared-road information and replanning-on-update interface; communication blackout is disabled. M and L suppress weather and stochastic road failure while retaining paired task metadata. MB and LB activate a spatially and temporally varying synthetic wind/rain field together with persistent blockage. RB uses the earthquake-related spatial disruption proxy with wind and rain inactive. Within each condition, the environment and information process are shared across methods.

Because blocked roads do not reopen, the shared map accumulates observed failures and may lag behind the physical graph. Shortest-path and feasibility queries use only the currently revealed decision graph. If a vehicle attempts to enter an unobserved failed edge, the environment rejects the movement and shares the observation. The resulting loop—physical failure, local observation, map and cache update, and replanning—generates the replanning events studied here.

2.2.3. Truck and UAV Operational Proxy

Trucks travel at a nominal 10 m s^{-1}, with execution speed adjusted for roads, slope, roughness, payload, and weather. The truck operating profile is anchored to a manufacturer-referenced vehicle class, and the UAV profile is referenced to the DJI FlyCart 100 (DJI, Shenzhen, China) in dual-battery lifting configuration. DJI specifies an 85 kg maximum dual-battery payload, transport of up to 65 kg over 12 km under the stated conditions, and idealized range and endurance values for windless sea-level operation [53–55]. The simulations use a 50 kg payload, 10 m s^{-1} route speed, 6000 m cumulative sortie limit, and 25% normalized-energy reserve as the common operating settings.

For one flight leg of length d, payload m, local headwind component w_h, and rainfall r, normalized energy is represented by

$$e(d, m, w_h, r) = c_0 d(1 + \alpha_w w_h + \alpha_r r)\left(1 + \alpha_m \min\left\{1, \frac{m}{50}\right\}\right),$$

where $c_0 = 1/(12000 \times 1.5) = 5.5556 \times 10^{-5}$ m^{-1}, $\alpha_m = 0.50$, $\alpha_w = 0.03$ s m^{-1}, and $\alpha_r = 0.01$ h mm^{-1}. For wind vector $\mathbf{w}$ and unit travel direction $\hat{\mathbf{d}}$, $w_h = \max\left(0, -\mathbf{w} \cdot \hat{\mathbf{d}}\right)$; tailwind therefore receives no energy credit. Execution speed additionally applies max(0.30, $1 - 0.03w_h$), max(0.35, $1 - 0.018r$) and max(0.25, $1 - 0.16m/50$) multipliers. The outbound leg uses current package mass; after service, the package is removed before recovery is evaluated. A stationary, undocked 20 s interval debits $1/36 = 0.02778$ of normalized battery, representing a 12 min hover budget; forced-return waiting scales this debit by 0.25, and docked UAVs recharge. These monotone engineering relationships define the common energy and speed model used in the comparisons.

Weather thresholds determine flight availability. New launches are blocked when wind exceeds 6 m s^{-1} or rainfall exceeds 24 mm h^{-1}; recovery uses thresholds of 10 m s^{-1} and 30 mm h^{-1}. The MB/LB base wind of 6 m s^{-1} equals the launch threshold, and base rainfall is 12 mm h^{-1}; local values vary in space and time. DJI's IP55 classification and take-

off/landing wind-resistance rating provide platform context [53,56], while the weather gates and rain–energy relationship remain operational model settings. The literature supports including payload, wind, and precipitation effects [48–52]. A one-at-a-time LB sensitivity analysis scales flight discharge per meter and base-wind severity by 0.8 and 1.2. Wind scaling modifies the coupled weather field and can therefore affect both road and UAV conditions.

The physical model provides a common energy, speed, and weather environment for comparing coordination policies. Local sensitivity analyses examine nearby input settings, while Section 4.3 discusses the additional calibration required for field transfer.

2.3. Formal Problem Definition, Execution State, and Outcome Measures

2.3.1. Observed State and Installed Plan

Figure 2 summarizes the operational truck–UAV service geometry, road-knowledge states, and scenario-level disruption proxies.

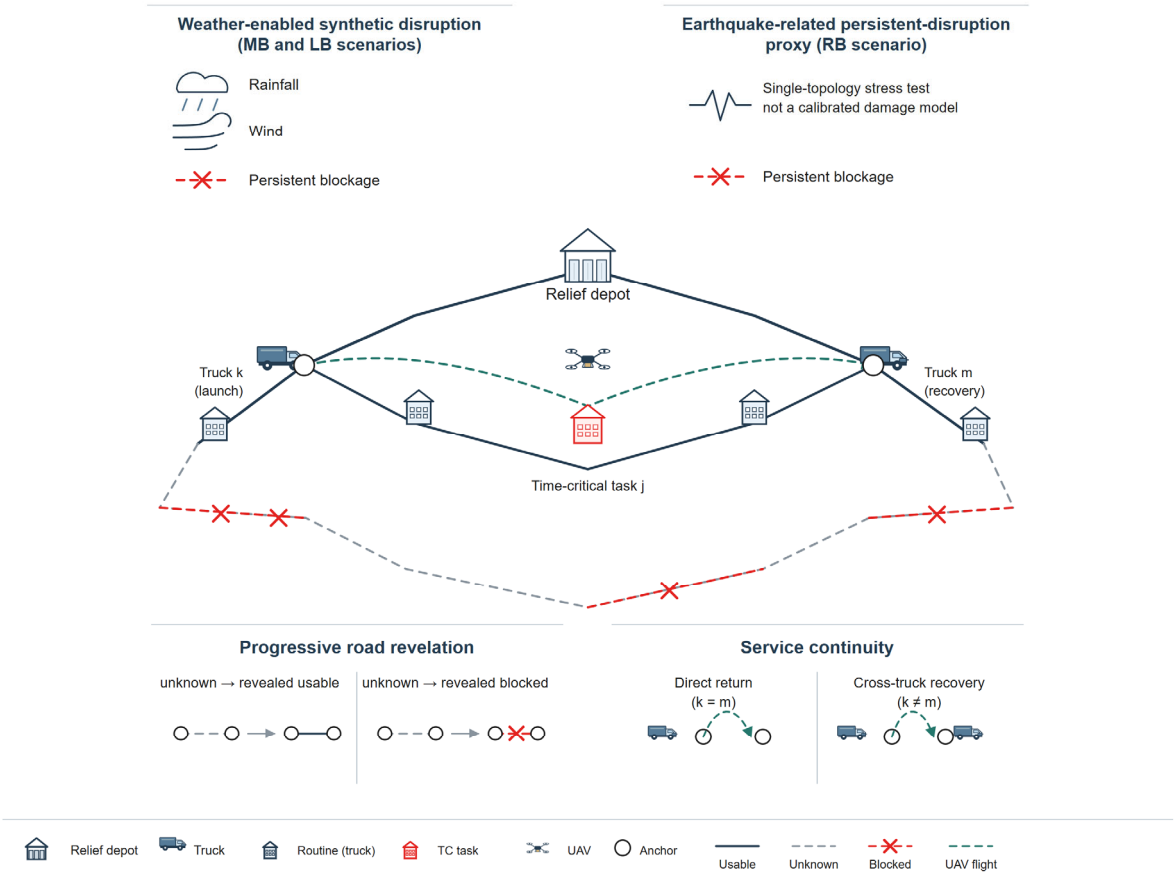


Figure 2. Truck–UAV relief network under progressively revealed disruption. Trucks follow revealed usable road edges to serve routine bulk tasks. An airborne sortie launches from Truck k, serves one TC task, and recovers to a compatible Truck m or returns to the launch carrier. A loaded UAV can also release a package while co-located with its carrier at a TC task site. The schematic distinguishes unknown, revealed usable, and revealed blocked roads. Synthetic weather-enabled scenarios include rainfall, wind, and persistent blockage; RB uses an earthquake-related spatial disruption proxy. All tasks are present initially, and blocked roads remain closed. Arrows in the lower panels indicate the direction of road-state revelation and UAV recovery movement.

At a planning epoch, the realized state is

$$S_t = \left(G_t, \hat{G}_t, X_t, A_t, H_t\right),$$

where G_t is the physical road graph, $\hat{G}_t$ is the shared revealed-road view, X_t contains vehicle position, motion, docking, battery, payload, and inventory, A_t contains task status and remaining lifeline, and H_t records execution-side commitments such as active sorties, recovery latches, current service, and route cursors. A planner returns only commands that can be enacted at the current step. Future stops remain in a route plan rather than being exposed simultaneously as independent low-level goals.

For plan version v, ER-HLNS represents the installed plan as

$$\Pi_t^v = \left(\{R_k^v\}_{k \in \mathcal{K}},\ \{C_i^v\}_{i \in A_t},\ \rho_t,\ J_t \right).$$

Here $R_k^v = \left(s_{k,1}, \ldots, s_{k,n_k}\right)$ is the remaining route sequence for truck cluster k, C_i^v is the exclusive contract for task i, ρ_t is the known blocked-edge signature, and J_t is the internal route rank. A routine stop sends a truck to the demand node. An aerial stop specifies a time-critical task, UAV, supporting truck, launch anchor, optional recovery truck and anchor, expected air/road distances, and a plan version. The contract also records the material source and reload requirements. Routes and contracts are installed atomically, preserving exclusive task ownership during a plan update.

Plan persistence follows realized execution. Each route cursor advances past delivered or otherwise terminal tasks. Claimed tasks, tasks in service, and tasks assigned to airborne UAVs are protected from ordinary removal. The next unresolved stop defines the execution boundary: eligible unfinished suffixes remain revisable, while completed or physically committed work is retained.

2.3.2. Protected and Revisable Decision Sets

At each admitted replanning epoch t, the current task set is partitioned according to realized execution status. Resolved tasks have already been delivered or otherwise terminated; execution-protected tasks are claimed, in service, assigned to an airborne UAV, or linked to an active recovery commitment; and eligible tasks are unfinished decisions that may still be revised. The three sets are mutually exclusive and collectively exhaustive for the active task ledger.

$$\mathcal{N} = \mathcal{N}_t^{\text{res}} \uplus \mathcal{N}_t^{\text{prot}} \uplus \mathcal{N}_t^{\text{elig}}.$$

Let Π_t^- denote the installed route–contract plan before an admitted update and let $\mathcal{F}(o_t)$ be the set of plans feasible under the currently observed state o_t. The replanning problem selects an updated plan only from $\mathcal{F}(o_t)$, minimizes the ordered planning rank, preserves installed decisions associated with execution-protected work, and excludes resolved work from future route and contract decisions.

$$\Pi_t^+ \in \arg\min\{J_{\text{lex}}(\Pi; o_t) : \Pi \in \mathcal{F}(o_t)\}, \quad \text{with} \quad \Pi_t^+ \mid \mathcal{N}_t^{\text{prot}} = \Pi_t^- \mid \mathcal{N}_t^{\text{prot}}.$$

Resolved tasks remain terminal in the task ledger and are excluded from future route and contract decisions. The feasible set enforces revealed-road reachability, exclusive task ownership, truck and UAV capacity, material availability, UAV energy and weather limits, launch/recovery compatibility, and agreement between route and contract versions. Only $\mathcal{N}_t^{\text{elig}}$ enters the repair neighborhood; a later road observation may change that eligible set and trigger another decision epoch.

2.3.3. Decision-Epoch Execution Consistency

Proposition 1 (decision-epoch execution consistency). *Assume that the realized state is internally consistent and that candidate route and contract updates pass the stated feasibility checks. At the installation epoch, an atomic ER-HLNS-PR update preserves exclusive task ownership,*

all execution-protected service and recovery commitments, material and capacity feasibility, and route–contract version consistency.

The result follows directly from the update construction: resolved work is removed from future routes, protected work is retained without ordinary reassignment, only eligible work enters repair, every candidate is checked against the revealed state, and accepted routes and contracts are installed as one version. The proposition establishes consistency at the current decision epoch; global optimality and robustness to future unrevealed disruptions are separate questions.

2.3.4. Outcome Measures

Let $z_{is} = 1$ if task i is completed in episode s, and 0 otherwise. Overall and class-specific completion are defined as the completion count divided by the number of tasks in the relevant set: $CR_s(A) = (\Sigma_i \in A\ z_{is})/|A|$, for $A \in \{N, N^R, N^c\}$.

To account for both delivered and unfinished work, priority-weighted censored completion time (PWCT) assigns each unfinished task the scenario horizon H_s:

$$\widetilde{C}_{is} = \begin{cases} \min(C_{is}, H_s), & z_{is} = 1, \\ H_s, & z_{is} = 0, \end{cases} \qquad PWCT_s = \frac{\sum_i u_i \widetilde{C}_{is}}{\sum_i u_i},$$

where u_i is task urgency. The horizons are 11,200 s for M/MB, 28,800 s for L/LB, and 48,000 s for RB. Lower PWCT is better. Mean completion time among completed tasks is also reported, but it is conditional on $z_{is} = 1$ and is interpreted alongside completion and PWCT. The secondary weighted-service score combines task urgency and remaining lifeline at realized service under the execution model; for task i, v_i is the fulfilled-demand fraction for a routine task, the remaining-lifeline fraction at service for a delivered TC task, or zero for an unfinished TC task, while u_i is urgency clipped to [0, 1]. The reported score is weighted_service_score = sum_i u_i v_i/sum_i max(u_i, 10^{-6}), with v_i clipped to [0, 1]. This study-specific diagnostic reports urgency-weighted service on a scale distinct from PWCT and is interpreted alongside the primary completion and PWCT outcomes. Makespan is the simulated time until termination or horizon, and runtime is wall-clock computation for one episode. Terminal UAV-loss events, feasibility diagnostics, and guard activations are reported separately.

The route manager uses the ordered criteria in Supplementary Table S16 as a lexicographic planning rank. Predicted TC misses and lateness have highest priority, followed by emergency-load balance and structural routine exposure. Under the parallel-rescue overlay, ties in these terms are resolved using low-slack emergency coverage, routine prefix burden, on-time routine coverage and speed, TC completion, and scalar travel cost. This internal rank guides plan construction; PWCT and the other evaluation metrics are calculated separately from realized task outcomes.

2.4. ER-HLNS-PR Hierarchical Coordination Framework

Figure 3 shows event admission, the boundary between protected and revisable work, deterministic route repair, and execution checks.

2.4.1. Active Planning Chain and Event Admission

ER-HLNS-PR couples an event-responsive rolling controller with a persistent hierarchical route manager. Each decision follows eight steps: classify the observed update; admit or suppress replanning; reconcile the installed plan with realized state; diagnose current stops and release eligible invalid contracts; reconstruct affected unfinished work;

improve the candidate through bounded deterministic search; install route and contract updates atomically; and issue current truck, UAV, launch, and recovery commands.

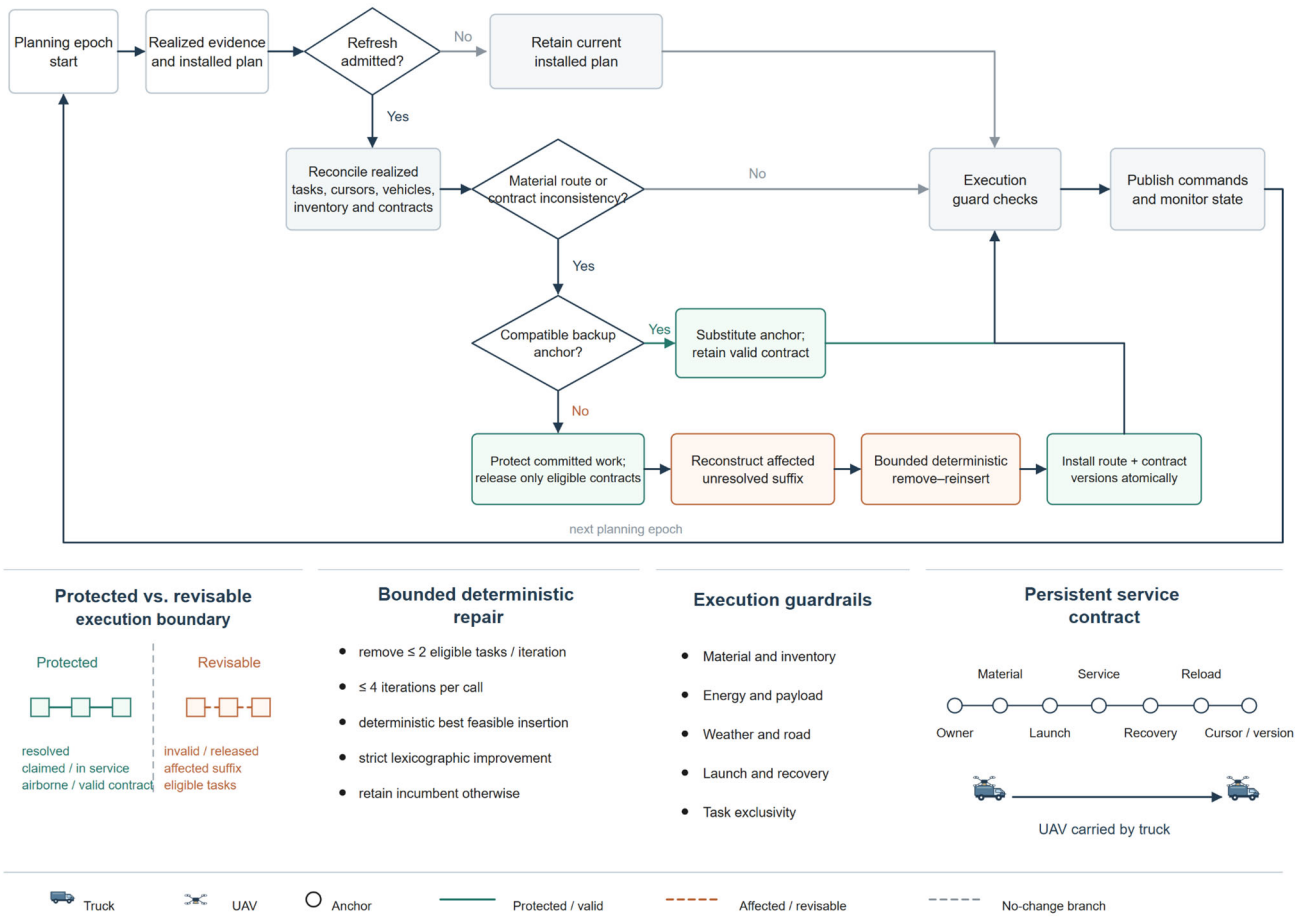


Figure 3. Execution-consistent event-responsive replanning. Suppressed updates retain the installed plan and proceed to execution checks; admitted updates trigger realized-state reconciliation. Feasible backup anchors and valid contracts are retained. Resolved work is advanced past the execution boundary; claimed, in-service, and airborne work remains protected, while eligible affected suffixes are repaired. Deterministic search removes at most two eligible tasks per iteration for at most four iterations, applies best feasible insertion, and accepts strict lexicographic improvements. Route and contract updates are installed atomically before material, road, energy, weather, and recovery checks determine the current commands.

High-priority replanning triggers include invalid or terminal task goals, unreachable current paths, UAV recovery needs, truck dead ends, uncovered high-priority tasks, and sustained non-progress. Lower-priority updates include routine task resolution, idle vehicles, and noncritical map changes. The depot-recovery sentinel is treated as a valid recovery identifier and excluded from missing-task checks. The controller uses a five-step decision interval, four-step active replanning cooldown, one low-value event replan per five-step window, five-step maximum fallback interval, and 12-step no-op event cooldown. Low-value events are merged or suppressed when they require no action, fall within cooldown, or exceed the budget; high-priority triggers bypass the low-value budget. Trigger categories record why replanning occurred, while feasibility and terminal-event outcomes are reported separately. This policy limits redundant searches while retaining an interval-based fallback.

After event admission, the route manager checks the affected routes. A changed road signature invalidates distance caches; repair is triggered when it also changes route or

contract feasibility. Examples include an unreachable stop, excessive detour, loss of the launch anchor and backups, an unavailable carrier or UAV, a route without an executable stop, or a contract that meets a release rule because of stalled progress, insufficient inventory, stock-out, or deadline risk. When a backup anchor restores feasibility, the assignment continues without reconstructing unrelated work.

2.4.2. Realized-State Reconciliation and Atomic Contracts

Before search, route cursors advance according to realized task status, and contracts for terminal tasks are removed. Compatible pending-task contracts are retained as prior assignments, with cooperative truck ownership preserved where feasible. Claimed, in-service, and airborne tasks enter the protected set. The remaining plan is reconstructed from current vehicle positions and inventories.

An aerial contract requires a compatible loaded UAV attached to the supporting truck, a road-reachable launch anchor, delivery and recovery legs within the 6000 m sortie limit, a predicted 25% energy reserve, an available recovery truck or depot, and valid weather conditions. Recovery may use a different truck when the launch and recovery trucks can reach their anchors and the combined aerial route is feasible. Launch and recovery support commands are issued together, and the UAV receives one contracted task. The execution layer rechecks launch eligibility at the realized anchor. Route repair is triggered when the detour ratio reaches 1.35 or when a high-priority feasibility condition occurs. Anchor search considers at most 320 candidates and retains three backups. Road-observation radii are 340 m for UAVs and 220 m for trucks. Deadline-rescue and emergency-deadline reserves are 6 and 20 steps, respectively; parallel-rescue ranking additionally uses the 30-step reserve below. The detour threshold permits a 35% increase over the current route estimate, while the candidate cap and backup count bound anchor-search effort.

Anchor selection starts from road nodes within the task's aerial-service radius. Candidates are ranked by air distance to the task and road distance from the supporting truck, and a bounded subset is retained. Direct return uses the launch truck as the recovery carrier. A cross-truck alternative is admitted only when both trucks can reach their anchors and the loaded outbound and empty recovery legs satisfy the joint distance and energy limits. Stored backup anchors allow the controller to respond to newly revealed cuts while retaining the task assignment where feasible.

The execution layer checks each proposed action against realized physical conditions. Invalid commands are rejected or replaced by an admissible hold or recovery response, and the resulting state informs the next decision. Task completion is recorded only after the execution layer confirms service and material transfer.

2.4.3. Parallel Rescue and Mixed Routine/Time-Critical Coverage

ER-HLNS without the parallel-rescue overlay contains the hierarchical route manager and event-responsive repair mechanism. ER-HLNS-PR adds a fixed coordination overlay that addresses stalled TC queues while retaining routine truck service. The overlay combines mixed coverage, residual routine matching, controlled reassignment, and stalled-queue anchor rescue. Its thresholds are algorithm settings, distinct from physical-environment parameters.

Mixed coverage retains predicted TC misses and lateness as the highest lexicographic priorities, applies a 30-step emergency reserve, and favors feasible routine stops when emergency terms tie. A routine stop can precede aerial service when it preserves the reserved emergency slack. The joint-corridor test allows a truck to continue toward a routine task while its UAV serves a TC task, provided that launch, delivery, recovery, truck progress, and delay limits remain jointly feasible. When recovery is possible near the truck's

ongoing corridor, the truck continues moving; otherwise, the plan assigns a move-or-hold launch anchor or a separate recovery-truck anchor. Parallel rescue therefore coordinates truck progress and UAV service within shared feasibility limits.

When all active time-critical work is terminal, residual routine tasks are matched to available stocked trucks by a maximum-cardinality bipartite matching procedure. Tasks with the fewest feasible trucks are considered first, while prior ownership is a soft preference. An augmenting path can reassign a flexible truck so that a structurally constrained routine task is not stranded. During mixed operation, a nearby pending routine task may be transferred once to another truck only when its revealed-road estimated arrival improves by at least 3 steps and 20%, the task lies within 800 m of the candidate truck, inventory is sufficient, and no incompatible UAV support or recovery commitment is broken. The three-step and 20% improvement requirements limit marginal ownership transfers; 800 m restricts the local candidate set. Their nearby alternatives are assessed in the threshold sensitivity.

Stalled-queue rescue is activated after 18 steps without completion progress or when a queue becomes explicitly urgent. It advertises at most one rescue task per UAV and no more than three simultaneous rescues. A task that passes the launch checks can be served from the current carrier location. Otherwise, the truck moves to the closest road-reachable launch anchor, the UAV remains bound during the approach, and launch feasibility is rechecked on arrival. A four-step launch handshake and an 18-step anchor timeout release stale rescues; a newly discovered cut can trigger bounded re-anchoring without changing the task–UAV contract. The routine-service gate prevents creation of the first nonurgent rescue before any routine task has entered service. These rules target queue stagnation while constraining pre-emption.

2.4.4. Bounded Deterministic Remove–Reinsert Search

Initial construction orders active tasks deterministically and inserts each at its best feasible truck and route position. For a candidate route set R, define

$$\Phi(R) = (M_C, L_C, B_C, D_R, L_R, Q_C, S_C, -G_C, P_R, -F_R, -V_R, U_R, T_C, c_R),$$

where $M_C = \sum_{i \in \mathcal{N}^C} 1[\eta_i > d_i]$ and $L_C = \sum_{i \in \mathcal{N}^C} \max(\eta_i - d_i,\ 0)$. B_C is the sum of squared excess emergency assignments over configured cluster targets (or, when that target mode is inactive, the squared excess of the maximum over the minimum cluster count). D_R and L_R count and sum lateness for routine tasks protected by the revealed-disconnection profile. With reserve $h_C = 30$ steps, $Q_C = \sum_i 1[d_i - \eta_i < h_C]$, $S_C = \sum_i \max(h_C - \max(d_i - \eta_i,\ 0),\ 0)$, and $G_C = \sum_i \max(d_i - \eta_i,\ 0)$. P_R counts emergency stops preceding routine stops when the balanced-all-tasks capability is active. $F_R = \sum_{i \in \mathcal{N}^R_{\text{on-time}}} (1 + u_i)/\max(\eta_i - t,\ 1)$ and $V_R = \sum_{i \in \mathcal{N}^R_{\text{on-time}}} (1 + u_i)$. U_R is the number of active routine tasks absent from all routes plus scheduled routine tasks predicted late. $T_C = \sum_{i \in \mathcal{N}^C} (1 + u_i)\ \max(\eta_i - t,\ 0)$. Finally, c_R is the sum of route-stop costs plus $0.75(\max_k |R_k| - \min_k |R_k|)$. Components whose capability is inactive evaluate to zero. The 14 components are compared lexicographically, so a lower-order distance improvement cannot offset a predicted time-critical miss.

A feasible route predicting one TC miss ranks ahead of a shorter route predicting two misses; travel cost breaks ties only after all higher-priority criteria agree. Feasibility checks precede lexicographic comparison.

The search uses a finite, deterministic remove–reinsert neighborhood. At each of four iterations, at most two unprotected tasks are removed in a stable cyclic order and reinserted in deterministic priority order at their best feasible positions. Only a strict lexicographic improvement replaces the incumbent. Although ER-HLNS is retained as the method identifier, the implemented inner search is a bounded deterministic remove–reinsert procedure;

it uses neither adaptive operator weights nor simulated-annealing acceptance. The four-iteration limit was fixed before the formal comparisons as a prespecified computational budget. Supplementary Table S17 shows no monotonic service improvement as the budget increases, whereas upper-tail replanning latency and objective evaluations rise substantially. The four-iteration setting is therefore treated as a bounded computation choice rather than a performance-tuned optimum. Algorithm 1 summarizes the complete rolling decision and unresolved-suffix repair procedure.

Algorithm 1. ER-HLNS-PR rolling decision and unresolved-suffix repair.

Line	Operation
Input	realized state S_t, installed plan Π^v, iteration bound $I = 4$
Output	executable commands g_t and updated plan $\Pi^{v'}$
1–4	Classify refresh evidence; suppress weak nonactionable, cooling-down, or over-budget refreshes; otherwise admit a refresh.
5–8	Advance route cursors from realized status; diagnose reachability, detour, vehicle and anchor state; substitute a compatible reachable backup anchor.
9–11	Identify contracts eligible for release or transfer; if no route/contract inconsistency remains, publish current stops.
12–14	Retain compatible contracts, reconstruct remaining route lines by best insertion, and protect claimed, in-service, and airborne work.
15–18	For $q = 0, \ldots, I - 1$, remove at most two unprotected tasks, reinsert them at best feasible positions, and accept only strict lexicographic improvement.
19–23	Install route and contract versions atomically; append feasible uncovered fallback suffixes; apply mixed-coverage, residual-routine, and stalled-queue rules; publish current commands and anchors.

Let n be the number of active tasks, m the number of truck clusters, $I = 4$, and $r \leq 2$. One candidate evaluation scans O(n) route stops and makes shortest-path/anchor queries. With cached routes, a query is O(1) lookup; a cache miss uses Dijkstra in O((V + E) log V), while an anchor scan is linear in its bounded candidate set. Thus Q = O(n + (V + E) log V + a) in the conservative cache-miss case, where a is the anchor-candidate count. Best insertion examines O(mn^2) task–position alternatives, for O(mn^2Q) time; improvement examines O(Irmn) reinsertion alternatives, for O(IrmnQ). Storage is O(n + m + V + E + C), including the graph and a shortest-path cache of size C. The finite candidate set, fixed iteration bound, bounded removal, and strict acceptance guarantee finite termination of each call; the procedure is used as a heuristic search.

Execution consistency here means that current commands and retained contracts agree with realized task ownership, protected work, route-cursor state, and execution-layer feasibility checks at the decision epoch. New disruptions may invalidate future route segments, at which point the same consistency checks are applied during the next update.

2.5. Comparators, Ablations, and Capability Isolation

The five-scenario comparison includes the configured planners below; the principal hard-LB cohort additionally includes a full-route rolling-horizon planner under the same physical environment. PG assigns each available agent the lexicographically smallest eligible-task key $(I_R, -u_i, d_i - t, \delta_{ai}, \mathrm{id}_i)$, where $I_R = 0$ for TC and 1 for routine work, u_i is urgency, and δ_{ai} is current agent–task distance. Greedy selects the nearest eligible task and ignores class, urgency and deadline ranking. C-ALNS uses a fixed eight-step decision interval in LB, four iterations per call, two-task sequences, canonical random/worst/related/segment removal, greedy/regret insertion, adaptive operator weights (learning rate 0.25, 12-iteration segment, minimum weight 0.1), and simulated-annealing

acceptance (initial temperature 0.05, cooling 0.985, minimum 10^{-4}); event triggering is disabled. DR-ALNS uses the same four-iteration canonical search and a five-step decision interval, but additionally replans when the pending signature changes, a goal becomes infeasible, road feasibility changes, or its interval fallback expires. ER-HLNS without the parallel-rescue overlay contains event admission, persistent route sequences, contracts, and the four-iteration deterministic remove–reinsert search. ER-HLNS-PR is the complete coordination framework.

The comparators use distinct planning capabilities within a shared execution environment. C-ALNS and DR-ALNS omit persistent cooperative contracts, joint-corridor coordination, anchor rescue, and the parallel-rescue overlay; ER-HLNS-PR uses its deterministic route manager instead of canonical ALNS acceptance. C-ALNS-MP separates mission-priority effects from these architectural differences. It retains fixed-period canonical operators, four iterations, simulated-annealing acceptance, and the native two-task representation, but ranks candidates by hard violations, unserved TC tasks, lifeline loss, unserved routine tasks, and residual travel, energy, switching, and support costs. Thus, the leading mission priorities are matched while candidate representations remain method-specific. In the computational control, each instance's ER-HLNS-PR objective-evaluation count becomes the ceiling for C-ALNS-MP, whose loop stops at that ceiling or its calibrated iteration bound. Target and realized evaluation counts and wall-clock runtime are reported separately. All methods share the physical environment, tasks, initial fleet, exogenous randomness, information interface, low-level execution rules, and feasibility checks within each condition–seed block. Table 4 summarizes their planning capabilities.

Table 4. Capability boundary used for comparison.

Method	Mission-Priority Rank	Interval-Only Replanning	Dynamic Trigger	Persistent Contracts	Bounded Route-v2 Search	Parallel-Rescue Overlay
PG	yes	no	no	no	no	no
Greedy	no; nearest eligible	no	no	no	no	no
C-ALNS	no	yes	interval only	no	no	no
C-ALNS-MP	matched leading rank	yes	interval only	no	no	no
DR-ALNS	no	no	road/task/goal + interval	no	no	no
DR-ALNS-K2	no	no	event + interval	no	no	no
RH-ALNS-FR	internal full-route	no	admitted + interval	no	no; full route	no
FULL-GLOBAL	this study	no	event admission	yes; global	yes	yes
ER-HLNS (no PR overlay)	this study	no	event admission	yes	yes	no
ER-HLNS-PR	this study	no	event admission	yes	yes	yes

RH-ALNS-FR provides a full-route rolling-horizon benchmark implemented for the shared environment. At each admitted planning opportunity, it can revise the unresolved route set under the same task, fleet, road-information, and execution rules. Its settings, objective evaluations, feasibility outcomes, and wall-clock cost are logged alongside ER-HLNS-PR. This is a full-system comparison; the E4 contrast between suffix-local repair and global re-auction isolates repair scope within ER-HLNS-PR.

Dong et al. [28] and Zhang and Li [29] study related flood-routing problems with different service and environmental representations, while Long et al. [57] propose integrated truck–UAV tabu search for urban emergency response. Supplementary Table S1a,b extend this comparison to recent dynamic, stochastic, robust, and belief-based methods [31–37]. The studies differ in solver type as well as task release, truck/UAV service eligibility, capacity representation, launch and recovery coupling, road-information dynamics, and treatment of service and recovery commitments already under execution. These differences alter the feasible decision space. Accordingly, quantitative comparisons are restricted to implementations validated under the same simulation, information, and execution rules, while recent published methods with different formulations are compared structurally in Supplementary Table S1a,b.

Four LB ablations examine the coordination configuration: A1 removes parallel rescue, A2 disables event triggering, A3 disables affected-suffix event repair, and A4 disables joint-corridor coordination. Each contrast measures the effect of its configuration change with the remaining settings retained; interactions prevent interpreting the contrasts as additive component contributions. The truck-only (TO) control removes UAV service while retaining truck eligibility rules. Road-severity tests compare C-ALNS, DR-ALNS, and ER-HLNS-PR at LB blockage asymptotes of 0.15 and 0.35.

Additional experiments address specific comparison questions. E1 examines mission-priority matching and targeted isolation, E5 controls the objective-evaluation budget, and E6 applies shared conditional truck eligibility for TC tasks. E4 compares suffix-local repair with global re-auction of all unresolved pending contracts while retaining the remaining ER-HLNS-PR settings. E2 varies energy and wind inputs, E3 varies policy thresholds, and E7 varies TC lifeline decay. The Supplementary Information reports their settings and paired results.

2.6. Experimental Protocol and Statistical Analysis

The principal hard-LB comparison uses 20 independently generated task–road instances (seeds 100–119), with three algorithm repetitions per method and instance. Methods share tasks, road conditions, initial fleets, exogenous realizations, and execution rules within each instance. The five-scenario display and targeted controls use their documented ten-seed sets. Episodes provide measurements; instance-level averages over algorithm repetitions form the units of paired inference. All planned hard-LB cells were retained under the same physical and software configuration.

Within each paired block, methods share topology, tasks, initial agents, horizon, physical configuration, and information access. M–MB and L–LB also share the seed-specific network and tasks across open-road and disruption conditions. RB uses a fixed topology and the same seed-specific disruption realization across methods. Integrity checks found no missing or duplicate episode keys. Runtime is measured per episode and reported descriptively under the hardware and workload conditions specified below.

Primary service outcomes are overall, routine, and TC completion, with PWCT reporting urgency-weighted timing while retaining unfinished work. The study-defined weighted-service score is a secondary diagnostic. Other secondary measures include completion time among delivered tasks, TC on-time completion, makespan, wall-clock runtime, failed-task counts, terminal UAV-loss events, feasibility diagnostics, and guard activations. Aggregate task counts describe service volume rather than inferential sample size. Class-specific completion and delivered mass distinguish 50 kg TC packages from 800 kg routine deliveries.

Paired identity was verified from the condition, seed, task, road, and environment records before analysis; detailed row-count and audit checks are reported in the Supplementary Information and Source Data S1.

For the principal hard-LB analysis, three algorithm repetitions are averaged within each of 20 independent instances before paired method differences are computed. Marginal 95% percentile-bootstrap intervals use 10,000 resamples of these instance-level differences. Ten-instance controls are analyzed separately using their own paired vectors. The intervals are exploratory and are not adjusted for simultaneous inference or multiple comparisons; an interval containing zero is treated as uncertainty about the difference, not evidence of equivalence. Task rows and algorithm repetitions are not counted as independent replicates. The supplementary eight-contrast analysis also reports the signed-rank effect-size diagnostic below.

$$r_{rb} = \frac{W^+ - W^-}{W^+ + W^-}.$$

Here W^+ and W^- denote the favorable and unfavorable signed-rank sums, respectively.

Wins, ties, and losses follow the favorable direction: higher completion and lower PWCT or runtime. Relationships between logged mechanism measures and service outcomes are summarized descriptively.

For the five-method set (PG, Greedy, C-ALNS, DR-ALNS, and ER-HLNS-PR), tie-corrected Friedman statistics and Kendall's W summarize rankings separately by scenario for overall completion and PWCT. ER-HLNS without the parallel-rescue overlay is shown as an ablation and excluded from this ranking analysis. With ten blocks, the chi-square approximation is exploratory. LB ablations, blockage sensitivity at 0.15 and 0.35, RB single-topology contrasts, and the TO comparison use paired mean differences and seed-block percentile-bootstrap intervals as secondary analyses.

PWCT is reconstructed from the task ledger using the exact scenario horizon for unfinished work. Descriptive tables report means and standard deviations; paired analyses report mean differences. Each table identifies whether it uses 20 hard-LB instances or ten seed blocks and reports exploratory sign-flip p-values and wins, ties, and losses where applicable. Terminal events, guard activations, and replayed task/route integrity checks are tabulated separately, and overlapping guard reasons are not summed as unique events. Because methods share paired seeds, zero observed audit violations are reported as counts without an independent-binomial safety bound.

Analyses use deterministic Python scripts; hardware and software versions are listed in Section 2.7.

Mechanism summaries use ten targeted-isolation ER-HLNS-PR episodes. Their logs record refresh admission, repair time, affected-suffix size, route edits, prior contract signatures, task-owner changes, and execution rebindings. Supplementary Tables S5 and S12 define these fields and the corresponding audit scope.

2.7. Software and Reproducibility

Experiments ran on a Windows workstation with an Intel Core i9-13900HX CPU (Intel Corporation, Santa Clara, CA, USA; 24 cores, 32 logical processors), 16 GB RAM, Python 3.9.23, NumPy 2.0.2, pandas 2.3.2, SciPy 1.13.1, NetworkX 3.2.1, and PyTorch 2.4.1+cu121. The accompanying reproducibility archive contains experiment and analysis scripts, seed and repetition protocols, processed Source Data S1, expected cell counts, executable configurations, and RB road, POI, and task assets. File manifests record SHA-256 hashes; the data notice provides OpenStreetMap/ODbL attribution, and the README gives reproduction commands. OpenAI Codex (model: GPT-6 Sol; OpenAI, San Francisco,

CA, USA; https://openai.com/codex/; accessed on 12 August 2026) assisted with code inspection and checks against archived analysis outputs.

3. Results

3.1. Main Service Pattern Across Scenarios

Figure 4 and Tables 5 and 6 summarize service outcomes across five scenarios. All planned method–scenario–seed cells were retained, with no missing or duplicate episode records. These descriptive summaries use ten seed realizations per scenario; the expanded 20-instance hard-LB comparison is reported separately in Section 3.2.

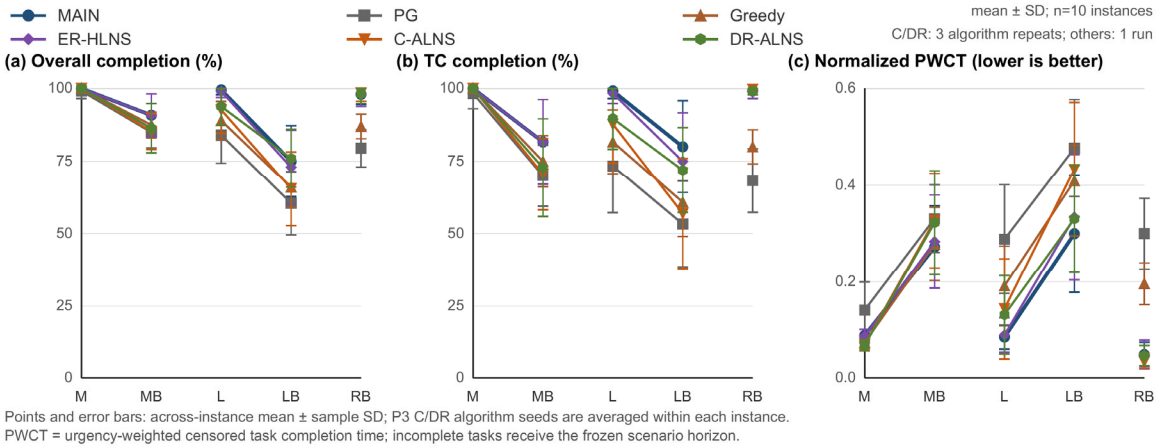


Figure 4. Service outcomes across five scenarios: (**a**) overall completion, (**b**) TC completion, and (**c**) PWCT normalized by the scenario horizon. Points are means over ten seed-level observations (m), with algorithm repetitions averaged within seed where applicable; error bars show standard deviations. Higher completion and lower normalized PWCT are favorable. Connectors link the paired M–MB and L–LB conditions; RB is shown separately. Scenario labels are categorical. Source Data S1: instance_level_summary.csv.

Table 5. Descriptive outcomes pooled over five scenario families and ten seeds per family, giving 50 scenario–seed observations per method. Completed-task counts use 840 tasks per method as the pooled denominator. For methods with repeated algorithm seeds, counts are averaged within instance and may be fractional. Pooled observations are descriptive, and weighted service is the study-defined secondary score. Source Data S1: SourceData_Table5_pooled_complete_r2.csv.

Method	Episode Completion	Completed Tasks	Routine Completion	TC Completion	PWCT (s)	Weighted Score	Runtime (s)
PG	0.8163	669/840	0.9350	0.7267	8349.6	0.6427	35.67
Greedy	0.8590	709/840	0.9425	0.7950	6132.5	0.7153	33.52
C-ALNS	0.8839	735.7/840	0.9500	0.8300	4564.8	0.7264	67.63
DR-ALNS	0.9088	759.7/840	0.9575	0.8672	3999.6	0.7299	67.32
ER-HLNS (no PR overlay)	0.9217	769/840	0.9325	0.9083	3747.3	0.7700	34.84
ER-HLNS-PR	0.9267	774/840	0.9275	0.9200	3512.0	0.7698	33.94

The five-scenario display characterizes performance across open-road, synthetic disruption, and fixed-map conditions. The expanded hard-LB cohort below provides the principal comparison with mission-priority-matched and full-route baselines.

Table 6. Scenario-specific mean outcomes for ER-HLNS-PR, C-ALNS, and DR-ALNS over ten seed realizations per scenario. Algorithm repetitions are averaged within seed where applicable; paired uncertainty is reported separately. Source Data S1: Table6_MAIN_C_DR_by_scenario.csv.

Scenario	Method	Overall Completion	Routine Completion	TC Completion	PWCT (s)
M	ER-HLNS-PR	1.0000	1.0000	1.0000	1000.7
M	C-ALNS	1.0000	1.0000	1.0000	795.0
M	DR-ALNS	1.0000	1.0000	1.0000	776.0
MB	ER-HLNS-PR	0.9083	1.0000	0.8167	3048.9
MB	C-ALNS	0.8528	1.0000	0.7056	3666.3
MB	DR-ALNS	0.8639	1.0000	0.7278	3613.5
L	ER-HLNS-PR	0.9950	1.0000	0.9917	2473.7
L	C-ALNS	0.9267	1.0000	0.8778	4158.5
L	DR-ALNS	0.9383	1.0000	0.8972	3797.2
LB	ER-HLNS-PR	0.7500	0.6750	0.8000	8635.3
LB	C-ALNS	0.6550	0.7833	0.5694	12,478.4
LB	DR-ALNS	0.7600	0.8208	0.7194	9521.5
RB	ER-HLNS-PR	0.9800	0.9625	0.9917	2401.4
RB	C-ALNS	0.9850	0.9667	0.9972	1725.7
RB	DR-ALNS	0.9817	0.9667	0.9917	2289.8

Scenario responses differed (Table 6). All three tabulated methods completed every task in the open-road M condition. ER-HLNS-PR had higher descriptive TC completion and lower PWCT than C-ALNS and DR-ALNS in MB and L. In LB, higher TC completion was accompanied by lower routine completion, while the RB completion rates were closely grouped. The expanded hard-LB analysis examines these task-class differences with matched mission-priority and full-route baselines implemented under the shared execution rules, together with paired intervals.

The MB/LB paired contrasts provide supporting scenario context, while the 20-instance hard-LB cohort is the primary basis for the comparison with the matched mission-priority and full-route baselines.

3.2. Hard-LB Comparisons with Matched and Full-Route Baselines

Figure 5 and Tables 7 and 8 report the principal comparison over 20 independent hard-LB instances with three repetitions per method. ER-HLNS-PR completed 73.8% of TC tasks versus 57.2% for C-ALNS-MP, a paired increase of 16.5 percentage points (descriptive 95% interval: 8.2–25.0). Routine completion was lower, at 68.1% versus 79.6% (difference: −11.5 points; interval: −18.8 to −4.4). Mean PWCT decreased from 12,402.8 to 9958.6 s (difference: −2444.2 s; interval: −4295.7 to −587.4). The weighted-service score increased by 0.083 (0.042–0.124), and whole-episode runtime decreased by 24.7 s (−40.0 to −7.3). Relative to RH-ALNS-FR, the differences were +0.015 (−0.021 to +0.056) for weighted service, −38.8 s (−1560.8 to +1540.5) for PWCT, and −3.0 s (−14.3 to +6.3) for runtime; all three intervals included zero. FULL-GLOBAL had a higher mean weighted-service score, 0.501 versus 0.483 (ER-HLNS-PR minus FULL-GLOBAL: −0.018; interval: −0.036 to −0.001).

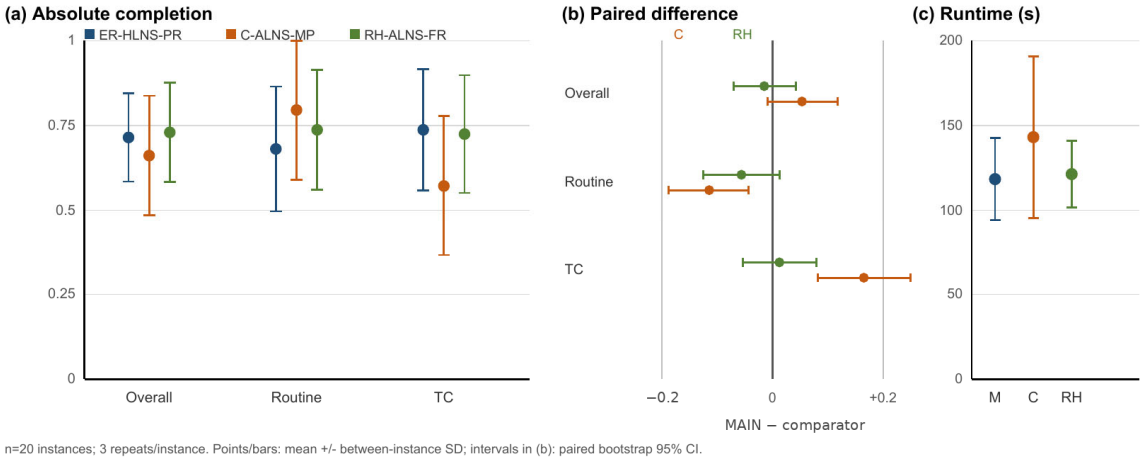


Figure 5. Principal hard-LB comparison over 20 independent task–road instances. (**a**) Completion rates for ER-HLNS-PR, C-ALNS-MP, and RH-ALNS-FR, with standard deviations across instance-level means; (**b**) paired completion differences with marginal 95% percentile-bootstrap intervals; and (**c**) whole-episode runtime, with standard deviations. Three algorithm repetitions are averaged within each instance. Intervals in (**b**) use 10,000 resamples of the 20 paired instance differences. Source Data S1: Figure5_G1_data.csv.

Table 7. Absolute outcomes in the principal hard-LB comparison. Entries are means ± standard deviations over 20 independent task–road instance means, each averaged across three algorithm repetitions. PWCT is measured in seconds; lower values are favorable. Source Data S1: Table7_G1_absolute_summary.csv and G1_PWCT_method_summary.csv.

Method	Overall	Routine	TC	Weighted Service	Runtime (s)	PWCT (s)
ER-HLNS-PR	0.715 ± 0.130	0.681 ± 0.184	0.738 ± 0.178	0.483 ± 0.114	118.5 ± 24.1	9958.6 ± 3814.0
C-ALNS-MP	0.662 ± 0.176	0.796 ± 0.205	0.572 ± 0.206	0.400 ± 0.140	143.2 ± 47.6	12,402.8 ± 4880.1
RH-ALNS-FR	0.730 ± 0.146	0.738 ± 0.176	0.725 ± 0.173	0.468 ± 0.099	121.4 ± 19.5	9997.4 ± 3884.0
DR-ALNS-K2	0.711 ± 0.156	0.792 ± 0.201	0.657 ± 0.182	0.420 ± 0.124	124.5 ± 31.1	10,852.8 ± 4215.0
FULL-GLOBAL	0.733 ± 0.144	0.681 ± 0.188	0.767 ± 0.161	0.501 ± 0.106	116.4 ± 24.7	9275.6 ± 3775.0

The same task-class pattern appeared in the separate ten-instance control without targeted isolation: ER-HLNS-PR showed higher TC completion and lower routine completion than C-ALNS-MP. This consistency supports interpreting the hard-LB result as a service-allocation trade-off rather than an artifact of targeted isolation.

The TC completion difference relative to C-ALNS-MP persisted after matching leading mission priorities. Separate controls address targeted isolation, objective-evaluation budget, and conditional truck service. Table S15 reports mean within-instance standard deviations across three algorithm repetitions, whereas Table 7 reports variation among instance means. C-ALNS-MP and DR-ALNS-K2 vary with algorithm seed; deterministic ER-HLNS-PR and FULL-GLOBAL give identical outcomes across the three algorithm seeds within a fixed instance. Candidate representations and coordination capabilities differ, but all methods use the same low-level execution rules. For weighted service relative to RH-ALNS-FR, all 20 leave-one-instance-out point estimates remained positive (+0.0033 to +0.0222), although the full-cohort paired interval included zero.

Table 8. Paired differences for ER-HLNS-PR minus C-ALNS-MP and RH-ALNS-FR in the principal hard-LB comparison. Marginal 95% percentile-bootstrap intervals use 10,000 resamples of the 20 paired instance differences. Positive completion and weighted-service differences, and negative PWCT and runtime differences, favor ER-HLNS-PR. Source Data S1: Table8_G1_paired_differences.csv and G1_PWCT_paired_differences.csv.

Contrast	Outcome	Mean Difference	95% CI Low	95% CI High	n
MAIN − C-ALNS-MP	Overall completion	+0.053	−0.008	0.118	20
MAIN − C-ALNS-MP	Routine completion	−0.115	−0.188	−0.044	20
MAIN − C-ALNS-MP	TC completion	+0.165	0.082	0.250	20
MAIN − C-ALNS-MP	Weighted service score	+0.083	0.042	0.124	20
MAIN − C-ALNS-MP	Episode runtime	−24.7	−40.0	−7.3	20
MAIN − C-ALNS-MP	PWCT (s)	−2444.2	−4295.7	−587.4	20
MAIN − RH-ALNS-FR	Overall completion	−0.015	−0.070	0.043	20
MAIN − RH-ALNS-FR	Routine completion	−0.056	−0.125	0.013	20
MAIN − RH-ALNS-FR	TC completion	+0.013	−0.054	0.079	20
MAIN − RH-ALNS-FR	Weighted service score	+0.015	−0.021	0.056	20
MAIN − RH-ALNS-FR	PWCT (s)	−38.8	−1560.8	+1540.5	20
MAIN − RH-ALNS-FR	Episode runtime	−3.0	−14.3	6.3	20

3.3. Repair Locality and Plan Stability

RQ3 examines how selectively the framework updates an ongoing plan. In ten logged LB episodes, it admitted 1240 of 10,812 event opportunities (11.47% pooled; 11.62% as the mean of seed-specific rates). Each admitted repair affected an average of 3.29% of the unresolved route and retained 99.62% of previously recorded contract signatures. Across these episodes, the logs recorded 703 route-position edits, 92 task-owner changes, and 92 execution rebindings. Figure 6a,b summarize the event and repair patterns.

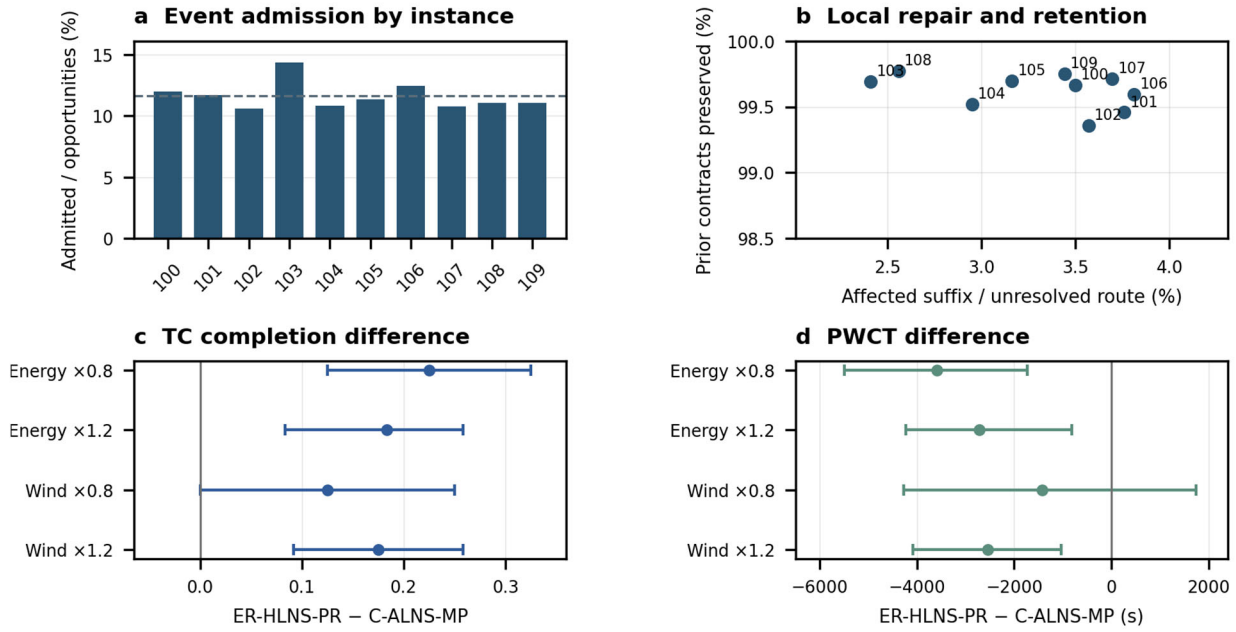


Figure 6. Event-response behavior and physical-input sensitivity. (**a**) Seed-specific refresh-admission rates after excluding the depot-recovery sentinel from missing-task triggers; the dashed line is the unweighted ten-seed mean. (**b**) Affected-suffix fractions and retention of previously recorded

contract signatures, with one point per episode. (**c**,**d**) Paired ER-HLNS-PR minus C-ALNS-MP differences in TC completion and PWCT after scaling energy-discharge or base-wind inputs by 0.8 or 1.2. Error bars are marginal 95% percentile-bootstrap intervals from 10,000 resamples of ten paired seeds. Wind scaling affects the coupled weather input and can influence both road and UAV conditions. Source Data S1: SourceData_Figure6_mechanism_same_version.csv and SourceData_Table9_E2_same_version.csv.

The 20-instance repair-scope control directly compares suffix-local repair with global pending-task re-auction. Local repair reduced task-owner changes by 8.4 on average (descriptive 95% interval: −12.2 to −5.2 for local minus global) and execution rebindings by 8.5 (−12.2 to −5.2). The corresponding differences were −0.017 (−0.047 to +0.010) for overall completion and +682.9 s (−247.4 to +1706.5) for PWCT. Because both service intervals spanned zero, the clearest observed effect of locality was lower plan disruption.

The ten-instance ablations examine the parallel-rescue and joint-corridor configurations. Relative to A1, which omits parallel rescue, ER-HLNS-PR increased TC completion by 0.050 (descriptive 95% interval: +0.017 to +0.083) and overall completion by 0.020 (−0.010 to +0.050). Relative to A4, which omits joint-corridor coordination, the differences were +0.025 (−0.033 to +0.092) and +0.045 (+0.000 to +0.090), respectively. These contrasts describe changes to the complete coordination configuration; the components may interact.

3.4. Sensitivity, Scalability, and Execution Audits

The sensitivity analyses examine changes in energy, wind, policy thresholds, lifeline decay, and blockage severity (Figure 6c,d and Table 9). Single-factor and coupled-input tests characterize the response around the chosen operating settings.

Table 9. Physical-input sensitivity: paired ER-HLNS-PR minus C-ALNS-MP differences in overall, routine, and TC completion and PWCT. E and W denote energy-discharge and base-wind scaling, respectively. Brackets show marginal 95% percentile-bootstrap intervals from 10,000 resamples of ten paired seeds. Source Data S1: SourceData_Table9_E2_same_version.csv.

Condition	Overall Completion	Routine Completion	TC Completion	PWCT (s)
E × 0.8	0.080 [0.020, 0.140]	−0.138 [−0.225, −0.050]	0.225 [0.125, 0.325]	−3595 [−5496, −1728]
E × 1.2	0.060 [0.000, 0.110]	−0.125 [−0.212, −0.050]	0.183 [0.083, 0.258]	−2712 [−4234, −814]
W × 0.8	0.020 [−0.095, 0.120]	−0.138 [−0.300, −0.013]	0.125 [0.000, 0.250]	−1429 [−4273, 1742]
W × 1.2	0.055 [0.000, 0.105]	−0.125 [−0.225, −0.050]	0.175 [0.092, 0.258]	−2551 [−4086, −1028]

Most tested policy-threshold changes had small observed effects around the nominal setting. With the stalled-queue threshold reduced to 12 steps, overall completion changed by −0.010 (95% interval: −0.025 to 0) and PWCT by +207.4 s (−14.5 to +540.7; Supplementary Table S4). TC completion differences remained positive at lifeline-decay values of 0.16, 0.22, and 0.28 (+0.150, +0.167, and +0.142; Supplementary Table S11). Supplementary Table S17 reports the existing 50-cell iteration-budget analysis. Mean overall completion was 0.755, 0.755, 0.750, 0.725, and 0.760 at limits of 1, 2, 4, 8, and 16 iterations, respectively, while the mean episode-specific 95th-percentile replanning latency increased from 391 to 2422 ms and mean objective evaluations increased from 2890 to 5416. These values support four iterations as a bounded computation setting rather than a performance-maximizing choice. Additional sweeps examined the detour ratio, anchor-candidate cap, backup-anchor count, and routine-transfer conditions.

Topology and scale tests complement the local sensitivities. RB evaluates disruption on one fixed Dujiangyan-derived map and task manifest. In the task-scale check, five paired instances at each size and one run per method gave mean weighted-service scores

for ER-HLNS-PR versus RH-ALNS-FR of 0.820 versus 0.769 at 20 tasks (mean paired difference: +0.051; instance range: −0.010 to +0.125), 0.724 versus 0.607 at 50 tasks (+0.117; −0.032 to +0.236), and 0.801 versus 0.348 at 100 tasks (+0.453; +0.405 to +0.487). These descriptive checks extend the evaluation to 100 tasks and motivate broader scaling and geographic tests.

Execution checks complement the service and timing analyses. A replay audit covering 440 episode cells and 275,142 environment steps found no violations of the specified task-route exclusivity, delivered-mass bound, protected-commitment, or terminal-rescue checks (Supplementary Table S12). A separate three-cell shadow audit over 3197 steps recorded zero global material-flow residuals, event-instrumentation mismatches, and truck inventory-field desynchronizations. The two audits address distinct invariants, and Supplementary Table S7a maps terminal-loss, recovery, and feasibility fields so that compatibility exports are not conflated with independently measured outcomes.

The material-flow audit tracks truck inventory, UAV payload, and cumulative delivered mass, with depot transfers recorded as external inflow or outflow. Zero residual means that initial tracked mass plus net depot inflow equals final tracked mass. For audited LB seed 100, a 2500 kg depot replenishment increased tracked mass from 10,400 to 12,900 kg. Delivered mass totaled 5300 kg, and eight UAV reloads were internal transfers. The stepwise accounting residual was 0 kg (Supplementary Table S12).

Planner-call timing was recorded over 60 hard-LB episodes. The mean of episode-specific median latencies was 123.8 ms for ER-HLNS-PR and 290.6 ms for RH-ALNS-FR; the corresponding mean 95th-percentile latencies were 1255.8 and 923.8 ms. ER-HLNS-PR therefore had a lower typical latency but a higher upper-tail latency. Its longest recorded call was 5.61 s among 4080 calls. These planner-only measurements exclude environment stepping and output. Simulation steps last 20 s, and routine replanning occurs every 100 s with additional event-triggered updates. Whole-episode means were 118.5 and 121.4 s, with a paired interval containing zero. Measurements were obtained on an Intel Core i9-13900HX Windows workstation with 16 GB RAM while other experiment matrices were running and characterize the observed computational workload on that platform.

4. Discussion

4.1. Execution-Consistent Planning and Service Trade-Offs

ER-HLNS-PR treats maintenance of an executable plan as a distinct coordination objective. Its state–contract interface identifies work already resolved or physically committed and restricts repair to eligible unfinished decisions. In the direct repair-scope control, suffix-local repair produced fewer task-owner changes, execution rebindings, and repeated reassignments than global re-auction. The completion and PWCT differences in that control spanned zero, so the clearest effect of locality was reduced plan disruption. This separation shows why plan stability and task completion should be evaluated as complementary outcomes.

The principal hard-LB comparison shows how the full framework reallocates service. Relative to mission-priority-matched C-ALNS-MP, TC completion increased by 16.5 percentage points and PWCT decreased by 2444.2 s, while routine completion decreased by 11.5 percentage points. Relative to RH-ALNS-FR, the weighted-service difference was small (+0.015; descriptive 95% interval: −0.021 to +0.056), and the PWCT and runtime intervals also included zero. Global re-auction achieved a modestly higher weighted-service score. Together, these results separate two effects: stronger urgent-service performance relative to the matched baseline and greater plan stability under local repair, with the routine-service and global-reauction trade-offs made explicit. The parallel-rescue and joint-

corridor ablations further describe interacting configuration effects rather than additive component contributions.

4.2. Relation to Prior Work and Operational Use

This work connects tactical routing with the management of ongoing execution. Dong et al. [28] address mobile launch/recovery and multi-point sorties; Zhang and Li [29] incorporate predictive flood and weather effects; Otto et al. [26] study online re-optimization and scouting under incomplete road information; and Li, Song and Li [30] optimize active road verification. ER-HLNS-PR maintains the plan after it has been installed, reconciling realized task, payload, carrier, road, and recovery states while preserving binding commitments. Its contribution lies in this explicit boundary between protected execution and revisable planning decisions.

This positioning also clarifies how learning-based extensions could be incorporated. A POMDP, Bayesian, or reinforcement-learning layer could anticipate future uncertainty and propose new decisions [22,23,36], while the execution-consistency interface would govern how those decisions are installed without breaking commitments already in progress. The protected-versus-revisable decision boundary therefore remains useful whether future uncertainty is handled reactively or predictively.

On the UAV side, the framework contributes joint coordination rather than standalone trajectory optimization. Task–UAV binding, launch eligibility, carrier relation, launch and recovery anchors, cross-truck recovery, payload and energy feasibility, and bounded re-anchoring are updated together with truck routes. The resulting evidence therefore characterizes system-level truck–UAV coordination.

Delivery-mode and mass records clarify the operational meaning of TC completion. Across 60 hard-LB episodes, ER-HLNS-PR completed 531 TC tasks: 222 by airborne delivery, 306 by package release while carried by a truck, and three with unknown mode. C-ALNS-MP completed 412 tasks: 224 airborne and 188 carried. The 119-task difference is reflected primarily in recorded carried-release counts rather than additional airborne deliveries. RH-ALNS-FR completed 522 TC tasks, but its mode labels are unavailable. Accordingly, the mode counts are descriptive rather than causal evidence. Mean delivered mass per episode was 442.5 kg of TC and 4360.0 kg of routine demand for ER-HLNS-PR, 343.3 and 5093.3 kg for C-ALNS-MP, and 435.0 and 4720.0 kg for RH-ALNS-FR. Task-class completion and delivered mass jointly characterize the urgent-versus-bulk service trade-off.

4.3. Limitations

The evaluation covers 20 independent hard-LB task–road instances with three repetitions per method, smaller targeted control sets, and five-instance checks at 20, 50, and 100 tasks. RB varies disruption on one fixed map and task manifest. Consequently, the intervals describe variation among the tested instances, and transfer across cities or disaster types requires broader geographic sampling.

The aircraft, weather, and disruption models are operational proxies designed for controlled comparison. The FC100-referenced profile, weather gates, blockage process, and RB spatial field provide platform and scenario context; field-calibrated endurance, local damage, and safety require dedicated empirical validation. Supplementary Table S7a distinguishes terminal UAV-loss states, feasibility diagnostics, and recovery activations, while the audit results document compliance with the specified simulation invariants.

The evaluated setting fixes all tasks at initialization and keeps blocked roads closed; communication blackouts, split delivery, multiple commodities, heterogeneous aircraft, and temporary depots are outside the present model. Physical inputs and operational-policy thresholds require local calibration for deployment, whereas the structural state–

contract rules are calibration-independent. Because the published integrated truck–UAV method [57] could not be validated under the shared execution rules, it is compared structurally in Supplementary Table S1a,b rather than included in the numerical ranking. Cross-formulation benchmarking, broader operational conditions, and field-calibrated service inputs are priorities for future work.

5. Conclusions

This study develops ER-HLNS-PR for maintaining truck–UAV relief plans as road disruptions are progressively revealed. The framework reconciles realized state, preserves active service and recovery commitments, and repairs eligible unfinished route suffixes through bounded deterministic search. Across 20 independent hard-LB instances with three repetitions per method, ER-HLNS-PR increased TC completion by 16.5 percentage points and reduced PWCT by 2444.2 s relative to mission-priority-matched C-ALNS-MP, while routine completion decreased by 11.5 percentage points. The principal comparison therefore identifies a clear urgent-versus-routine service trade-off under the tested operating conditions.

The direct repair-scope control further reduced task-owner changes, execution rebindings, and repeated reassignment under suffix-local repair relative to global re-auction, while completion and PWCT differences spanned zero. Together, the matched-baseline and repair-scope results separate service allocation from plan stability and support the resolved/protected/eligible decision boundary and state–contract interface as the central execution-management contribution. Further work can add predictive beliefs or learned policies, validated cross-formulation benchmarks, broader geographic settings, calibrated aircraft and hazard models, dynamic demand, and road recovery.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/drones10100778/s1, Figure S1: Same-version LB capability-package and road-blockage sensitivity; Figure S2: Dujiangyan-derived RB road network and fixed task configuration; Table S1a,b: Structured comparison with recent closely related studies; Table S2: Eight exploratory paired five-scenario contrasts (ER-HLNS-PR minus C-ALNS/DR-ALNS); Table S3: ER-HLNS-PR activity counters; Table S4: One-factor ER-HLNS-PR policy sensitivity; Table S5: Event-level mechanism summary; Table S6: Selected parameters required to reconstruct the evaluated LB planning, task, and road-disruption processes; Table S7 and Table S7a: Recorded UAV guard and terminal-energy diagnostic events, and interpretation of UAV terminal, recovery, and feasibility fields, respectively; Table S8: Direct repair-scope control: suffix-local minus global pending-task re-auction; Table S9: Seed-matched objective-evaluation ceiling (E5); Table S10: Conditional truck time-critical eligibility (E6); Table S11: Time-critical lifeline-decay sensitivity (E7); Table S12: Execution and material-flow audits; Table S13: Selected physical and disruption parameter provenance; Table S14: Class-specific delivered/unmet demand mass and recorded time-critical completion modes; Table S15: Mean within-instance standard deviation across three algorithm repetitions; Table S16: The 14 ordered route-construction criteria when mixed coverage is active; Table S17: Existing iteration-budget sensitivity analysis; Table S18: Parameter classes, evidentiary role, tuning status, sensitivity evidence, and deployment interpretation. The supplementary package also provides the processed Source Data files cited in the figure/table captions, deterministic analysis outputs, and the reproducibility archive. References [22,23,26,28,29,31–37,43,44,53,57] cited in the Supplementary Information are included in the main reference list.

Author Contributions: Conceptualization, P.L.; methodology, P.L.; software, P.L.; validation, P.L., S.L., B.G. and Z.L.; formal analysis, P.L.; investigation, P.L.; resources, B.G.; data curation, P.L. and S.L.; writing—original draft preparation, P.L.; writing—review and editing, P.L., D.W., S.L., B.G., Z.L. and A.Y.; visualization, P.L., B.G. and Z.L.; supervision, D.W.; project administration, D.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Office of the Decision-Making Advisory Committee of Shanxi Provincial People's Government, the Shanxi Earthquake Administration, and the Shanxi Academy of Social Sciences under Grant SXFZJZZS2601; and the National Institute of Natural Hazards, Ministry of Emergency Management of China under Grant ZDJ2026-54.

Institutional Review Board Statement: Not applicable. The study is a simulation and algorithm experiment and did not involve human participants or animals.

Informed Consent Statement: Not applicable. No human participants or identifiable personal information were included.

Data Availability Statement: Episode-level and paired Source Data, analysis scripts, executable configurations, seed and repetition protocols, and fixed RB road, POI, and task assets are provided in the Supplementary Materials and accompanying reproducibility archive. The archive identifies the code and data version used for the reported results, with file hashes, expected cell counts, and reproduction commands. Raw execution ledgers are available from the corresponding author on request.

Acknowledgments: During manuscript preparation, the authors used OpenAI Codex (model: GPT-6 Sol; OpenAI, San Francisco, CA, USA; https://openai.com/codex/; accessed on 12 August 2026) for code inspection and checks of reported values against archived outputs. The authors verified the reported results and take full responsibility for the manuscript.

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

References

1. Faiz, T.I.; Vogiatzis, C.; Noor-E-Alam, M. Computational approaches for solving two-echelon vehicle and UAV routing problems for post-disaster humanitarian operations. *Expert Syst. Appl.* **2024**, *237*, 121473. [CrossRef]
2. Dukkanci, O.; Koberstein, A.; Kara, B.Y. Drones for relief logistics under uncertainty after an earthquake. *Eur. J. Oper. Res.* **2023**, *310*, 117–132. [CrossRef]
3. van Steenbergen, R.; Mes, M.; van Heeswijk, W. Reinforcement learning for humanitarian relief distribution with trucks and UAVs under travel time uncertainty. *Transp. Res. Part C Emerg. Technol.* **2023**, *157*, 104401. [CrossRef]
4. Zhang, L.; Ding, Y.; Lin, H. Optimizing synchronized truck-drone delivery with priority in disaster relief. *J. Ind. Manag. Optim.* **2023**, *19*, 5143–5162. [CrossRef]
5. Lu, Y.; Yang, J.; Yang, C. A humanitarian vehicle routing problem synchronized with drones in time-varying weather conditions. *Comput. Ind. Eng.* **2023**, *184*, 109563. [CrossRef]
6. Murray, C.C.; Chu, A.G. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transp. Res. Part C Emerg. Technol.* **2015**, *54*, 86–109. [CrossRef]
7. Agatz, N.; Bouman, P.; Schmidt, M. Optimization Approaches for the Traveling Salesman Problem with Drone. *Transp. Sci.* **2018**, *52*, 965–981. [CrossRef]
8. Wang, Z.; Sheu, J.B. Vehicle routing problem with drones. *Transp. Res. Part B Methodol.* **2019**, *122*, 350–364. [CrossRef]
9. Dell'Amico, M.; Montemanni, R.; Novellani, S. Modeling the flying sidekick traveling salesman problem with multiple drones. *Networks* **2021**, *78*, 303–327. [CrossRef]
10. Luo, Z.; Poon, M.; Zhang, Z.; Liu, Z.; Lim, A. The Multi-visit Traveling Salesman Problem with Multi-Drones. *Transp. Res. Part C Emerg. Technol.* **2021**, *128*, 103172. [CrossRef]
11. Salama, M.R.; Srinivas, S. Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites. *Transp. Res. Part E Logist. Transp. Rev.* **2022**, *164*, 102788. [CrossRef]
12. Thomas, T.; Srinivas, S.; Rajendran, C. Collaborative truck multi-drone delivery system considering drone scheduling and en route operations. *Ann. Oper. Res.* **2024**, *339*, 693–739. [CrossRef]
13. Li, H.; Chen, J.; Wang, F.; Zhao, Y. Truck and drone routing problem with synchronization on arcs. *Nav. Res. Logist. NRL* **2022**, *69*, 884–901. [CrossRef]
14. Liang, Y.J.; Luo, Z.X. A Survey of Truck–Drone Routing Problem: Literature Review and Research Prospects. *J. Oper. Res. Soc. China* **2022**, *10*, 343–377. [CrossRef]
15. Macrina, G.; Di Puglia Pugliese, L.; Guerriero, F.; Laporte, G. Drone-aided routing: A literature review. *Transp. Res. Part C Emerg. Technol.* **2020**, *120*, 102762. [CrossRef]
16. Dang, S.; Liu, Y.; Luo, Z.; Liu, Z.; Shi, J. A Survey of the Routing Problem for Cooperated Trucks and Drones. *Drones* **2024**, *8*, 550. [CrossRef]

17. Duan, J.; Luo, H.; Wang, G. Approaches to the truck-drone routing problem: A systematic review. *Swarm Evol. Comput.* **2025**, *92*, 101825. [CrossRef]
18. Wang, Y.; Wang, Z.; Hu, X.; Xue, G.; Guan, X. Truck–drone hybrid routing problem with time-dependent road travel time. *Transp. Res. Part C Emerg. Technol.* **2022**, *144*, 103901. [CrossRef]
19. Yang, Y.; Yan, C.; Cao, Y.; Roberti, R. Planning robust drone-truck delivery routes under road traffic uncertainty. *Eur. J. Oper. Res.* **2023**, *309*, 1145–1160. [CrossRef]
20. Ghiasvand, M.R.; Rahmani, D.; Moshref-Javadi, M. Data-driven robust optimization for a multi-trip truck-drone routing problem. *Expert Syst. Appl.* **2024**, *241*, 122485. [CrossRef]
21. Deng, M.; Li, Y.; Ding, J.; Zhou, Y.; Zhang, L. Stochastic and robust truck-and-drone routing problems with deadlines: A Benders decomposition approach. *Transp. Res. Part E Logist. Transp. Rev.* **2024**, *190*, 103709. [CrossRef]
22. Peng, W.; Wang, D.; Yin, Y.; Cheng, T.C.E. Multi-agent deep reinforcement learning-based truck-drone collaborative routing with dynamic emergency response. *Transp. Res. Part E Logist. Transp. Rev.* **2025**, *195*, 103974. [CrossRef]
23. Cui, H.; Li, K.; Jia, S.; Meng, Q. Dynamic collaborative truck-drone delivery with en-route synchronization and random requests. *Transp. Res. Part E Logist. Transp. Rev.* **2024**, *192*, 103802. [CrossRef]
24. Weng, X.; She, W.; Fan, H.; Zhang, J.; Yun, L. Multi-depot vehicle routing problem with drones in emergency logistics. *Clust. Comput.* **2025**, *28*, 64. [CrossRef]
25. Sun, J.; Lin, G.H.; Zhu, X. A hybrid drone–truck model for fair and efficient emergency relief distribution. *Comput. Ind. Eng.* **2026**, *211*, 111607. [CrossRef]
26. Otto, A.; Golden, B.; Lorenz, C.; Luo, Y.; Pesch, E.; Rocha, L.A. On delivery policies for a truck-and-drone tandem in disaster relief. *IISE Trans.* **2025**, *57*, 1198–1214. [CrossRef]
27. Sun, W.; Wu, L.; Zhang, F. An Exact Algorithm to Solve Vehicle Routing Problem with Drones for Delivery and Surveillance Tasks After Disasters. *Transp. Sci.* **2026**, *60*, 132–154. [CrossRef]
28. Dong, X.; Gao, B.; Liu, R. Path Optimization for Multi-Vehicle and Multi-UAV Collaborative Delivery in Flood Rescue Under Road Disruptions: A Case Study of the 2024 Guangdong Flood Disaster. *Drones* **2026**, *10*, 322. [CrossRef]
29. Zhang, Y.; Li, N. Collaborative route optimization for urban emergency vehicles and drones under dynamic evolution of heavy rainfall-induced urban flooding. *J. Ind. Manag. Optim.* **2026**, *22*, 3789–3834. [CrossRef]
30. Li, H.; Song, X.; Li, Y. Bayesian active verification of lifeline roads for risk aware emergency logistics with trucks and drones. *Discov. Artif. Intell.* **2026**, *6*, 962. [CrossRef]
31. Xing, J.; Guo, T.; Tong, L.C. Reliable truck-drone routing with dynamic synchronization: A high-dimensional network programming approach. *Transp. Res. Part C Emerg. Technol.* **2024**, *165*, 104698. [CrossRef]
32. Zhao, J.; Long, Y.; Xie, B.; Xu, G.; Liu, Y. A matheuristic solution for efficient scheduling in dynamic truck–drone collaboration. *Expert Syst. Appl.* **2025**, *267*, 126218. [CrossRef]
33. Yang, X.; Cao, W.; Wang, K.; Yin, H.; Wu, J.; Wu, L. Integrated scheduling of truck and drone fleets for cargo transportation in post-disaster relief: A two-stage stochastic optimization approach. *Transp. Res. Part E Logist. Transp. Rev.* **2025**, *196*, 104015. [CrossRef]
34. Xu, X.; Wang, S.; Zhang, T. Truck-drone supported humanitarian relief logistics network design: A two-stage distributionally robust optimization approach. *Transp. Res. Part C Emerg. Technol.* **2025**, *178*, 105231. [CrossRef]
35. Sun, W.; Wu, L.; Zhang, F. Robust optimization for truck-and-drone collaboration with travel time uncertainties. *Transp. Res. Part B Methodol.* **2026**, *204*, 103378. [CrossRef]
36. Li, H.; Li, Y.; Song, X.; Li, Y. Bayesian belief-to-action planning for truck–drone emergency routing. *Discov. Artif. Intell.* **2026**, *6*, 1283. [CrossRef]
37. Li, H.; Li, Y.; Zhang, J. Multi-Stage Optimization of Multi-Truck-Drone Cooperative Emergency Delivery Under Progressive Road Recovery. *IEEE Access* **2026**, *14*, 84618–84633. [CrossRef]
38. Ropke, S.; Pisinger, D. An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. *Transp. Sci.* **2006**, *40*, 455–472. [CrossRef]
39. Sacramento, D.; Pisinger, D.; Ropke, S. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. *Transp. Res. Part C Emerg. Technol.* **2019**, *102*, 289–315. [CrossRef]
40. Chen, C.; Demir, E.; Huang, Y. An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots. *Eur. J. Oper. Res.* **2021**, *294*, 1164–1180. [CrossRef]
41. Mulumba, T.; Najy, W.; Diabat, A. The drone-assisted pickup and delivery problem: An adaptive large neighborhood search metaheuristic. *Comput. Oper. Res.* **2024**, *161*, 106435. [CrossRef]
42. Mara, S.T.W.; Rifai, A.P.; Sopha, B.M. An adaptive large neighborhood search heuristic for the flying sidekick traveling salesman problem with multiple drops. *Expert Syst. Appl.* **2022**, *205*, 117647. [CrossRef]
43. Li, K.; Cui, H.; Meng, Q.; Zhang, X. Dynamic truck-drone collaborative transportation during hurricanes. *Transp. Res. Part C Emerg. Technol.* **2025**, *180*, 105322. [CrossRef]

44. Shi, Y.; Zhang, K.; Zhang, J.; Chen, M.; Yang, J.; Guo, H. Dynamic truck-drone cooperative delivery of emergency supplies considering secondary disasters. *Transp. Res. Part E Logist. Transp. Rev.* **2026**, *206*, 104543. [CrossRef]
45. Liu, Y.; Fei, Y.; Ma, C. Advances in truck–drone cooperative delivery routing and coordination: A review. *Eur. Transp. Res. Rev.* **2026**, *18*, 49. [CrossRef]
46. Mitsubishi FUSO. Canter 4x4. Available online: https://www.fuso.com.pe/camiones/livianos/canter-4x4 (accessed on 12 August 2026).
47. Ministry of Emergency Management of the People's Republic of China. Guidelines for the Construction of Emergency Shelters. 2023. Available online: https://www.mem.gov.cn/gk/zfxxgkpt/fdzdgknr/202312/t20231226_473145.shtml (accessed on 25 September 2026).
48. Jeong, H.Y.; Song, B.D.; Lee, S. Truck-drone hybrid delivery routing: Payload-energy dependency and No-Fly zones. *Int. J. Prod. Econ.* **2019**, *214*, 220–233. [CrossRef]
49. Ito, S.; Akaiwa, K.; Funabashi, Y.; Nishikawa, H.; Kong, X.; Taniguchi, I.; Tomiyama, H. Load and Wind Aware Routing of Delivery Drones. *Drones* **2022**, *6*, 50. [CrossRef]
50. Ito, S.; Nishikawa, H.; Kong, X.; Funabashi, Y.; Shibata, A.; Negoro, S.; Taniguchi, I.; Tomiyama, H. Energy-aware Routing of Delivery Drones under Windy Conditions. *IPSJ Trans. Syst. LSI Des. Methodol.* **2021**, *14*, 30–39. [CrossRef]
51. Xia, Y.; Wu, T.; Xia, B.; Zhang, J. Truck-Drone Pickup and Delivery Problem with Drone Weight-Related Cost. *Sustainability* **2023**, *15*, 16342. [CrossRef]
52. Zhang, S.; Liu, S.; Xu, W.; Wang, W. A novel multi-objective optimization model for the vehicle routing problem with drone delivery and dynamic flight endurance. *Comput. Ind. Eng.* **2022**, *173*, 108679. [CrossRef]
53. DJI. DJI FlyCart 100: Specifications. Available online: https://www.dji.com/flycart-100/specs (accessed on 12 August 2026).
54. DJI. DJI FlyCart 100: Frequently Asked Questions. Available online: https://www.dji.com/flycart-100/faq (accessed on 12 August 2026).
55. DJI. DJI Redefines Aerial Delivery with DJI FlyCart 100. 4 December 2025. Available online: https://www.prnewswire.com/news-releases/dji-redefines-aerial-delivery-with-dji-flycart-100-302633039.html (accessed on 12 August 2026).
56. DJI. *DJI FlyCart 100 Safety Guidelines, Version 1.0*; DJI: Shenzhen, China, 2025. Available online: https://dl.djicdn.com/downloads/DJI_FlyCart_100/20251127/DJI_FlyCart_100_Safety_Guidelines_v1.0_i.pdf (accessed on 12 August 2026).
57. Long, Y.; Xu, G.; Zhao, J.; Xie, B.; Fang, M. Dynamic Truck-UAV Collaboration and Integrated Route Planning for Resilient Urban Emergency Response. *IEEE Trans. Eng. Manag.* **2024**, *71*, 9826–9838. [CrossRef]

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