Robust Optimization for the Two-Echelon Crowdsourced Pickup and Delivery Problem with Time Windows under Travel Time Uncertainty

Dongze Li*[1] and Fangni Zhang[1]

[1]Department of Data and Systems Engineering, The University of Hong Kong, Hong Kong, China

Short summary

This study presents a robust two-echelon crowdsourced pickup and delivery problem with time windows (R2E-CPDPTW) to optimize routing and order allocation between first-echelon vehicles and second-echelon crowdsourced and contracted vehicles under travel time uncertainty. Formulated as a route-based set-partitioning model, it employs a cardinality-constrained uncertainty set to characterize travel time variations and preserve strict temporal synchronization between the two echelons. To solve this complex model to optimality, a customized branch-price-and-cut (BPC) exact algorithm is developed, where the pricing subproblem is solved as a robust shortest path problem with resource constraints. Computational results demonstrate the algorithm's efficiency in yielding exact solutions and highlight the critical trade-offs between system robustness and operational costs.

Keywords: Branch-price-and-cut, Crowdshipping, Robust optimization, Two-echelon vehicle routing, Pickup and delivery problem.

1 Introduction

In the era of on-demand economies, the growing consumer demand for rapid, end-to-end intracity services necessitates the efficient management of entire freight flows from pickups to deliveries. To meet this demand, modern urban logistics increasingly adopt the two-echelon pickup and delivery (2E-PDP) framework. In such systems, satellites serve as intermediate consolidation facilities distributed across the network. First-echelon (FE) vehicles transport consolidated parcels between these satellites, while second-echelon (SE) vehicles perform regional pickup and delivery tasks. When pickup and delivery locations are geographically distant and reside in different satellite service areas, SE vehicles first retrieve packages back to the origin satellite, transferring them via FE vehicles to the destination satellite, and dispatching local SE vehicles for final delivery. This hierarchical structure empowers logistics providers to exploit economies of scale, significantly enhancing routing efficiency and lowering system-wide mileage. Meanwhile, to alleviate the pressure of rising operational costs and traffic congestion, crowdsourced delivery has emerged as a promising paradigm. By tapping into the idle capacity of uncontracted travelers, crowdshipping injects high flexibility and cost-effectiveness into the 2E-PDP, allowing crowdsourced couriers to collaborate with contracted vehicles in SE to fulfill complex pickup and delivery tasks via transshipment at satellites and lockers.

Despite these benefits, integrating crowdsourced delivery into the 2E-PDP framework introduces critical vulnerabilities. For example, the seamless coordination required in two-echelon systems relies heavily on strict temporal synchronization at intermediate satellites. However, the highly stochastic nature of urban travel times, compounded by the profound behavioral and operational uncertainties inherent in crowdsourced couriers, poses formidable challenges to maintaining this fragile synchronization.

To address these challenges, this study proposes a robust two-echelon crowdsourced pickup and delivery problem with time windows (R2E-CPDPTW). Building upon the pickup and delivery problem with crowdshipping, transshipment, and consolidation (PDPCTC) model introduced by Li et

al. (2026), this research focuses exclusively on intracity pickup and delivery. Specifically, FE vehicles manage inter-satellite consolidation, while SE operations are jointly executed by contracted vehicles and crowdsourced couriers. Crucially, it incorporates travel time uncertainty by employing a cardinality-constrained uncertainty set, assigning a larger variation margin to crowdsourced couriers to capture their inherent unpredictability. Furthermore, while Li et al. (2026) relied on an Adaptive Large Neighborhood Search (ALNS) heuristic, we develop a customized branch-price-and-cut (BPC) exact algorithm to solve this complex model to optimality.

Fig. 1 illustrates an operational example of the R2E-CPDPTW framework. The network comprises six parcels to be picked up and delivered, three intermediate satellites (S_1, S_2, S_3) bridging FE and SE operations, and a locker facilitating transshipment between crowdsourced couriers and contracted vehicles in SE transportation. For all the pickup and delivery requests, an SE contracted vehicle firstly departs from S_2 to handle parcel 1 locally, sequentially visiting its pickup node p_1 and delivery node d_1. Meanwhile, parcels 2 through 5 are collected and returned to S_2 for further FE consolidation. From S_2, parcels 2 and 3 are consolidated via an FE vehicle to S_1. Parcel 4, after arriving at S_2, is retrieved by a crowdsourced courier for final delivery to destination d_4. Concurrently, parcel 5 is transferred via an FE vehicle to S_3 for further SE delivery. At S_3, an SE contracted vehicle is dispatched with parcel 5. En route to delivering parcel 5, the vehicle picks up parcel 6 and routes it back to S_3, from where an FE vehicle transports it to S_1. Upon receiving parcels 2, 3, and 6 at S_1, a local SE contracted vehicle departs to directly deliver parcels 2 and 6, while depositing parcel 3 into the locker. Subsequently, a crowdsourced courier retrieves parcel 3 from the locker to complete its delivery to its final destination d_3. Crucially, every pickup and delivery node imposes strict time windows, and this entire sequence of synchronized routing, consolidation, and transshipment operations is executed without temporal violation.

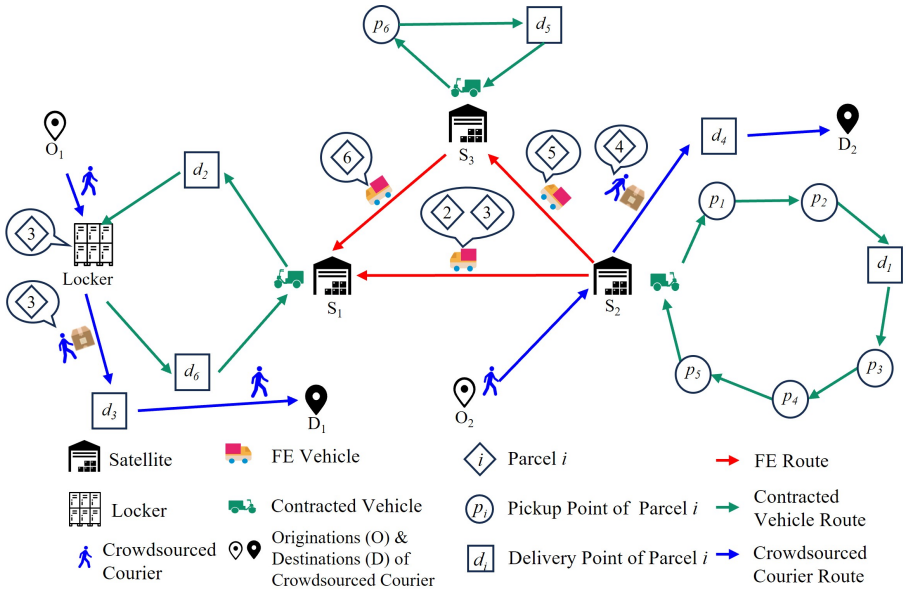


Figure 1: An example of the R2E-CPDPTW model

The main contributions of this paper are summarized as follows:

First, building upon the PDPCTC proposed in Li et al. (2026), we propose R2E-CPDPTW. To the best of our knowledge, this is the first study to explicitly incorporate the heterogeneous travel time uncertainties associated with distinct courier types. Given the strict temporal synchronization demanded by two-echelon logistics, our robust approach significantly enhances the system's resilience against temporal disruptions while maintaining operational cost-efficiency.

Second, we formulate the problem as a route-based set-partitioning model that explicitly captures the complex interdependencies and strict temporal synchronization required across the FE and SE, particularly given the integration of crowdsourced couriers into the SE operations. To solve this intricate model to guaranteed optimality, we develop a tailored branch-price-and-cut (BPC) algorithm. This framework employs a resource-constrained bidirectional labeling procedure to

efficiently solve the robust pricing subproblems, introduces a set of valid inequalities to tighten the lower bounds, and incorporates a hierarchical branching strategy to effectively break symmetry and accelerate overall convergence.

Third, computational experiments validate the effectiveness of the proposed algorithm and provide managerial insights into the trade-offs between robustness and operational costs.

2 Problem description and model formulation

Problem settings

The R2E-CPDPTW is defined on a directed network $G = (V, A)$, where the vertex set is given by $V = V^S \cup V^C \cup V_O^R \cup V_D^R \cup V^L$. Specifically, V^S is the set of satellites, and $V^C = V^P \cup V^D$ is the set of customer nodes, comprising pickup locations V^P and delivery locations V^D. Let R denote the set of available crowdsourced couriers. The sets $V_O^R = \{O_i : i \in R\}$ and $V_D^R = \{D_i : i \in R\}$ define the respective origins and destinations of these couriers. Furthermore, V^L represents the set of parcel lockers, which, together with V^S, constitute the transshipment node set $V^T = V^S \cup V^L$, serving as eligible pickup locations for crowdsourced couriers. For modeling convenience, each satellite $s \in V^S$ is split into an origin node s_O and a destination node s_D to represent the start and end points of the SE contracted vehicle routes. Let $V_O^S = \{s_O : s \in V^S\}$ and $V_D^S = \{s_D : s \in V^S\}$ denote the sets of these split nodes.

Let $\mathcal{P}$ denote a set of n packages. Each package $i \in \mathcal{P}$ pairs a pickup node i with a delivery node $i+n$, yielding the pickup node set $V^P = \{1, \ldots, n\}$ and the delivery node set $V^D = \{n+1, \ldots, 2n\}$. Each customer node $i \in V^C$ requires a service time τ_i and must be serviced within a specific time window $[e_i, l_i]$. Any contracted vehicle or crowdsourced courier arriving prior to the earliest start time e_i incurs waiting time at no additional cost. Intermediate satellites operate without restrictive time windows. The overall arc set $A = A^1 \cup A^2 \cup A^3$ represents the viable links for the entire network. Specifically, $A^1 = \{(s_1, s_2) : s_1, s_2 \in V^S, s_1 \neq s_2\}$ facilitates inter-satellite FE consolidations. The SE operations are jointly performed by contracted vehicles and crowdsourced couriers, whose respective routing links are captured by A^2 and A^3. The arc set $A^2 = \{(s_O, i), (i, s_D) : s \in V^S, i \in V^C \cup V^T\} \cup \{(i, j) : i, j \in V^C \cup V^T, i \neq j\}$ accommodates the routing of contracted vehicles between satellites and customers. For the crowdsourced workforce, the arc set is defined as $A^3 = \{(r_O, i), (i, j), (j, r_D) : r_O \in V_O^R, i \in V^T, j \in V^D, r_D \in V_D^R\}$. Given the absence of formal labor contracts, crowdsourced couriers are strictly limited to retrieving parcels exclusively from nodes in V^T and are dedicated solely to fulfilling final delivery tasks. Finally, traversing each arc $(i, j) \in A$ incurs a transportation cost c_{ij} and a travel time t_{ij}.

To immunize the routing plan against travel time uncertainties, we adopt the cardinality-constrained uncertainty set introduced by Bertsimas & Sim (2004). For each traversed arc $(i, j) \in A$, the actual travel time $\tilde{t}_{ij}$ is bounded within $[t_{ij}, t_{ij} + \hat{t}_{ij}]$, where t_{ij} is the nominal travel time and $\hat{t}_{ij}$ is the maximum possible delay. To capture the heterogeneous reliability across different transportation modes and workforces, we assign distinct maximum delay parameters to each arc set: $\hat{t}_{ij}^1$ for A^1, $\hat{t}_{ij}^2$ for A^2, and $\hat{t}_{ij}^3$ for A^3. Given that FE consolidations operate on relatively stable routes, whereas crowdsourced couriers exhibit the highest behavioral and operational unpredictability in urban settings, we explicitly enforce the relationship $\hat{t}_{ij}^1 < \hat{t}_{ij}^2 < \hat{t}_{ij}^3$. Furthermore, instead of imposing a universal robustness level across the entire network, we introduce budget parameters Γ^1, Γ^2, and Γ^3 to strictly bound the cumulative temporal deviations for the FE, contracted SE, and crowdsourced SE operations, respectively. Let ρ_{ij} be an auxiliary continuous variable representing the scaled travel time deviation on arc (i, j). The actual travel time vectors $\tilde{t}^1$, $\tilde{t}^2$, and $\tilde{t}^3$ are mathematically governed by the following independent cardinality-constrained uncertainty sets:

$$\mathscr{U}^k = \left\{ \tilde{t}^k : \ \tilde{t}_{ij}^k = t_{ij} + \rho_{ij}\hat{t}_{ij}^k, \quad 0 \leq \rho_{ij} \leq 1, \ \forall (i,j) \in A^k, \quad \sum_{(i,j)\in A^k} \rho_{ij} \leq \Gamma^k \right\}, \quad \forall k \in \{1,2,3\}. \tag{1}$$

Route-based formulation

To more efficiently solve the problem using the BPC algorithm, we model the R2E-CPDPTW as a route-based formulation. Let $\mathcal{L}_s$ denote the set of all feasible SE routes originating and terminating at satellite $s \in V^S$. Each route $r \in \mathcal{L}_s$ executes a comprehensive sequence of pickup and delivery operations, encompassing transshipments between contracted vehicles and crowdsourced couriers, as well as the subsequent crowdsourced deliveries. This incurs a total cost c_r that accounts for both the fixed usage cost of the contracted fleet (f_2) and the variable routing costs generated by both contracted vehicles and crowdsourced couriers. Thus, for a route $r \in \mathcal{L}_s$, c_r is calculated by:

$$c_r = f_2 + \sum_{i \in V^C \cup V^T \cup \{s_O\}} \sum_{j \in V^C \cup V^T \cup \{s_D\}} c_{ij}\phi_{ijr} + \sum_{p \in V^T} \sum_{i \in V^D} \sum_{k \in R} (c_{O_k p} + c_{pi} + c_{iD_k})\theta_{pikr}. \tag{2}$$

Let ϕ_{ijr} be a binary parameter indicating whether route $r \in \mathcal{L}_s$ traverses arc $(i, j) \in A^2$. Similarly, let θ_{pikr} be a binary parameter that equals 1 if a crowdsourced courier $k \in R$ retrieves parcel $i \in V^D$ from transshipment locker $p \in V^T$ to perform the final delivery on route r, and 0 otherwise. Let E_r and L_r denote the feasible earliest return time and latest departure time of route r, respectively. The parameter a_{ir} equals 1 if node i is visited by route r, and 0 otherwise. To capture FE consolidation demands, the parameter b_{ir} equals 1 if node i is visited by SE route r while its corresponding counterpart (i.e., node $i+n$ for a pickup $i \in V^P$, or node $i-n$ for a delivery $i \in V^D$) is not visited by the same route, indicating that the package requires inter-satellite transfer. Three types of decision variables are employed in the master problem. Let y_{sr} be a binary variable equal to 1 if SE route $r \in \mathcal{L}_s$ is selected, and 0 otherwise. Let $x^i_{ss'}$ be a binary variable equal to 1 if package $i \in \mathcal{P}$ is consolidated from satellite s to s' via an FE vehicle, and 0 otherwise. Finally, let $z_{ss'} \in \mathbb{Z}_+$ denote the number of FE vehicles dispatched from s to s'. The R2E-CPDPTW is formulated as follows:

$$\min \quad \sum_{s \in V^S} \sum_{r \in \mathcal{L}_s} c_r y_{sr} + \sum_{s \in V^S} \sum_{s' \in V^S \setminus \{s\}} \hat{c}_{ss'} z_{ss'} \tag{3a}$$

$$\text{s.t.} \quad \sum_{s \in V^S} \sum_{r \in \mathcal{L}_s} a_{ir} y_{sr} = 1, \quad \forall i \in V^P, \tag{3b}$$

$$\sum_{s \in V^S} \sum_{r \in \mathcal{L}_s} \left(a_{ir} + \sum_{p \in V^T} \sum_{k \in R} \theta_{pikr} \right) y_{sr} = 1, \quad \forall i \in V^D, \tag{3c}$$

$$\sum_{s \in V^S} \sum_{r \in \mathcal{L}_s} \sum_{p \in V^T} \sum_{i \in V^D} \theta_{pikr} y_{sr} \leq 1, \quad \forall k \in R, \tag{3d}$$

$$\sum_{r \in \mathcal{L}_s} b_{ir} E_r y_{sr} \leq \sum_{s' \in V^S \setminus \{s\}} \sum_{r' \in \mathcal{L}_{s'}} b_{i+n,r'} (L_{r'} - \tilde{t}^1_{ss'}) y_{s'r'}, \quad \forall s \in V^S, \forall i \in V^P, \tag{3e}$$

$$\sum_{r \in \mathcal{L}_s} b_{ir} y_{sr} = \sum_{s' \in V^S \setminus \{s\}} x^i_{ss'}, \quad \forall s \in V^S, \forall i \in V^P, \tag{3f}$$

$$\sum_{r' \in \mathcal{L}_{s'}} b_{i+n,r'} y_{s'r'} = \sum_{s \in V^S \setminus \{s'\}} x^i_{ss'}, \quad \forall s' \in V^S, \forall i \in V^P, \tag{3g}$$

$$z_{ss'} \geq \sum_{i \in C} x^i_{ss'}, \quad \forall C \in \mathcal{C}_{ss'}, \forall s, s' \in V^S : s \neq s', \tag{3h}$$

$$y_{sr} \in \{0, 1\}, \quad \forall s \in V^S, \forall r \in \mathcal{L}_s, \tag{3i}$$

$$x^i_{ss'} \in \{0, 1\}, \quad \forall s, s' \in V^S : s \neq s', \forall i \in V^P, \tag{3j}$$

$$z_{ss'} \in \mathbb{Z}_+, \quad \forall s, s' \in V^S : s \neq s'. \tag{3k}$$

The objective function (3a) minimizes the total system cost, which comprises the execution costs of the selected SE routes and the deployment costs of the inter-satellite FE vehicles. Constraints (3b) guarantee that every pickup location is visited exactly once by a contracted vehicle. Constraints (3c) enforce that each delivery demand is fulfilled, either directly by a contracted vehicle or relayed to a crowdsourced courier via a transshipment nodes. Constraints (3d) restrict each crowdsourced courier $k \in R$ to undertaking at most one delivery task. Constraints (3e) enforce the critical temporal synchronization for split requests. They dictate that the FE departure for the delivery leg at satellite s' must occur after the parcel has been picked up and transferred from satellite s, accounting for the FE inter-satellite travel time under uncertainty $\tilde{t}^1_{ss'}$. Furthermore, Constraints

(3f) and (3g) maintain spatial flow conservation by seamlessly linking the SE routing with the FE consolidation. Specifically, any parcel whose pickup and delivery operations are decoupled across different routes must be transported via an FE vehicle between the respective satellites. Crucially, Constraints (3h) determine the minimum number of FE vehicles required to facilitate the inter-satellite parcel consolidation. This is achieved by evaluating the maximal cliques $C \in \mathcal{C}_{ss'}$ in the conflict graph, ensuring that mutually incompatible packages (due to overlapping time windows or capacity violations) are not assigned to the same FE vehicle. Due to space constraints, the detailed separation procedure for generating these clique cuts is omitted. Finally, Constraints (3i)–(3k) define the variable domains.

3 Methodology

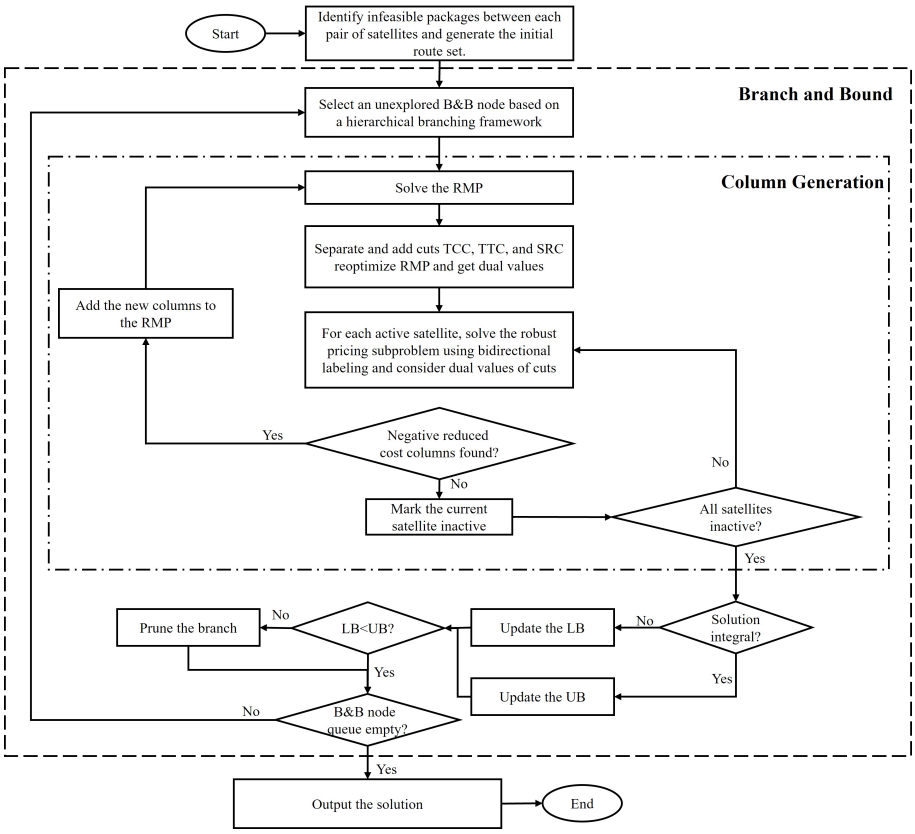


Figure 2: The flowchart of the proposed BPC algorithm

Fig. 2 illustrates the flowchart of the proposed BPC algorithm. At its core, column generation operates as an iterative procedure to solve the linear relaxation of the route-based formulation (3a)–(3k), commonly referred to as the restricted master problem (RMP). A key algorithmic innovation lies in the development of a hierarchical branching framework tailored to the specific problem structure. Furthermore, to strengthen the linear relaxation and improve the lower bound quality, we introduce two novel classes of valid inequalities: Truck Clique Cuts (TCCs), derived from the consolidation incompatibilities of packages on FE vehicles, and Time Threshold Cuts (TTCs), which capture the impact of inter-satellite package transfers on the overall SE temporal synchronization. These are integrated alongside classical Subset-Row Cuts (SRCs) to improve the lower bound quality of the proposed algorithm. Due to space limitations, the detailed separation algorithms for these cuts are omitted here.

We apply a bidirectional labeling approach to solve the pricing subproblems. Referring to Munari et al. (2019), a forward label ℓ_i^F with index i is defined as $\ell_i^F = (\delta_i^F, \bar{c}_i^F, R_{i,0}^{t,F}, \ldots, R_{i,\Gamma^2}^{t,F}, q_i^F, o_i^F, \eta_i^F, \mathcal{V}_i^F)$, where δ_i^F is the current node, $\bar{c}_i^F$ is the accumulated reduced cost, $R_{i,\gamma}^{t,F}$ for $\gamma = 0, \ldots, \Gamma^2$ is the worst-case earliest service start time at δ_i^F when up to γ contracted vehicle arcs attain their

maximum deviation, q_i^F is the accumulated vehicle load, o_i^F records the current load at each transshipment node $p \in V^T$, η_i^F encodes the crowdsourced courier assignments specifying which courier retrieves which parcel from which transshipment node, and $\mathcal{V}_i^F$ is the set of visited nodes.

Given label ℓ_i^F at node δ_i^F, an extension along a feasible arc (δ_i^F, j) to a customer node $j \in V^C$ creates a new label ℓ_{i+1}^F with $\delta_{i+1}^F = j$, $\bar{c}_{i+1}^F = \bar{c}_i^F + \bar{d}_{\delta_i^F,j}$, $q_{i+1}^F = q_i^F + q_j$, and $\mathcal{V}_{i+1}^F = \mathcal{V}_i^F \cup \{j\}$, where $\bar{d}_{\delta_i^F,j}$ is the reduced cost of the arc. The robust time resources are updated as $R_{i+1,0}^{t,F} = \max\{e_j, R_{i,0}^{t,F} + s_{\delta_i^F} + t_{\delta_i^F,j}\}$ and, for $\gamma = 1, \ldots, \Gamma^2$,

$$R_{i+1,\gamma}^{t,F} = \max\left\{e_j,\ R_{i,\gamma}^{t,F} + s_{\delta_i^F} + t_{\delta_i^F,j},\ R_{i,\gamma-1}^{t,F} + s_{\delta_i^F} + t_{\delta_i^F,j} + \hat{t}_{\delta_i^F,j}^2\right\}. \tag{4}$$

The extension is feasible only if $j \notin \mathcal{V}_i^F$, $R_{i+1,\Gamma^2}^{t,F} \le l_j$, and $q_{i+1}^F \le Q^{SE}$.

When the contracted vehicle visits a transshipment node $p \in V^T$ to deposit a parcel for delivery node $u \in V^D$ assigned to crowdsourced courier $k \in R$, the new label has $\delta_{i+1}^F = p$, $\bar{c}_{i+1}^F = \bar{c}_i^F + \bar{d}_{\delta_i^F,p} + B_{kpu}$, $q_{i+1}^F = q_i^F + q_u$ (noting $q_u < 0$ for delivery nodes), $o_{i+1}^F(p) = o_i^F(p) - q_u$, $\eta_{i+1}^F = \eta_i^F \cup \{(p,u,k)\}$, and $\mathcal{V}_{i+1}^F = \mathcal{V}_i^F \cup \{u\}$, where B_{kpu} is the reduced assignment cost for crowdsourced courier k retrieving parcel u from transshipment node p. The time resources $R_{i+1,\gamma}^{t,F}$ are updated via the same recursive equation for arc (δ_i^F, p).

The feasibility of this extension requires checking the temporal synchronization between the contracted vehicle and the crowdsourced courier at locker p. The crowdsourced courier can depart p only after both the parcel has been deposited by the contracted vehicle and the courier has arrived from O_k. $R_{i+1,\Gamma^2}^{t,F}$ represents the worst-case time at which the parcel becomes available at p (governed by Γ^2). The earliest time at which the crowdsourced courier can begin retrieving the parcel from p, under γ_3 worst-case deviations on the courier's arcs, is:

$$w_{p,\gamma_3} = \max\left(R_{i+1,\Gamma^2}^{t,F},\ t_{O_k,p} + \min(\gamma_3, 1) \cdot \hat{t}_{O_k,p}^3\right), \quad \gamma_3 = 0, \ldots, \Gamma^3. \tag{5}$$

Here, the max reflects that retrieval cannot begin until both the contracted vehicle has deposited the parcel and the courier has arrived at p. The worst-case arrival time at delivery node u is then:

$$w_{u,\gamma_3} = \max\left(e_u,\ w_{p,\gamma_3} + s_p + t_{p,u},\ w_{p,\gamma_3-1} + s_p + t_{p,u} + \hat{t}_{p,u}^3\right), \quad \gamma_3 = 1, \ldots, \Gamma^3, \tag{6}$$

with $w_{u,0} = \max(e_u,\ w_{p,0} + s_p + t_{p,u})$, where s_p is the service time at locker p for parcel retrieval. This extension is feasible only if $u \notin \mathcal{V}_i^F$, $q_{i+1}^F \ge 0$, $o_{i+1}^F(p) \le Q^L$, and $w_{u,\Gamma^3} \le l_u$.

A label $\ell_i^{F,1}$ dominates $\ell_i^{F,2}$ if $\delta_i^{F,1} = \delta_i^{F,2}$, $\bar{c}_i^{F,1} \le \bar{c}_i^{F,2}$, $R_{i,\gamma}^{t,F,1} \le R_{i,\gamma}^{t,F,2}$ for all $\gamma = 0, \ldots, \Gamma^2$, $q_i^{F,1} \le q_i^{F,2}$, $o_i^{F,1}(p) \le o_i^{F,2}(p)$ for all $p \in V^T$, $\mathcal{K}_i^{F,1} \subseteq \mathcal{K}_i^{F,2}$, and $\mathcal{V}_i^{F,1} \subseteq \mathcal{V}_i^{F,2}$, where $\mathcal{K}_i^F = \{k \in R : \exists (p,u,k) \in \eta_i^F\}$ denotes the set of crowdsourced couriers already assigned in the label. The backward label and merging procedure are defined symmetrically, with $R_{i,\gamma}^{t,B}$ representing the latest feasible departure time under γ worst-case deviations.

4 Results and discussion

All experiments are conducted on a computer with an AMD Ryzen 7 5800H CPU operating at 3.2 GHz and equipped with 16 GB of RAM. The BPC algorithm is implemented in Python using Gurobi 11.0 as the LP solver, with a time limit of 3600 seconds per instance. We test on instances with $|\mathcal{P}| \in \{15, 20, 25\}$ packages, $|V^S| \in \{3, 4\}$ satellites, and layout types r (random), c (clustered), rc (mixed), with 5 instances per configuration. The robustness parameters are set as $\Gamma^1 = 1$, $\Gamma^2 = 2$, $\Gamma^3 = 1$, $\hat{t}_{ij}^1 = 0.05 t_{ij}$, $\hat{t}_{ij}^2 = 0.10 t_{ij}$, and $\hat{t}_{ij}^3 = 0.20 t_{ij}$.

Table 1 compares the BPC algorithm against Gurobi applied to the arc-based compact formulation. For each group, we report the average optimality gap and solution time over 10 instances, along with the number of instances solved to optimality (#Opt).

The BPC algorithm solves all P15 and P20 instances to optimality, with average solution times under 75 seconds for P15 and under 9 minutes for P20. In contrast, Gurobi solves all P15 instances

Table 1: BPC vs. Gurobi on the compact formulation (3600s time limit)

		BPC			Gurobi (compact)		
Type	$\|\mathcal{P}\|$-$\|V^S\|$	Gap%	Time(s)	#Opt	Gap%	Time(s)	#Opt
c	P15-S3	0.00	14.2	10	0.00	218.5	10
c	P20-S3	0.00	92.4	10	3.82	3600	4
c	P25-S3	1.43	548.7	9	9.45	3600	0
c	P15-S4	0.00	21.5	10	0.35	2841.3	7
c	P20-S4	0.00	156.3	10	6.18	3600	1
c	P25-S4	2.06	924.1	8	13.72	3600	0
r	P15-S3	0.00	48.6	10	1.24	3012.7	6
r	P20-S3	0.00	328.5	10	8.63	3600	0
r	P25-S3	3.21	1342.8	6	16.51	3600	0
r	P15-S4	0.00	72.1	10	2.47	3285.4	4
r	P20-S4	0.00	512.6	10	11.38	3600	0
r	P25-S4	4.34	1896.4	4	21.07	3600	0
rc	P15-S3	0.00	41.3	10	0.87	2745.6	7
rc	P20-S3	0.00	272.1	10	7.25	3600	1
rc	P25-S3	2.16	1108.5	7	14.83	3600	0
rc	P15-S4	0.00	56.8	10	1.62	3124.8	5
rc	P20-S4	0.00	402.4	10	9.74	3600	0
rc	P25-S4	3.25	1587.2	5	18.26	3600	0
Average		0.91	523.7	8.8	8.19	3401.5	2.5

to optimality only for clustered layouts, and fails to find optimal solutions for most P20 instances, leaving average gaps of 3.8–11.3%. The difference is most striking on P25 instances, where BPC achieves gaps of 1.4–4.3% and still solves the majority to optimality, while Gurobi consistently exceeds 9% and solves none within the time limit. These results confirm the computational superiority of the route-based formulation with the proposed BPC algorithm over the compact arc-based formulation solved by Gurobi. Due to space limitations, detailed analyses of the impact of individual cut families, robustness parameter sensitivity, and the role of crowdsourced couriers are omitted here. Our experiments further reveal that the three families of valid inequalities collectively yield substantial reductions in both the root node gap and overall solution time, the cost of robustness increases approximately linearly with Γ^2, and that integrating crowdsourced couriers meaningfully reduces total system cost, with greater benefits observed under moderate robustness.

Acknowledgements

The work described in this paper was partially supported by the National Natural Science Foundation of China (No. 72101222, 72371216) and the Research Grants Council of Hong Kong (No. 27202221, 17205124, T32-707-22).

References

Bertsimas, D., & Sim, M. (2004). The price of robustness. *Operations research*, *52*(1), 35–53.

Li, D., Sun, W., & Zhang, F. (2026). Order allocation and vehicle routing with collaborative pickup and delivery by crowdsourced and contracted couriers in a two-echelon urban logistics system. *Transportation Research Part E: Logistics and Transportation Review*, *207*, 104598.

Munari, P., Moreno, A., De La Vega, J., Alem, D., Gondzio, J., & Morabito, R. (2019). The robust vehicle routing problem with time windows: Compact formulation and branch-price-and-cut method. *Transportation Science*, *53*(4), 1043–1066.