Robust Service Network Design under Travel Time Uncertainty: Formulations and Exact Solutions

Shengnan Shu
Department of Logistics and Maritime Studies, Hong Kong Polytechnic University, shengnan.shu@connect.polyu.hk

Zhou Xu
Department of Logistics and Maritime Studies, Hong Kong Polytechnic University, lgtzx@polyu.edu.hk

Jin Qi
Department of Industrial Engineering and Decision Analytics, Hong Kong University of Science and Technology, jinqi@ust.hk

We study the continuous-time service network design problem (CTSNDP) under travel time uncertainty, aiming to design a transportation service network along a continuous-time planning horizon, with robust operational efficiency even in the presence of travel time deviations. Incorporating travel time uncertainty holds a great practical value. However, it poses a significant challenge in both problem formulation and solution computation, as the time-indexed mixed-integer linear programming (MILP) formulations commonly used to solve the CTSNDP with deterministic travel times become impractical. To tackle this challenge, we propose a new consolidation-indexed MILP formulation for the deterministic CTSNDP, which enables us to derive computationally tractable formulations for a robust optimization model and a robust satisficing model. Both of these robust models provide solutions that can mitigate the impact of travel time uncertainty, without knowledge of the precise joint probability distribution of travel times. To compute the exact optimal solutions for these models, we develop two tailored column-and-constraint generation algorithms. This particularly marks the first success of such algorithms in solving a two-stage robust satisficing model, featuring a novel enhanced bisection search procedure for a challenging max-min fractional optimization problem. Our computational results demonstrate the effectiveness of these algorithms, the tractability of the proposed formulations, as well as the trade-offs involved in achieving the robust solutions.

Key words: service network design; continuous time; travel time uncertainty; transportation; robust optimization; robust satisficing; exact algorithm; column-and-constraint generation

1. Introduction

In the transportation industry, a significant portion of the freight is moved by consolidation carriers, including railroads, container shipping lines, less-than-truckload motor carriers, and regular and express postal service providers. These consolidation carriers transport shipments that are small compared with their vehicles' capacities. As a result, they need to consolidate their shipments to achieve cost-effectiveness, which poses a service network design problem (SNDP).

The SNDP involves routing of shipments from their origins to destinations through a network of terminals, where shipments can be transferred from inbound vehicles to outbound vehicles. Each shipment has an available time for departure from its origin and a due time for arrival at its destination. To transport shipments between terminals, vehicles with limited capacities are needed. At each terminal, when multiple shipments are consolidated, the outbound vehicle carrying these shipments cannot be dispatched until all the inbound vehicles bringing these shipments have arrived. Accordingly, a classic SNDP seeks to determine routing and consolidation plans of the shipments, and numbers and dispatch times of the vehicles on each terminal-to-terminal movement, so that the shipment available times and due times are met, with the total operational cost minimized.

The classic SNDP and other variants of the SNDP have been extensively studied, due to their wide applications and theoretical significance (Wieberneit 2008). However, the existing studies primarily focus on deterministic variants, assuming that all problem parameters, such as shipment quantities and travel times, are known in advance. To model the *deterministic* variants, a widely used technique is *discretization*, which discretizes the planning horizon into a number of time points (see, e.g., Crainic et al. 2016). Using these time points, a deterministic SNDP can be modeled on a time-expanded network, in which each time-space node represents a combination of a time point and a terminal, and each arc connecting two time points represents a shipment's movement between terminals or its waiting at a terminal during a specific period.

The time-expanded network constructed from the discretization can be effectively used to incorporate decisions of the SNDP. Based on this, a mixed-integer linear programming (MILP) formulation can be established. It is *time-indexed*, since the decision variables involved are indexed by a pair of time points of the time-space nodes. The time-indexed MILP formulation can be solved by commercial optimization solvers and received significant attention (Wieberneit 2008).

However, the time-indexed MILP formulation is only an approximation of the SNDP, where the planning horizon is continuous and vehicles can be dispatched at any time. Achieving high-quality solutions requires a fine discretization, which often leads to a large and typically intractable time-indexed MILP formulation of the SNDP. To solve the continuous-time variant of the SNDP (or *CTSNDP* in short), Boland et al. (2017) proposed a Dynamic Discretization Discovery (DDD) algorithm, which iteratively adjusts the discretization level until reaching an optimal solution. This exact algorithm was later enhanced by Hewitt (2019), Marshall et al. (2021), and Shu et al. (2024).

The formulations and solution algorithms introduced above are applicable to only deterministic variants of the SNDP. In contrast, this paper focuses on the development of an exact solution algorithm for a new variant of the CTSNDP that incorporates travel time uncertainty. It is an under-explored and challenging task that holds significant practical value.

1.1. Significance and Challenges of Incorporating Travel Time Uncertainty

Actual travel times for transportation services are often unknown in advance due to factors such as weather conditions and traffic congestion, leading to delays and late deliveries. This can result in disruptive impacts, such as penalties and customer dissatisfaction (Lanza et al. 2021). Bertsimas et al. (2019) has highlighted the significant importance of travel time estimation. A considerable amount of literature has investigated travel time uncertainty in different applications, including traffic assignment and vehicle routing (Nikolova and Stier-Moses 2014, Jaillet et al. 2016).

Despite its significance, travel time uncertainty has rarely been considered for transportation service network design. Existing studies incorporating uncertainty in SNDP mainly focus on demand uncertainty, either using a stochastic optimization approach to optimize the average performance (Wang et al. 2019, Lium et al. 2009), or a robust optimization approach (Wang and Qi 2020) to optimize the worst-case performance. When dispatch times are involved, existing solution methods all use the time-indexed formulation to incorporate uncertain demands. This is achievable because, even with demand uncertainty, all decision variables retain the same time indices as those in the time-indexed formulation of the deterministic SNDP. In communication network design, the robust optimization approach has also been employed to address demand and cost uncertainties (Koster et al. 2013, Pessoa and Poss 2015, Altın et al. 2011), where time-related decisions are not involved.

Unlike uncertain demands, incorporating uncertain travel times poses a significant challenge, as the time-indexed formulation of the deterministic SNDP becomes impractical. Decisions involving travel times, such as actual departure times of vehicles, require time-indexed decision variables with varying indices for different realizations of uncertain travel times. This significantly increases the size and complexity of the optimization model.

We study the CTSNDP under travel time uncertainty over a continuous-time planning horizon. Due to the impracticality of the time-indexed formulation, exact solution methods for the deterministic CTSNDP, such as the DDD algorithm, cannot be applied. Compared with other transportation problems, such as traffic assignment and vehicle routing, incorporating travel time uncertainty in the CTSNDP poses another challenge. When a shipment has a delay, it can trigger delays for other shipments awaiting consolidation at a terminal, consequently causing delays at other terminals. Such a delay propagation caused by consolidation greatly complicates the problem.

1.2. Related Work

Among the limited studies that consider travel time uncertainty in SNDP, the delay propagation caused by consolidation has been overlooked, except for Demir et al. (2016), Hrušovskỳ et al. (2018), Layeb et al. (2018), which focus on a restricted SNDP where services can only be selected from a small candidate set. These three studies adopt a stochastic optimization approach to incorporate travel time uncertainty and optimize the average performance. However, it requires to know the joint probability distribution of all the travel times, which is often not possible. Due to the large number of decision variables and the multitude of possible realizations of uncertain factors, solving the stochastic optimization model derived from this approach to optimality is challenging. These three studies apply only heuristic methods based on simulations or limited travel time samples.

Our paper is the first to adopt a robust optimization approach to study the CTSNDP under travel time uncertainty. *Our goal is to design a transportation service network that maintains reliable operational efficiency, even in the presence of travel time deviations and without knowledge of the actual travel time distribution.* The robust optimization approach only requires a distribution-free uncertainty set that defines the possible realizations of uncertain factors (Bertsimas et al. 2011). It relaxes the need for precise distribution information, and often leads to an optimization formulation with a tractable reformulation that can be efficiently solved. In the classic robust optimization approach, the objective is to optimize the worst-case objective value of a solution over different realizations of uncertain factors.

A recent study by Long et al. (2023) proposes a robust satisficing approach, aiming to find a solution that best achieves a prescribed target of the objective value, with the worst-case normalized violation from the target minimized. Unlike the traditional robust optimization which restricts the possible realizations of uncertain factors within a pre-specified uncertainty set, the robust satisficing framework allows the nature to choose the realization in the whole space. It has also been demonstrated through several applications to have the advantage of improving out-of-sample performance over the classic robust optimization approach (Zhou et al. 2022, Cui et al. 2023).

To apply the robust optimization and robust satisficing approaches, we need to incorporate travel time uncertainty into the CTSNDP formulation. As discussed in Section 1.1, this is a challenging task, since the time-indexed formulations commonly used for the deterministic CTSNDP become impractical. Moreover, the MILP formulations presented in Demir et al. (2016) and Hrušovskỳ et al. (2018) for the SNDP consist of decision variables with service indices, which are essentially time-indexed, as services are defined by their departure and arrival times. Hewitt and Lehuédé (2023) propose an MILP formulation for the deterministic CTSNDP without time indices. However, its total number of decisions and constraints is proportional to the total number of possible shipment combinations, which can be exponentially large, making it very challenging to solve.

1.3. Our Contributions

This study tackles the challenge of formulating and solving the robust CTSNDP under travel time uncertainty, without knowledge of the travel time distribution. We first propose *a new consolidation-indexed MILP formulation for the deterministic CTSNDP*, eliminating the need for time indices. This enables us to derive *tractable MILP formulations for both a robust optimization model and a robust satisficing model of the robust CTSNDP*, which are based on distribution-free polyhedral uncertainty sets and involve two stages of optimization. These two robust modeling frameworks enable decision makers to better align their preferences regarding the robustness guarantees of the solutions obtained and the trade-offs involved. It is worthy noting that our formulations for robust models can also be used to derive MILP formulations for other optimization models, including stochastic programming, to incorporate travel time uncertainty.

To solve the robust optimization model and the robust satisficing model, we develop two exact algorithms, respectively. They both follow a column-and-constraint generation (C&CG) framework, which has widely been applied to two-stage robust optimization models (see, e.g., Zeng and Zhao 2013, Wang and Qi 2020), but has never been applied to two-stage robust satisficing models before. The critical step in our C&CG algorithms is its solution to a subproblem that finds the worst-case realized travel times for any given first-stage solution, for which we need to analyze and utilize *optimality properties* of the subproblem. For our robust optimization model, this subproblem involves max-min optimization where the inner minimization is a linear program (LP). We can reformulate it as an MILP and solve it by an optimization solver. For our robust satisficing model, the subproblem is much more complicated, as it involves *max-min fractional optimization*, requiring us to develop *a novel enhanced bisection search procedure*. Through extensive computational experiments, we demonstrate the tractability of our proposed models, the effectiveness of our developed algorithms, and the trade-off involved in achieving the robustness guarantees.

In what follows, Section 2 introduces problem statements for both the deterministic CTSNDP and the robust CTSNDP under travel time uncertainty. Section 3 presents our new MILP formulation for the deterministic CTSNDP, and its extensions to the robust optimization model and the robust satisficing model of the robust CTSNDP. Section 4 explains and analyzes our C&CG algorithms. The computational results are discussed in Section 5, followed by a conclusion in Section 6. All notation is summarized in Table A.1 of Appendix A and all proofs are presented in Appendix B.

2. Problem Statements

In this section, we first introduce the deterministic CTSNDP where travel times are given, and then define two variants of the robust CTSNDP where travel times are uncertain.

2.1. Deterministic CTSNDP

The deterministic CTSNDP examined in this paper extends the problem setting in Boland et al. (2017), with shipment holding costs being incorporated. Unlike Boland et al. (2017), we define its feasible solutions over the physical network of the terminals, instead of the time-expanded network.

Consider a network $\mathcal{D} = (\mathcal{N}, \mathcal{A})$ with a physical node set $\mathcal{N}$ and a directed arc set $\mathcal{A}$, which is referred to as the *flat network*. Each physical node represents a terminal, and each arc represents a direct transport service from one terminal to another. Consider a commodity set $\mathcal{K}$, where each commodity $k \in \mathcal{K}$ represents a shipment, with its origin denoted by $o^k \in \mathcal{N}$, its destination denoted by $d^k \in \mathcal{N}$, and its shipping quantity denoted by $q^k \in \mathbb{N}_{>0}$. Each commodity $k \in \mathcal{K}$ has an earliest available time $e^k \in \mathbb{N}$ for departure from its origin o^k, and has a due time $l^k \in \mathbb{N}_{>0}$ for arrival at its destination d^k. No commodity can be delivered separately, and thus each commodity can only be assigned exactly one delivery path. As a result, each commodity must be picked up exactly once from the origin after the earliest available time and delivered exactly once to the destination before the due time. However, commodities can be temporarily stored at any nodes, waiting to be consolidated for shipping together on different arcs of the network.

In the flat network $\mathcal{D}$, each arc $(i,j) \in \mathcal{A}$ is associated with four attributes: (1) travel time $\tau_{ij} \in \mathbb{N}_{>0}$; (2) a per-unit-of-flow (travel) cost $c_{ij}^k \in \mathbb{R}_{>0}$ for each commodity $k \in \mathcal{K}$; (3) a fixed cost $f_{ij} \in \mathbb{R}_{>0}$ per vehicle for (shipping) service on the arc; and (4) a capacity $u_{ij} \in \mathbb{N}_{>0}$ per vehicle for (shipping) service on the arc. Additionally, both *in-transit* and *in-storage* holding costs are considered here for each commodity. In particular, the in-transit holding costs are incorporated into the flow costs c_{ij}^k for commodities $k \in \mathcal{K}$ and arcs $(i,j) \in \mathcal{A}$. A per-unit-of-demand-and-time in-storage holding cost $h^k \in \mathbb{R}_{\geq 0}$ is incurred when a commodity $k \in \mathcal{K}$ is stored at any node.

The deterministic CTSNDP requires the deciding of delivery paths and consolidation plans for all commodities, as well as the numbers and the dispatch times of the required vehicles. Its objective is to satisfy all delivery requirements while keeping the total cost minimized.

A *feasible solution* to the deterministic CTSNDP consists of (i) a routing plan, (ii) a consolidation plan, and (iii) a departure schedule, these being defined as follows (see the illustration in Figure 2 of Appendix A). We call a directed path P in the flat network $\mathcal{D}$ a *flat path*, which is represented by its node sequence $(\nu_1, \nu_2, \ldots, \nu_{m+1})$ and arc sequence $(a_1, a_2, \ldots, a_m)$, with $m \in \mathbb{N}_{>0}$ denoting the total number of its arcs. As in actual practice, the delivery path of each commodity cannot have repeated vertices or arcs, and thus must be an elementary flat path from the origin to the destination of the commodity. Accordingly, a *routing plan* $\mathcal{P}$ is defined as a collection of $|\mathcal{K}|$ elementary flat paths in the flat network $\mathcal{D}$, with each flat path $P^k \in \mathcal{P}$ for $k \in \mathcal{K}$ representing the delivery path of commodity k from its origin o^k to destination d^k, where the node and arc sequences of P^k are denoted by $(\nu_1^k, \nu_2^k, \ldots, \nu_{m^k+1}^k)$ and $(a_1^k, a_2^k, \ldots, a_{m^k}^k)$, respectively, with $\nu_1^k = o^k$ and $\nu_{m^k+1}^k = d^k$, and with no repeated nodes or arcs.

Given a routing plan $\mathcal{P}$, we need to specify how commodities are consolidated for each arc $\alpha \in \mathcal{A}$ of the flat network $\mathcal{D}$. Let $\mathcal{K}(\mathcal{P}, \alpha) = \{k \in \mathcal{K} \mid \exists a_n^k = \alpha, 1 \leq n \leq m^k\}$ indicate a subset of commodities whose flat paths in $\mathcal{P}$ pass through arc α. A consolidation on α for $\mathcal{P}$ can be represented by a subset of $\mathcal{K}(\mathcal{P}, \alpha)$, so that commodities in this subset are consolidated to be shipped together on α. Since each commodity cannot be split during transportation, there are at most $|\mathcal{K}|$ consolidations on each arc α. We use $C_r^\alpha \subseteq \mathcal{K}(\mathcal{P},\alpha)$ to indicate the r-th consolidation on α for the routing plan $\mathcal{P}$, so that all the commodities in C_r^α are shipped through α together. A *consolidation plan* $\mathcal{C}$ for $\mathcal{P}$ can thus be defined as a collection of consolidations $C_r^\alpha \subseteq \mathcal{K}(\mathcal{P},\alpha)$ for $\alpha \in \mathcal{A}$ and $r \in \{1,2,\dots,|\mathcal{K}|\}$, where r is referred to as the *consolidation index*. If the consolidations C_r^α for $r = 1,2,\dots,|\mathcal{K}|$ cover all the commodities $k \in \mathcal{K}(\mathcal{P},\alpha)$ for each arc $\alpha \in \mathcal{A}$, i.e., $\bigcup_{r=1}^{|\mathcal{K}|} C_r^\alpha = \mathcal{K}(\mathcal{P},\alpha)$ is satisfied for each $\alpha \in \mathcal{A}$, then such a routing-consolidation pair $(\mathcal{P},\mathcal{C})$ forms a *flat solution* to the deterministic CTSNDP.

Given a flat solution $(\mathcal{P},\mathcal{C})$, we need to further specify the departure time of each commodity from every node it passes through. Since each flat path in $\mathcal{P}$ is an elementary path, every commodity can depart from the same node at most once. Accordingly, a *departure schedule* $\mathcal{T}$ is defined as a collection of departure times $t^k_{\nu^k_n}$ for $k \in \mathcal{K}$ and $n \in \{1,2,\dots,m^k\}$, indicating when commodity k departs from node ν^k_n via arc a^k_n of its flat path P^k. Thus, $(\mathcal{P},\mathcal{C},\mathcal{T})$ forms a *feasible solution* to the deterministic CTSNDP if the departure schedule $\mathcal{T}$ satisfies that

$$t^k_{\nu^k_n} \geq e^k, \quad \text{for } n = 1, \tag{2.1}$$

$$t^k_{\nu^k_{n+1}} \geq t^k_{\nu^k_n} + \tau_{a^k_n}, \ \text{for } n \in \{1,2,\dots,m^k-1\}, \tag{2.2}$$

$$t^k_{\nu^k_n} + \tau_{a^k_n} \leq l^k, \ \text{for } n = m^k, \tag{2.3}$$

$$t^k_i = t^{k'}_i, \quad \text{for } k \in C_r^{(i,j)} \text{ and } k' \in C_r^{(i,j)} \text{ with } (i,j) \in \mathcal{A} \text{ and } r \in \{1,2,\cdots,|\mathcal{K}|\}. \tag{2.4}$$

Here, for each commodity $k \in \mathcal{K}$, constraints (2.1) and (2.3) together ensure that the departure time from its origin and arrival time at its destination are both within the time window $[e^k, l^k]$, and constraints (2.2) are due to the travel times of arcs on its flat path. Constraints (2.4) ensure that commodities consolidated on the same arc all pass the arc at the same time. A flat solution $(\mathcal{P},\mathcal{C})$ is *timely-implementable*, if there exists such a departure schedule $\mathcal{T}$ that satisfies (2.1)–(2.4).

From a feasible solution $(\mathcal{P},\mathcal{C},\mathcal{T})$, we can obtain holding times H^k_n for nodes ν^k_n with $n = 1,2,\dots,m^k+1$ on the flat path P^k of each commodity $k \in \mathcal{K}$, with $H^k_1 = t^k_{\nu^k_1} - e^k$, $H^k_n = t^k_{\nu^k_n} - (t^k_{\nu^k_{n-1}} + \tau_{a^k_{n-1}})$ for $n \in \{2,\dots,m^k\}$, and $H^k_{m^k+1} = l^k - (t^k_{\nu^k_{m^k}} + \tau_{a^k_{m^k}})$. Accordingly, we can define $h(\mathcal{P},\mathcal{T})$ to represent the total holding cost, where $h(\mathcal{P},\mathcal{T}) = \sum_{k\in\mathcal{K}} \sum_{n=1}^{m^k+1} h^k q^k H^k_n$, and define $f(\mathcal{P},\mathcal{C})$ to represent the total fixed cost and flow cost, where $f(\mathcal{P},\mathcal{C}) = \sum_{\alpha\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} f_\alpha \lceil \sum_{k\in C_r^\alpha} q^k/u_\alpha \rceil + \sum_{k\in\mathcal{K}} \sum_{n=1}^{m^k} c^k_{a^k_n} q^k$. Thus, the total cost of solution $(\mathcal{P},\mathcal{C},\mathcal{T})$ equals $f(\mathcal{P},\mathcal{C}) + h(\mathcal{P},\mathcal{T})$.

Without loss of generality, we assume that for each commodity $k \in \mathcal{K}$, the difference $(l^k - e^k)$ between its latest arrival time l^k at the destination and available time e^k at the origin is not smaller than the length of the shortest-time path from o^k to d^k in the flat network $\mathcal{D}$. This is sufficient to ensure the existence of a feasible solution to the deterministic CTSNDP. The deterministic CTSNDP can thus be formulated as follows, where $\mathbb{S}$ indicates the domain of all the feasible solutions.

$$\text{(Deterministic CTSNDP)} \quad \min_{(\mathcal{P},\mathcal{C},\mathcal{T})\in\mathbb{S}} [f(\mathcal{P},\mathcal{C}) + h(\mathcal{P},\mathcal{T})].$$

2.2. Robust CTSNDP

We now introduce the robust CTSNDP under travel time uncertainty, which we refer to as the robust CTSNDP for short. Suppose that for each arc $\alpha \in \mathcal{A}$, the actual travel time $\tilde{\tau}_\alpha$ for commodities passing through α is determined by $\tilde{\tau}_\alpha = \overline{\tau}_\alpha + \hat{\tau}_\alpha \delta_\alpha$. Here, $\overline{\tau}_\alpha \in \mathbb{N}_{>0}$ is the *nominal* value of $\tilde{\tau}_\alpha$, and $\hat{\tau}_\alpha \in \mathbb{N}$ with $\hat{\tau}_\alpha < \overline{\tau}_\alpha$ is the maximum deviation of $\tilde{\tau}_\alpha$ with respect to the nominal value $\overline{\tau}_\alpha$. The coefficient δ_α is a random variable (but with unknown distribution), and its value falls within the range $[-1,1]$. Thus, $\tilde{\tau}_\alpha$ falls within the range $[\overline{\tau}_\alpha - \hat{\tau}_\alpha, \overline{\tau}_\alpha + \hat{\tau}_\alpha]$.

For each $\alpha \in \mathcal{A}$, since there can be at most $|\mathcal{K}|$ consolidations on arc α, we use $\tilde{\tau}_{\alpha r}$ for $r \in \{1,2,\ldots,|\mathcal{K}|\}$ to indicate the travel time of the r-th consolidation on arc α. Let $\mathbb{U}$ indicate the support of the vector $\boldsymbol{\delta}$ of random variables $\delta_{\alpha r}$ for $\alpha \in \mathcal{A}$ and $r \in \{1,2,...,|\mathcal{K}|\}$. We have that

$$\mathbb{U} = \Big\{ \boldsymbol{\delta} : \delta_{\alpha r} \in [-1,1], \forall \alpha \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\} \Big\}. \tag{2.5}$$

For each realized coefficient value $\boldsymbol{\delta} \in \mathbb{U}$, we use $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ to indicate the vector of the corresponding realized travel times $(\overline{\tau}_\alpha + \hat{\tau}_\alpha \delta_{\alpha r})$ for $\alpha \in \mathcal{A}$ and $r \in \{1,2,...,|\mathcal{K}|\}$, which can be defined as

$$\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}) = \Big\{ \tilde{\boldsymbol{\tau}} : \tilde{\tau}_{\alpha r} = \overline{\tau}_\alpha + \hat{\tau}_\alpha \delta_{\alpha r}, \forall \alpha \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\} \Big\}. \tag{2.6}$$

We refer to vector $\boldsymbol{\delta}$ as *scenario*, and refer to $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ as the realized travel time for $\boldsymbol{\delta}$.

Similar to the stochastic SNDP studied in Lanza et al. (2021), the decision process for the robust CTSNDP has two stages. In the first stage, before actual values of the travel times are realized, the problem needs to determine a routing plan $\mathcal{P}$ and a consolidation plan $\mathcal{C}$ that form a flat solution $(\mathcal{P},\mathcal{C})$. As $\mathcal{P}$ and $\mathcal{C}$ are crucial to determine resources for transportation, such as vehicles, which require time for preparation, they need to be independent of the realized travel times. Given $(\mathcal{P},\mathcal{C})$, in the second stage, which is after the actual values of the travel times are realized, the problem needs to determine an actual departure schedule $\mathcal{T}$, which can adapt to the realized travel times.

Due to travel time uncertainty, it is costly to satisfy the due time constraints for every possible realization of the travel times. Accordingly, in the second stage of the robust CTSNDP, we relax the due time constraints and impose a penalty g^k per unit of time for the delay of each commodity k's arrival at its destination. Thus, given a flat solution $(\mathcal{P},\mathcal{C})$, and after actual travel times $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ with $\boldsymbol{\delta} \in \mathbb{U}$ are realized, we need to compute the holding costs and delay penalty costs incurred in the second stage. For this, we need to determine an optimal departure schedule $\mathcal{T} = (t^k_{\nu^k_n})_{k \in \mathcal{K}, 1 \le n \le m^k}$, where each $t^k_{\nu^k_n}$ denotes the departure time of commodity k from node ν^k_n on the flat path P^k of $\mathcal{P}$. Specifically, for each commodity $k \in \mathcal{K}$ and arc $a^k_n = (\nu^k_n, \nu^k_{n+1})$ of P^k, since $(\mathcal{P},\mathcal{C})$ is a flat solution, there must exist $r(k,n) \in \{1,2,\ldots,|\mathcal{K}|\}$ such that the commodity k is contained in the consolidation $C^{a^k_n}_{r(k,n)}$ of $\mathcal{C}$. Thus, the actual travel time of commodity k on arc a^k_n equals $\tilde{\tau}_{a^k_n, r(k,n)}$. Accordingly, the actual departure schedule $\mathcal{T}$ needs to satisfy constraints (2.1), (2.4), and (2.7) below:

$$t^k_{\nu^k_{n+1}} \ge t^k_{\nu^k_n} + \tilde{\tau}_{a^k_n, r(k,n)}, \text{ for } k \in \mathcal{K}, \; n \in \{1,2,\ldots,m^k-1\}. \tag{2.7}$$

Constraints (2.7) are similar to (2.2) with $\tau_{a_n^k}$ replaced by $\tilde{\tau}_{a_n^k, r(k,n)}$. Thus, the domain of such actual departure schedules $\mathcal{T}$ is denoted by $\mathbb{T}(\mathcal{P}, \mathcal{C}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$. Let $g(\mathcal{P}, \mathcal{T}) = \sum_{k\in\mathcal{K}} g^k \cdot \max\{t^k_{\nu^k_{m^k}} + \tilde{\tau}_{a^k_{m^k}, r(k,m^k)} - l^k, 0\}$, indicating the total delay penalty for an actual departure schedule $\mathcal{T}$ with respect to flat paths in $\mathcal{P}$. Hence, under the realized travel times $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ with $\boldsymbol{\delta} \in \mathbb{U}$, the corresponding second-stage cost, including the holding costs and delay penalties, equals $h(\mathcal{P}, \mathcal{T}) + g(\mathcal{P}, \mathcal{T})$. Its minimum value, $\min_{\mathcal{T}\in\mathbb{T}(\mathcal{P},\mathcal{C},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))} [h(\mathcal{P}, \mathcal{T}) + g(\mathcal{P}, \mathcal{T})]\}$, is referred to as the second-stage cost under scenario $\boldsymbol{\delta}$.

For the first stage of the robust CTSNDP, a flat solution $(\mathcal{P}, \mathcal{C})$ needs to be determined before the actual travel times are realized. Following the *light robustness* approach, proposed by Fischetti and Monaci (2009), we require that the flat solution $(\mathcal{P}, \mathcal{C})$ to be determined in the first stage needs to form a feasible solution to the deterministic CTSNDP under a nominal scenario, where travel times take their nominal values with no deviations. This is also commonly required in practice. Accordingly, such a flat solution $(\mathcal{P}, \mathcal{C})$ must be timely-implementable under the nominal scenario, and we refer to it as a *nominal timely-implementable first-stage solution.* We use $\mathbb{F}$ to indicate the domain of all nominal timely-implementable first-stage solutions.

The robust CTSNDP aims to find a robust nominal timely-implementable first-stage solution under the travel time uncertainty. Below, we adopt two modeling frameworks, namely robust optimization and robust sacrificing, to characterize the robustness of such solutions.

Remark 2.1 (Uncertainty Revelation) *The second stage of the robust CTSNDP introduced above assumes that actual travel times are revealed before the departure schedule is determined. This aligns with the literature (see, e.g., Atamtürk and Zhang 2007, Yanıkoğlu et al. 2019). However, in some situations, the actual travel time for transporting a shipment through an arc cannot be determined until the transportation is completed. Such a mechanism is referred to as the dynamic uncertainty revelation. Appendix C shows that solutions to our two-stage formulation of the robust CTSNDP can be adapted for the dynamic uncertainty revelation, without increasing objective values.*

2.2.1. Robust Optimization Variant of CTSNDP Given an integer $\Gamma \in \mathbb{N}$, known as the *budget of uncertainty*, we can use it to adjust the level of robustness as needed, by introducing the following budgeted uncertainty set $\mathbb{U}(\Gamma)$ of scenarios $\boldsymbol{\delta}$:

$$\mathbb{U}(\Gamma) = \Big\{\boldsymbol{\delta} : \|\boldsymbol{\delta}\|_1 \le \Gamma, \delta_{\alpha r} \in [-1, 1], \forall \alpha \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}\Big\}, \tag{2.8}$$

where $\|\boldsymbol{\delta}\|_1 = \sum_{\alpha\in\mathcal{A}, r\in\{1,2,...,|\mathcal{K}|\}} |\delta_{\alpha r}|$. It contains all possible $\boldsymbol{\delta}$ such that $\|\boldsymbol{\delta}\|_1$, the total relative deviation of the travel times $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ from their nominal values, does not exceed Γ.

The robust optimization variant of the CTSNDP under travel time uncertainty (or RO-CTSNDP in short) has an objective to minimize the worst-case total two-stage cost with respect to the budgeted uncertainty set $\mathbb{U}(\Gamma)$ on $\boldsymbol{\delta}$. To achieve this, the RO-CTSNDP needs to determine a nominal timely-implementable first-stage solution $(\mathcal{P},\mathcal{C})\in\mathbb{F}$ that minimizes the sum of the first-stage cost (which is independent of the realization of $\boldsymbol{\delta}$) and the worst-case second-stage cost (which is over the budgeted uncertainty set $\mathbb{U}(\Gamma)$ on $\boldsymbol{\delta}$). Accordingly, the RO-CTSNDP can be formulated as follows:

$$[\text{RO-CTSNDP}]\quad \min_{(\mathcal{P},\mathcal{C})\in\mathbb{F}} \{f(\mathcal{P},\mathcal{C}) + \max_{\boldsymbol{\delta}\in\mathbb{U}(\Gamma)} \min_{\mathcal{T}\in\mathbb{T}(\mathcal{P},\mathcal{C},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))} [h(\mathcal{P},\mathcal{T})+g(\mathcal{P},\mathcal{T})]\}.$$

2.2.2. Robust Satisficing Variant of CTSNDP We follow the modeling framework of Long et al. (2023) to establish the robust satisficing variant of the CTSNDP under travel time uncertainty (or RS-CTSNDP in short). Let $\mathcal{Z}_0$ represent the optimal objective value of the deterministic CTSNDP under nominal travel times. Given a prescribed target $\mathcal{Z}$ of the total two-stage cost with $\mathcal{Z}\geq\mathcal{Z}_0$, the RS-CTSNDP needs to determine a nominal timely-implementable first-stage solution $(\mathcal{P},\mathcal{C})\in\mathbb{F}$ that best achieves the prescribed target $\mathcal{Z}$, so that the worst-case normalized magnitude of the deviation from the target is minimized. Accordingly, the RS-CTSNDP can be formulated as

$$\min_{(\mathcal{P},\mathcal{C})\in\mathbb{F}} \{\rho\in\mathbb{R}_{\geq 0} : f(\mathcal{P},\mathcal{C}) + \min_{\mathcal{T}\in\mathbb{T}(\mathcal{P},\mathcal{C},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))} [h(\mathcal{P},\mathcal{T})+g(\mathcal{P},\mathcal{T})] - \mathcal{Z} \leq \rho\|\boldsymbol{\delta}\|_1,\ \forall \boldsymbol{\delta}\in\mathbb{U}\}.$$

Here, the constraints imposed on the first-stage solution $(\mathcal{P},\mathcal{C})\in\mathbb{F}$ restrict the deviation of the total two-stage cost from the prescribed target $\mathcal{Z}$ to not exceed $\rho\|\boldsymbol{\delta}\|_1$ for every possible scenario $\boldsymbol{\delta}$ in the uncertainty set $\mathbb{U}$. As a result, ρ indicates the worst-case magnitude of the deviation from the prescribed cost target, normalized by the total relative deviation $\|\boldsymbol{\delta}\|_1$ of the travel times. This quantity measures the fragility of a given solution and needs to be minimized to attain robustness.

3. Optimization Models

We first propose a new compact MILP model for the deterministic CTSNDP using consolidation indices instead of time indices. Based on this, we then derive two-stage mixed-integer nonlinear programming (MINLP) models for the two variants of the robust CTSNDP (RO-CTSNDP and RS-CTSNDP), with their second-stage costs for each scenario formulated as linear programs (LP).

3.1. Consolidation-Indexed MILP Model for Deterministic CTSNDP

As described in Section 2, a feasible solution to the deterministic CTSNDP consists of a routing plan $\mathcal{P}$, a consolidation plan $\mathcal{C}$, and a departure schedule $\mathcal{T}$. (i) To represent the routing plan $\mathcal{P}$, we introduce a binary variable x_{ij}^k for each $(i,j)\in\mathcal{A}$ and $k\in\mathcal{K}$, indicating whether commodity $k\in\mathcal{K}$ passes through arc (i,j). (ii) To represent the consolidation plan $\mathcal{C}$, we first introduce a binary variable z_{ijr}^k for each $(i,j)\in\mathcal{A}$, $r\in\{1,2,\cdots,|\mathcal{K}|\}$, and $k\in\mathcal{K}$, indicating whether the r-th consolidation $C_r^{(i,j)}$ on arc (i,j) contains commodity k. We then introduce a non-negative integer variable y_{ijr} for each $(i,j)\in\mathcal{A}$ and $r\in\{1,2,\cdots,|\mathcal{K}|\}$, indicating the number of vehicles needed by consolidation $C_r^{(i,j)}$ of arc (i,j) to accommodate the commodities in consolidation $C_r^{(i,j)}$. (iii) To represent the departure schedule $\mathcal{T}$, we first introduce a non-negative continuous variable v_{ij}^k for each $(i,j) \in \mathcal{A}$ and $k \in \mathcal{K}$, which indicates the time when commodity k departs from node i when passing through arc (i,j). If commodity k does not pass through arc (i,j), then v_{ij}^k equals 0. We introduce a non-negative continuous variable b_{ijr} for each $(i,j) \in \mathcal{A}$ and $r \in \{1,2,...,|\mathcal{K}|\}$, which represents the time when commodities of the r-th consolidation $C_r^{(i,j)}$ on arc (i,j) depart from node i. (iv) We also introduce a non-negative continuous variable w_i^k for $i \in \mathcal{N}$ and $k \in \mathcal{K}$ to represent the holding time for commodity k at terminal i. It equals 0 if commodity k does not pass node i.

Accordingly, the deterministic CTSNDP can be represented by the following compact MILP model, referred to as model DO, where M denotes a sufficiently large constant:

$$\text{[DO]} \ \min \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} f_{ij} \cdot y_{ijr} + \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} (c_{ij}^k q^k) \cdot x_{ij}^k + \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{N}} (h^k q^k) \cdot w_i^k \tag{3.1}$$

$$\text{s.t.} \ \sum_{(i,j)\in\mathcal{A}} x_{ij}^k - \sum_{(j,i)\in\mathcal{A}} x_{ji}^k = \begin{cases} 1, & i = o^k, \\ -1, & i = d^k, \\ 0, & \text{otherwise}, \end{cases} \qquad \forall k \in \mathcal{K}, i \in \mathcal{N}, \tag{3.2}$$

$$\sum_{k\in\mathcal{K}} q^k z_{ijr}^k \le u_{ij} y_{ijr}, \qquad \forall\ (i,j) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{3.3}$$

$$\sum_{k\in\mathcal{K}} q^k z_{ijr}^k \ge u_{ij} y_{ijr} - u_{ij} + 1, \qquad \forall\ (i,j) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{3.4}$$

$$\sum_{r=1}^{|\mathcal{K}|} z_{ijr}^k = x_{ij}^k, \qquad \forall\ (i,j) \in \mathcal{A}, k \in \mathcal{K}, \tag{3.5}$$

$$\sum_{j:(j,i)\in\mathcal{A}} (v_{ji}^k + \tau_{ji} x_{ji}^k) \le \sum_{j:(i,j)\in\mathcal{A}} v_{ij}^k, \qquad \forall\ i \in \mathcal{N} \setminus \{o^k, d^k\}, k \in \mathcal{K}, \tag{3.6}$$

$$\sum_{j:(o^k,j)\in\mathcal{A}} v_{o^k j}^k \ge e^k, \qquad \forall\ k \in \mathcal{K}, \tag{3.7}$$

$$\sum_{j:(j,d^k)\in\mathcal{A}} (v_{jd^k}^k + \tau_{jd^k} x_{jd^k}^k) \le l^k, \qquad \forall\ k \in \mathcal{K}, \tag{3.8}$$

$$v_{ij}^k \le M x_{ij}^k, \qquad \forall\ (i,j) \in \mathcal{A}, k \in \mathcal{K}, \tag{3.9}$$

$$b_{ijr} - M(1 - z_{ijr}^k) \le v_{ij}^k \le b_{ijr} + M(1 - z_{ijr}^k), \quad \forall\ (i,j) \in \mathcal{A}, k \in \mathcal{K}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{3.10}$$

$$w_i^k = \begin{cases} \sum_{j:(i,j)\in\mathcal{A}} v_{ij}^k - e^k, & i = o^k, \\ l^k - \sum_{j:(j,i)\in\mathcal{A}} (v_{ji}^k + \tau_{ji} x_{ji}^k), & i = d^k, \qquad \forall\ i \in \mathcal{N}, \forall\ k \in \mathcal{K}, \\ \sum_{j:(i,j)\in\mathcal{A}} v_{ij}^k - \sum_{j:(j,i)\in\mathcal{A}} (v_{ji}^k + \tau_{ji} x_{ji}^k), & \text{otherwise}, \end{cases} \tag{3.11}$$

$$x_{ij}^k \in \{0,1\} \text{ and } v_{ij}^k \ge 0, \qquad \forall\ (i,j) \in \mathcal{A}, k \in \mathcal{K}, \tag{3.12}$$

$$y_{ijr} \in \mathbb{N}_{\ge 0} \text{ and } b_{ijr} \ge 0, \qquad \forall\ (i,j) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{3.13}$$

$$z_{ijr}^k \in \{0,1\}, \qquad \forall\ (i,j) \in \mathcal{A}, k \in \mathcal{K}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{3.14}$$

$$w_i^k \ge 0, \qquad \forall\ i \in \mathcal{N}, k \in \mathcal{K}. \tag{3.15}$$

The objective function (3.1) indicates the total cost to be minimized, which includes three terms for the total fixed cost, total flow cost, and total holding cost, respectively. Constraints (3.2)–(3.5) are imposed to define the routing and the consolidation plans. Specifically, constraints (3.2) are *flow balance constraints*, ensuring that each commodity travels along one flat path from its origin to its destination. Constraints (3.3) and (3.4) are *capacity constraints*. They ensure that the total quantity of commodities in each consolidation of an arc does not exceed the total capacity of the vehicles assigned to each consolidation of the arc, and restrict that $y_{ijr} = \lceil (\sum_{k\in\mathcal{K}} q^k z^k_{ijr})/u_{ij} \rceil$ for every $(i,j)\in\mathcal{A}$, $r\in\{1,2,...,|\mathcal{K}|\}$, which equals the number of vehicles needed by consolidation $C^{(i,j)}_r$ of arc (i,j). Constraints (3.5) are *consolidation coverage constraints*, ensuring that for every arc (i,j) on the flat path of commodity $k\in\mathcal{K}$, there exists a consolidation of arc (i,j) that contains k. Constraints (3.6) are imposed to define the departure schedule. Specifically, constraints (3.6)–(3.8) are imposed on commodities' departure times with respect to the travel time of each arc, as well as the earliest available time and the due time of each commodity. Constraints (3.9) ensure that for each commodity, its departure time from each of its unvisited nodes is zero. Constraints (3.10) ensure that for each arc $(i,j)\in\mathcal{A}$, the commodities that are consolidated to be shipped together through (i,j) have the same departure time from node i. They are similar to synchronization constraints in vehicle routing literature (Soares et al. 2024). Constraints (3.11) are imposed to define the holding time for each commodity $k\in\mathcal{K}$ and each node $i\in\mathcal{N}$. Constraints (3.12)-(3.15) define the domains of all the decision variables.

For each feasible solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\boldsymbol{v},\boldsymbol{b},\boldsymbol{w})$ of model DO, $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ corresponds to a flat solution $(\mathcal{P},\mathcal{C})$, and $\boldsymbol{v}$ corresponds to a departure schedule $\mathcal{T}$ that satisfies (2.1)–(2.4), which imply that such $(\mathcal{P},\mathcal{C},\mathcal{T})$ forms a feasible solution to the deterministic CTSNDP. As far as we know, model DO is the first compact MILP model of the deterministic CTSNDP that utilizes consolidation indices, and thus, we refer to it as the *consolidation-indexed* MILP model of the deterministic CTSNDP.

3.2. Two-Stage MINLP Optimization Models for Robust CTSNDP

As explained in Section 2, for both the two variants of the robust CTSNDP, $(\boldsymbol{x},\boldsymbol{z})$ of the first-stage decisions needs to ensure the existence of a departure schedule that satisfies the constraints with respect to commodities' earliest available times and due times under the nominal scenario. For this, we need to introduce decision variables $\overline{v}^k_{ij}$ and $\overline{b}_{ijr}$ to indicate commodities' departure times and consolidations' departure times for the nominal scenario, similar to the variables v^k_{ij} and b_{ijr} of model DO. Moreover, in the second stage, constraints with respect to the commodities' due times are relaxed, but delay penalties are imposed. As a result, we need to introduce an additional decision variable s^k for each $k\in\mathcal{K}$, indicating the delay of commodity k's arrival at its destination.

3.2.1. Robust Optimization Model Following Section 2.2.1, the RO-CTSNDP can be formulated as a two-stage MINLP (model RO) below, where M denotes a sufficiently large constant:

$$[\text{RO}]\ \min \sum_{(i,j)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|} f_{ij}\cdot y_{ijr} + \sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}} (c_{ij}^k q^k)\cdot x_{ij}^k + F_{RP}(\boldsymbol{x},\boldsymbol{z}) \tag{3.16}$$

$$\text{s.t. } (3.2)-(3.5), (3.12)-(3.14) \tag{3.17}$$

$$\sum_{j:(j,i)\in\mathcal{A}} (\overline{v}_{ji}^k + \overline{\tau}_{ji} x_{ji}^k) \le \sum_{j:(i,j)\in\mathcal{A}} \overline{v}_{ij}^k, \qquad \forall\, i\in\mathcal{N}\setminus\{o^k,d^k\}, k\in\mathcal{K}, \tag{3.18}$$

$$\sum_{j:(o^k,j)\in\mathcal{A}} \overline{v}_{o^kj}^k \ge e^k, \qquad \forall\, k\in\mathcal{K}, \tag{3.19}$$

$$\sum_{j:(j,d^k)\in\mathcal{A}} (\overline{v}_{jd^k}^k + \overline{\tau}_{jd^k} x_{id^k}^k) \le l^k, \qquad \forall\, k\in\mathcal{K}, \tag{3.20}$$

$$\overline{v}_{ij}^k \le M x_{ij}^k, \qquad \forall\, (i,j)\in\mathcal{A}, k\in\mathcal{K}, \tag{3.21}$$

$$\overline{b}_{ijr} - M(1-z_{ijr}^k) \le \overline{v}_{ij}^k \le \overline{b}_{ijr} + M(1-z_{ijr}^k), \qquad \forall\, (i,j)\in\mathcal{A}, k\in\mathcal{K}, r\in\{1,2,...,|\mathcal{K}|\}, \tag{3.22}$$

$$\overline{v}_{ij}^k \ge 0, \qquad \forall\, (i,j)\in\mathcal{A}, k\in\mathcal{K}, \tag{3.23}$$

$$\overline{b}_{ijr} \ge 0, \qquad \forall\, (i,j)\in\mathcal{A}, r\in\{1,2,...,|\mathcal{K}|\}. \tag{3.24}$$

Here, $F_{RP}(\boldsymbol{x},\boldsymbol{z})$ indicates the worst-case second-stage cost and can be calculated by the max-min optimization model below, referred to as model RP$(\boldsymbol{x},\boldsymbol{z})$, where M_1 is a sufficiently large constant.

$$[\text{RP}(\boldsymbol{x},\boldsymbol{z})]\quad F_{RP}(\boldsymbol{x},\boldsymbol{z}) = \max_{\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}):\boldsymbol{\delta}\in\mathbb{U}(\Gamma)}\ \min \sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{N}} (h^k q^k)\cdot w_i^k + \sum_{k\in\mathcal{K}} g^k\cdot s^k \tag{3.25}$$

$$\text{s.t. } \sum_{j:(j,i)\in\mathcal{A}} \Big(v_{ji}^k + \sum_{r=1}^{|\mathcal{K}|} \tilde{\tau}_{jir} z_{jir}^k\Big) \le \sum_{j:(i,j)\in\mathcal{A}} v_{ij}^k, \qquad \forall\, i\in\mathcal{N}\setminus\{o^k,d^k\}, k\in\mathcal{K}, \tag{3.26}$$

$$\sum_{j:(o^k,j)\in\mathcal{A}} v_{o^kj}^k \ge e^k, \qquad \forall\, k\in\mathcal{K}, \tag{3.27}$$

$$\sum_{j:(j,d^k)\in\mathcal{A}} \Big(v_{jd^k}^k + \sum_{r=1}^{|\mathcal{K}|} \tilde{\tau}_{jd^kr} z_{jd^kr}^k\Big) \le l^k + s^k, \qquad \forall\, k\in\mathcal{K}, \tag{3.28}$$

$$v_{ij}^k \le M_1 x_{ij}^k, \qquad \forall\, (i,j)\in\mathcal{A}, k\in\mathcal{K}, \tag{3.29}$$

$$v_{ij}^k \le b_{ijr} + M_1(1-z_{ijr}^k), \qquad \forall\, (i,j)\in\mathcal{A}, k\in\mathcal{K}, r\in\{1,2,...,|\mathcal{K}|\}, \tag{3.30}$$

$$v_{ij}^k \ge b_{ijr} - M_1(1-z_{ijr}^k), \qquad \forall\, (i,j)\in\mathcal{A}, k\in\mathcal{K}, r\in\{1,2,...,|\mathcal{K}|\}, \tag{3.31}$$

$$w_i^k \ge \begin{cases} \sum_{j:(i,j)\in\mathcal{A}} v_{ij}^k - e^k, & i = o^k, \\ (l^k+s^k) - \sum_{j:(j,i)\in\mathcal{A}} (v_{ji}^k + \sum_{r=1}^{|\mathcal{K}|} \tilde{\tau}_{jir} z_{jir}^k), & i = d^k, \qquad \forall\, i\in\mathcal{N}, \forall\, k\in\mathcal{K}, \\ \sum_{j:(i,j)\in\mathcal{A}} v_{ij}^k - \sum_{j:(j,i)\in\mathcal{A}} (v_{ji}^k + \sum_{r=1}^{|\mathcal{K}|} \tilde{\tau}_{jir} z_{jir}^k), & \text{otherwise}, \end{cases} \tag{3.32}$$

$$v_{ij}^k \geq 0, \qquad \forall\ (i,j) \in \mathcal{A}, k \in \mathcal{K}, \tag{3.33}$$

$$b_{ijr} \geq 0, \qquad \forall\ (i,j) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{3.34}$$

$$w_i^k \geq 0, \qquad \forall i \in \mathcal{N}, k \in \mathcal{K}, \tag{3.35}$$

$$s^k \geq 0, \qquad \forall k \in \mathcal{K}. \tag{3.36}$$

The objective (3.16) of model RO is to minimize the sum of the deterministic first-stage cost and the worst-case second-stage cost with respect to the uncertainty set $\mathbb{U}(\Gamma)$ on $\boldsymbol{\delta}$. The first-stage cost includes the fixed costs and the flow costs shown in the first two terms of (3.16). The worst-case second-stage cost is represented by $F_{RP}(\boldsymbol{x},\boldsymbol{z})$. In model RO, constraints in (3.17) are the same as those of model DO imposed on $(\boldsymbol{x},\boldsymbol{z})$. Constraints (3.18)–(3.24) are similar to those in model DO, with τ_{ji} replaced by the nominal travel times $\overline{\tau}_{ji}$. These constraints are imposed to ensure the existence of a feasible departure schedule under the nominal scenario.

The max-min optimization model RP$(\boldsymbol{x},\boldsymbol{z})$ is to compute the worst-case second-stage cost for $(\boldsymbol{x},\boldsymbol{z})$. Given any $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ with $\boldsymbol{\delta} \in \mathbb{U}(\Gamma)$, the inner minimization problem of RP$(\boldsymbol{x},\boldsymbol{z})$ needs to determine variables $(\boldsymbol{v},\boldsymbol{b},\boldsymbol{w},\boldsymbol{s})$, with the objective of minimizing the second-stage cost that equals the sum of the holding costs and delay penalties as shown in (3.25). Most of the constraints in the inner minimization problem are the same as those of model DO imposed on $(\boldsymbol{v},\boldsymbol{b},\boldsymbol{w})$, except (3.26), (3.28) and (3.32). Compared with constraints (3.6), (3.8), and (3.11) of model DO, constraints (3.26), (3.28), and (3.32) replace $\tau_{ji}x_{ji}^k$ with $\sum_{r=1}^{|\mathcal{K}|} \tilde{\tau}_{jir} z_{jir}^k$ for each $(j,i) \in \mathcal{A}$, as the latter indicates the actual travel time of commodity k on arc (j,i) if k passes through (j,i). Moreover, the decision variable s^k for $k \in \mathcal{K}$ is included in the right-hand sides of constraints (3.28) and (3.32), in order to represent the delay in commodity k's arrival at its destination.

Let $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ denote the optimal objective value of the inner minimization problem of model RP$(\boldsymbol{x},\boldsymbol{z})$, which is the second-stage cost of $(\boldsymbol{x},\boldsymbol{z})$ for scenario $\boldsymbol{\delta}$. This can be calculated by the following linear program, which is referred to as model LP$(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$:

$$[\text{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))] \quad F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) = \min \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{N}} (h^k q^k) \cdot w_i^k + \sum_{k\in\mathcal{K}} g^k \cdot s^k \tag{3.37}$$

$$\text{s.t.} \quad (3.26) - (3.36). \tag{3.38}$$

3.2.2. Robust Satisficing Model Following Section 2.2.2, RS-CTSNDP can also be formulated as a two-stage MINLP (model RS) below:

$$[\text{RS}] \ \min \ \rho \tag{3.39}$$

$$\text{s.t.} \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} (c_{ij}^k q^k) \cdot x_{ij}^k + \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} f_{ij} \cdot y_{ijr} + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} \leq \rho \|\boldsymbol{\delta}\|_1, \ \forall\ \boldsymbol{\delta} \in \mathbb{U}, \tag{3.40}$$

$$\rho \geq 0, \ (3.2) - (3.5), (3.12) - (3.14), (3.18) - (3.24). \tag{3.41}$$

Model RS aims to minimize ρ, which represents the worst-case magnitude of the deviation from the prescribed cost target, normalized by the total relative deviation $\|\boldsymbol{\delta}\|_1$ of the travel times. Here, $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ is the optimal objective value of model LP$(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ defined in (3.37) and (3.38), indicating the second-stage cost of $(\boldsymbol{x},\boldsymbol{z})$ under any given $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ with $\boldsymbol{\delta}$ in the uncertainty set $\mathbb{U}$. In model RS, constraints (3.40) specify that for every possible scenario $\boldsymbol{\delta}$, the deviation of the total two-stage cost from the prescribed target $\mathcal{Z}$ cannot exceed $\rho\|\boldsymbol{\delta}\|_1$. The domain of variable ρ is defined by $\rho \geq 0$ in (3.41). Other constraints in (3.41) are the same as those in model RO, ensuring that $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ forms a nominal timely-implementable first-stage solution.

4. Exact Algorithms

Following the C&CG framework of Zeng and Zhao (2013), we develop two exact algorithms to solve model RO and model RS, respectively. Their critical step is the solution to a subproblem, aiming to compute the worst-case scenario $\boldsymbol{\delta}$ for a given first-stage solution. For model RO, as in existing studies, this subproblem is a max-min problem, which can be reformulated as an MILP using the dual of its inner minimization LP, and solved by an optimization solver. For model RS, this subproblem is more complicated, as it involves max-min fractional optimization and cannot be reformulated as an MILP. To address this, we develop a novel enhanced bisection search procedure.

Below we first illustrate and analyze our C&CG algorithm for model RS in details, and then discuss our C&CG algorithm for model RO. Their implementations, including acceleration techniques, are illustrated in an online repository at `https://github.com/SSN0712/2024.06.26`.

4.1. C&CG Algorithm for Robust Satisficing Model

For any given first-stage solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ we define $F_1(\boldsymbol{x},\boldsymbol{y})$ below as its first-stage total cost:

$$F_1(\boldsymbol{x},\boldsymbol{y}) = \sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}}(c_{ij}^k q^k)\cdot x_{ij}^k + \sum_{(i,j)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|} f_{ij}\cdot y_{ijr} \tag{4.1}$$

Let $\mathcal{X}$ denote the domain of $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\overline{\boldsymbol{v}},\overline{\boldsymbol{b}})$ defined by linear constraints (3.17)–(3.24), and $\mathcal{Q}(\boldsymbol{\delta})$ denote the domain of $(\boldsymbol{v},\boldsymbol{b},\boldsymbol{w},\boldsymbol{s})$ defined by linear constraints (3.26)–(3.36) under the realized travel time $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$. For any possible scenario $\boldsymbol{\delta}\in\mathbb{U}$, $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ defined in (3.37)-(3.38) represents the second-stage cost of any first-stage solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ under $\boldsymbol{\delta}$. Thus, model RS proposed in Section 3.2.2 can be rewritten as the following noncompact MILP:

$$\begin{aligned}
[\text{RSMILP}]\ \ &\min\ \rho \\
&\text{s.t. } F_1(\boldsymbol{x},\boldsymbol{y}) + \sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{N}}(h^k q^k)\cdot w_i^{k(\boldsymbol{\delta})} + \sum_{k\in\mathcal{K}} g^k\cdot s^{k(\boldsymbol{\delta})} - \mathcal{Z} \leq \rho\|\boldsymbol{\delta}\|_1, \quad \forall\ \boldsymbol{\delta}\in\mathbb{U}, && (4.2)\\
&(\boldsymbol{v}^{(\boldsymbol{\delta})},\boldsymbol{b}^{(\boldsymbol{\delta})},\boldsymbol{w}^{(\boldsymbol{\delta})},\boldsymbol{s}^{(\boldsymbol{\delta})}) \in \mathcal{Q}(\boldsymbol{\delta}), \quad \forall\ \boldsymbol{\delta}\in\mathbb{U}, && (4.3)\\
&\rho \geq 0, (\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\overline{\boldsymbol{v}},\overline{\boldsymbol{b}}) \in \mathcal{X}.
\end{aligned}$$

Here, $(\boldsymbol{v}^{(\boldsymbol{\delta})}, \boldsymbol{b}^{(\boldsymbol{\delta})}, \boldsymbol{w}^{(\boldsymbol{\delta})}, \boldsymbol{s}^{(\boldsymbol{\delta})})$ represents a vector of second-stage decision variables associated with each possible scenario $\boldsymbol{\delta}$ in $\mathbb{U}$. Constraints (4.2) and (4.3) ensure that the deviation of the total two-stage cost from the prescribed target $\mathcal{Z}$ does not exceed $\rho\|\boldsymbol{\delta}\|_1$ for every possible scenario $\boldsymbol{\delta} \in \mathbb{U}$. As a result, solving model RS is reduced to solving the above noncompact MILP model RSMILP.

Model RSMILP can be relaxed by replacing $\mathbb{U}$ in (4.2) and (4.3) with any of its subsets $\Lambda \subseteq \mathbb{U}$. The resulting relaxation is referred to as model RSMILP(Λ), and can be strengthened by appending to Λ more scenarios $\boldsymbol{\delta} \in \mathbb{U}$. When $\Lambda = \mathbb{U}$, models RSMILP(Λ) and RSMILP are equivalent.

4.1.1. Computing the worst-case scenario $\boldsymbol{\delta}$ by enhanced bisection search Consider any given first-stage solution $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$. For any $\boldsymbol{\delta} \in \mathbb{U}$, the ratio $(F_1(\boldsymbol{x}, \boldsymbol{y}) + F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z})/\|\boldsymbol{\delta}\|_1$ represents a *normalized cost deviation* from the prescribed target under scenario $\boldsymbol{\delta}$. Here we slightly abuse the notation to define that $\sigma/\|\boldsymbol{0}\|_1 = 0$ for $\sigma = 0$, $\sigma/\|\boldsymbol{0}\|_1 = +\infty$ for $\sigma > 0$, and $\sigma/\|\boldsymbol{0}\|_1 = -\infty$ for $\sigma < 0$. Accordingly, constraints (4.2) and (4.3) in model RSMILP imply that the normalized cost deviation with respect to the prescribed target $\mathcal{Z}$ cannot exceed ρ for all $\boldsymbol{\delta} \in \mathbb{U}$.

The maximum value of the normalized cost deviation over all $\boldsymbol{\delta} \in \mathbb{U}$ is defined as the *worst-case normalized cost deviation* of $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, and the corresponding $\boldsymbol{\delta}$ that leads to the ratio achieving the maximum value is referred to as the *worst-case scenario* for $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, with respect to model RS. Computing such a worst-case scenario $\boldsymbol{\delta}$ can be formulated as the following *max-min fractional optimization model*, which is referred to as model FO$(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$:

$$[\mathrm{FO}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})] \;\; \rho^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) = \max_{\boldsymbol{\delta} \in \mathbb{U}} \; \frac{F_1(\boldsymbol{x}, \boldsymbol{y}) + F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z}}{\|\boldsymbol{\delta}\|_1}. \tag{4.4}$$

where $F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ is defined by a minimization problem LP$(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ in (3.37)-(3.38) and $\rho^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ denotes the optimal objective value of model FO$(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$.

Solving the above max-min fractional optimization model FO$(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ to exact optimality is challenging, as it cannot directly be reformulated as an MILP, nor can it directly be solved by the classic bisection search. To tackle this, we develop a novel enhanced bisection search procedure. It starts with a lower bound ρ_l and an upper bound ρ_h on the optimal objective value $\rho^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ of model FO$(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$. In each iteration, it first evaluates whether the middle point $\hat{\rho} = (\rho_l + \rho_u)/2$ is larger than $\rho^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$. If $\hat{\rho}$ is larger, the upper bound ρ_h is decreased to $\hat{\rho}$. Otherwise, the lower bound ρ_l is increased to $\hat{\rho}$. After the obtained lower bound ρ_l is further enhanced, the procedure proceeds to the next iteration unless ρ_l is proved to equal $\rho^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$.

Given any guessed value $\hat{\rho}$, consider the following optimization model, which does not involve fractional optimization and whose optimal objective value is denoted by $G(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \hat{\rho})$:

$$G(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \hat{\rho}) = \max_{\boldsymbol{\delta} \in \mathbb{U}} \; F_1(\boldsymbol{x}, \boldsymbol{y}) + F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}\|_1. \tag{4.5}$$

Lemma 4.1 below indicates that one can determine whether $\hat{\rho}$ is less than, greater than, or equal to $\rho^*(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ by evaluating the value of $G(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \hat{\rho})$.

Lemma 4.1 *If* $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})>0$, *then* $\hat{\rho}<\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. *Otherwise, if* $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})\leq 0$, *then* $\hat{\rho}\geq\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. *If* $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})=0$, *then* $\hat{\rho}=\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.

To solve $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$, which is a max-min problem, we need to reformulate it as an MILP. For this, we establish Proposition 4.1 below, stating an *optimality property of* $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$ *to restrict the domains of variables* δ_{ijr}.

Proposition 4.1 *There exists an optimal solution to the nonlinear optimization model defined in* (4.5) *such that (i)* $\delta_{ijr}\in\{-1,0,1\}$ *for each* $(i,j)\in\mathcal{A}$, *and that (ii)* $\delta_{ijr}\in\{0,1\}$ *for each* $(i,j)\in\mathcal{A}$ $r\in\{1,2,...,|\mathcal{K}|\}$ *with* $\sum_{k\in\mathcal{K}}h^kq^k\hat{\tau}_{ij}\leq\hat{\rho}$.

Recall that $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ is defined by a linear program in (3.37)–(3.38). Let β_i^k, γ^k, ψ^k, η_{ij}^k, θ_{ijr}^k, ξ_{ijr}^k, and λ_i^k denote the dual variables associated with its constraints (3.26)–(3.32). Let Ω indicate the feasible domain of its dual, which is a convex polyhedron defined by linear constraints. Accordingly, based on Proposition 4.1, we can reformulate $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$ as an MILP shown in Proposition 4.2.

Proposition 4.2 $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$ *defined in* (4.5) *can be equivalently written as the following MILP:*

$$\begin{aligned}
\max\quad & F_1(\boldsymbol{x},\boldsymbol{y})-\mathcal{Z}+\sum_{(j,i)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|}\hat{\varphi}_{jir}-\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}}(M_1x_{ij}^k)\cdot\eta_{ij}^k+\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|}[M_1(z_{ijr}^k-1)]\cdot(\theta_{ijr}^k+\xi_{ijr}^k)\\
&+\sum_{k\in\mathcal{K}}e^k\cdot(\gamma^k-\lambda_{o^k}^k)+\sum_{k\in\mathcal{K}}l^k\cdot(\lambda_{d^k}^k-\psi^k) && (4.6)\\
s.t.\quad & (\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\psi},\boldsymbol{\eta},\boldsymbol{\theta},\boldsymbol{\xi},\boldsymbol{\lambda})\in\Omega, && (4.7)\\
& \hat{\zeta}_{ijr,-1}+\hat{\zeta}_{ijr,1}+\hat{\zeta}_{ijr,0}=1,\quad\forall\ (i,j)\in\mathcal{A},r\in\{1,2,...,|\mathcal{K}|\}, && (4.8)\\
& M_2(\hat{\zeta}_{jir,\ell}-1)\leq\hat{\varphi}_{jir}-\Big(\sum_{k\in\mathcal{K}_i}z_{jir}^k(\beta_i^k-\lambda_i^k)+\sum_{k\in\mathcal{K}_i^d}z_{jir}^k(\psi^k-\lambda_i^k)\Big)\tilde{\tau}_{jir,\ell}+\hat{\rho}|\ell|\leq M_2(1-\hat{\zeta}_{jir,\ell}),\\
&\qquad\qquad\forall\ (j,i)\in\mathcal{A},r\in\{1,2,...,|\mathcal{K}|\},\ell\in\{-1,0,1\}, && (4.9)\\
& \hat{\zeta}_{ijr,-1}=0,\quad\forall\ (i,j)\in\mathcal{A}(\hat{\rho}),r\in\{1,2,...,|\mathcal{K}|\}, && (4.10)\\
& \hat{\zeta}_{ijr,\ell}\in\{0,1\},\quad\forall\ (i,j)\in\mathcal{A},r\in\{1,2,...,|\mathcal{K}|\},\ell\in\{-1,0,1\}. && (4.11)
\end{aligned}$$

where $\mathcal{K}_i=\{k\in\mathcal{K}:i\neq o^k$ *and* $i\neq d^k\}$, $\mathcal{K}_i^d=\{k\in\mathcal{K}:i=d^k\}$, $\mathcal{A}(\hat{\rho})=\{(i,j)\in\mathcal{A}:\sum_{k\in\mathcal{K}}h^kq^k\hat{\tau}_{ij}\leq\hat{\rho}\}$, *and* M_2 *is a sufficiently large constant. Given* $\hat{\zeta}_{ijr,-1}$ *and* $\hat{\zeta}_{ijr,1}$ *for* $(i,j)\in\mathcal{A}$ *and* $r\in\{1,...,|\mathcal{K}|\}$ *in the optimal solution of this MILP, the corresponding worst-case scenario* $\boldsymbol{\delta}$ *can be obtained by*

$$\delta_{ijr}=-\hat{\zeta}_{ijr,-1}+\hat{\zeta}_{ijr,1},\quad\forall\ (i,j)\in\mathcal{A},r\in\{1,...,|\mathcal{K}|\}. \tag{4.12}$$

Next, consider any given lower bound ρ_l on $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. Let $\boldsymbol{\delta}(\rho_l)$ indicate the realization of $\boldsymbol{\delta}$, derived by (4.12) from the optimal solution to model $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l)$ defined in (4.6)–(4.11). Define ρ_l' below to indicate the normalized cost deviation under $\boldsymbol{\delta}(\rho_l)$.

$$\rho_l'=(F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l)))-\mathcal{Z})\,/\|\boldsymbol{\delta}(\rho_l)\|_1. \tag{4.13}$$

Lemma 4.2 below implies that we can always enhance ρ_l to a better lower bound ρ_l'. As explained later in Remark 4.1, this enhancement is essential for computing the exact value of $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.

Lemma 4.2 *If $\rho_l \le \rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, then ρ_l' defined in (4.13) satisfies that $\rho_l \le \rho_l' \le \rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.*

Below, we provide a summary of our enhanced bisection search procedure for any given first-stage solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ in Algorithm 1, along with its correctness and convergence in Theorem 4.1.

Algorithm 1 Enhanced Bisection Search Procedure for Any Given $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$

1. If $(F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\mathbf{0})) - \mathcal{Z}) > 0$, return $+\infty$ as the value of the worst-case normalized cost deviation of $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, and $\mathbf{0}$ as the worst-case scenario for $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.
2. Initialize the values of ρ_l and ρ_h such that $\rho_l \le \rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) \le \rho_h$.
3. Set $\hat{\rho} = (\rho_h + \rho_l)/2$, solve the maximization MILP model (4.6)–(4.11) to compute $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$.
4. If $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho}) > 0$, increase ρ_l to $\hat{\rho}$, and if $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho}) < 0$, decrease ρ_h to $\hat{\rho}$. Then go to Step 5. However, if $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho}) = 0$, increase ρ_l to $\hat{\rho}$, derive the scenario $\boldsymbol{\delta}(\rho_l)$ from the optimal solution to the model by (4.12), and then go to Step 6.
5. Enhancement: Solve the MILP model (4.6)–(4.11) to compute $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l)$, derive the scenario $\boldsymbol{\delta}(\rho_l)$ by (4.12), and compute ρ_l' from $\boldsymbol{\delta}(\rho_l)$ by (4.13). If $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l) = 0$, then go to Step 6. Otherwise, set ρ_l to ρ_l', and go to Step 3 for the next iteration.
6. Return ρ_l as the worst-case normalized cost deviation $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, and return $\boldsymbol{\delta}(\rho_l)$ as the worst-case scenario for $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.

Theorem 4.1 *Consider any given feasible first-stage decisions $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. (i) Algorithm 1 is guaranteed to terminate within a finite number of iterations, with the value of $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ and a worst-case scenario $\boldsymbol{\delta}$ for $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ returned. (ii) Let $\rho_l^{(0)}$ and $\rho_h^{(0)}$ denote the initial values of ρ_l and ρ_h assigned in Step 3 of Algorithm 1. Then, for any $\epsilon > 0$, after $\lceil \log_2((\rho_h^{(0)} - \rho_l^{(0)})/\epsilon) \rceil$ iterations of Steps 3–6, Algorithm 1 obtains a lower bound ρ_l on $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ and a scenario $\boldsymbol{\delta}(\rho_l) \in \mathbb{U}$, such that $\rho_l \le \rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) \le \rho_l + \epsilon$ and $F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l))) - \mathcal{Z} \ge \rho_l \|\boldsymbol{\delta}(\rho_l)\|_1$.*

Theorem 4.1 indicates that our enhanced bisection search procedure in Algorithm 1 is an exact algorithm that solves model FO$(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ within a finite number of iterations. It also implies that Algorithm 1 solves model FO$(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ to an accuracy $\epsilon > 0$ within $\lceil \log_2((\rho_h^{(0)} - \rho_l^{(0)})/\epsilon) \rceil$ iterations.

Remark 4.1 (Enhancement in Step 5) *The enhancement in Step 5 is essential to guarantee that Algorithm 1 solves model FO$(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ to exact optimality within a finite number of iterations. Without it, Algorithm 1 functions only as a standard bisection search procedure, in which valid* *lower and upper bounds of $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ can be obtained with their gap smaller than a given tolerance $\epsilon > 0$ within a finite number of iterations. However, this standard bisection search procedure cannot guarantee to produce the exact value of $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ within a finite number of iterations.*

4.1.2. RS-C&CG Algorithm As shown in Algorithm 2 below, in each iteration n, where $n = 1, 2, \cdots$, our C&CG algorithm for model RS (referred to as RS-C&CG algorithm) first solves an optimal solution $(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}},\phi)$ to model RSMILP(Λ) as the master problem for a particular subset Λ of $\mathbb{U}$. It then applies the enhanced bisection search procedure in Algorithm 1 to solve FO($\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}$) as the subproblem, and obtains the worst-case normalized cost deviation $\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, denoted by $\rho^{(n)}$, as well as the corresponding worst-case scenario, denoted by $\boldsymbol{\delta}^{(n)}$. Since RSMILP($\Lambda$) is a relaxation of model RS, its optimal objective value obtained is a lower bound on the optimal objective value of model RS. Since $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ forms a nominal timely-implementable first-stage solution to model RS, $\max\{0,\rho^{(n)}\}$ provides an upper bound on the optimal objective value of model RS. If the lower bound equals the upper bound, model RS is solved to optimum. The algorithm terminates with an optimal solution given by $(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}})$. Otherwise, it appends the identified scenario $\boldsymbol{\delta}^{(n)}$ to Λ. Model RSMILP(Λ) of the master problem is extended and strengthened with new decision variables $(\boldsymbol{v}^{(\boldsymbol{\delta})},\boldsymbol{b}^{(\boldsymbol{\delta})},\boldsymbol{w}^{(\boldsymbol{\delta})},\boldsymbol{s}^{(\boldsymbol{\delta})})$ and their new constraints in (4.2)- (4.3). The algorithm proceeds to the next iteration. Theorem 4.2 below establishes the correctness and convergence of the algorithm.

Algorithm 2 RS-C&CG Algorithm for Solving Model RS

1. Initially, set the iteration number n to 1, and set the subset Λ of $\mathbb{U}$ to $\{\mathbf{0}\}$.
2. Solve the master problem, i.e., model RSMILP(Λ), to obtain its optimal objective value denoted by LB and its optimal solution denoted by $(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}},\phi)$.
3. Apply the enhanced bisection search procedure in Algorithm 1 to solve subproblem FO$(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}})$, so as to obtain the worst-case normalized cost deviation $\rho^*(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}})$, denoted by $\rho^{(n)}$, and to obtain the corresponding worst-case scenario, denoted by $\boldsymbol{\delta}^{(n)}$. Let UB denote $\max\{0,\rho^{(n)}\}$.
4. If $LB = UB$, then the algorithm terminates and returns an optimal solution given by $(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}})$. Otherwise, update $\Lambda = \Lambda \bigcup \{\boldsymbol{\delta}^{(n)}\}$, update $n = n+1$, and go to Step 2 for the next iteration.

Theorem 4.2 *Algorithm 2 returns an optimal solution to model RS in a finite number of iterations.*

4.2. C&CG Algorithm for Robust Optimization Model

Similarly, model RO proposed in Section 3.2.1 can be rewritten as the following noncompact MILP:

$$[\text{ROMILP}] \ \min \quad \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} (c_{ij}^k q^k)\cdot x_{ij}^k + \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} f_{ij}\cdot y_{ijr} + \phi \tag{4.14}$$

$$s.t.\quad \phi \geq \sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{N}}(h^k q^k)\cdot w_i^{k(\boldsymbol{\delta})} + \sum_{k\in\mathcal{K}} g^k \cdot s^{k(\boldsymbol{\delta})}, \quad \forall\, \boldsymbol{\delta}\in\mathbb{U}(\Gamma), \tag{4.15}$$

$$(\boldsymbol{v}^{(\boldsymbol{\delta})}, \boldsymbol{b}^{(\boldsymbol{\delta})}, \boldsymbol{w}^{(\boldsymbol{\delta})}, \boldsymbol{s}^{(\boldsymbol{\delta})}) \in \mathcal{Q}(\boldsymbol{\delta}), \quad \forall\, \boldsymbol{\delta}\in\mathbb{U}(\Gamma), \tag{4.16}$$

$$(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \overline{\boldsymbol{v}}, \overline{\boldsymbol{b}}) \in \mathcal{X}. \tag{4.17}$$

Here, ϕ is a newly introduced decision variable, and $(\boldsymbol{v}^{(\boldsymbol{\delta})}, \boldsymbol{b}^{(\boldsymbol{\delta})}, \boldsymbol{w}^{(\boldsymbol{\delta})}, \boldsymbol{s}^{(\boldsymbol{\delta})})$ represents a vector of second-stage decision variables associated with each possible scenario $\boldsymbol{\delta}$ in $\mathbb{U}(\Gamma)$. Constraints (4.15) and (4.16) ensure that ϕ equals the worst-case second-stage cost. As a result, solving the min-max-min model RO is reduced to solving the above noncompact MILP model ROMILP.

Model ROMILP can also be relaxed by replacing $\mathbb{U}(\Gamma)$ in constraints (4.15) and (4.16) with any of its subsets $\Lambda \subseteq \mathbb{U}(\Gamma)$. The resulting relaxation is referred to as model ROMILP(Λ). The relaxation can be strengthened by appending to Λ more scenarios $\boldsymbol{\delta}$ in $\mathbb{U}(\Gamma)$.

Our C&CG algorithm for model RO, referred to as RO-C&CG algorithm, iteratively solves model ROMILP(Λ) to obtain a first-stage solution $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ and appends its worst-case scenario $\boldsymbol{\delta}$ to Λ, until $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ implies an optimal solution. With respect to model RO, a scenario $\boldsymbol{\delta}$ is a worst-case scenario, if the second-stage cost of $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ equals under $\boldsymbol{\delta}$ the worst-case second-stage cost $F_{RP}(\boldsymbol{x}, \boldsymbol{z})$. Both $\boldsymbol{\delta}$ and $F_{RP}(\boldsymbol{x}, \boldsymbol{z})$ can be determined by solving model RP$(\boldsymbol{x}, \boldsymbol{z})$ in (3.25)–(3.36).

Our RO-C&CG algorithm for model RO is similar to Algorithm 2 for model RS, except that in Step 3, it can directly apply an optimization solver to solve model RP$(\boldsymbol{x}, \boldsymbol{z})$ as the subproblem. This is because by Proposition 4.3 below, RP$(\boldsymbol{x}, \boldsymbol{z})$ has an equivalent maximization MILP model.

Proposition 4.3 *The max-min model RP$(\boldsymbol{x}, \boldsymbol{z})$ defined by (3.25)–(3.36) for $F_{RP}(\boldsymbol{x}, \boldsymbol{z})$ can be equivalently written as a maximization MILP model.*

5. Computational Experiments

We performed two sets of computational experiments. The first set aimed to assess the performance of our exact algorithms in solving model RO and model RS. The second set aimed to evaluate the performance of solutions obtained from model RO and model RS with different parameters of the uncertainty sets and under different performance criteria. In the implementation of our RO-C&CG algorithm and RS-C&CG algorithm, we used the Gurobi solver (v.10.0.2) to solve the master problems and subproblems. All experiments were conducted on a PC with an Intel(R) Core(TM) i7-8700 CPU at 3.20 GHz and 64 GB RAM. The uncertainty budget Γ of model RO was set to $\lceil \mu_\Gamma \cdot |\mathcal{K}| \rceil$, the cost target $\mathcal{Z}$ of model RS was set to $\lceil (1+\mu_z)\cdot \mathcal{Z}_0 \rceil$. Values of μ_Γ and μ_z will be explained later for different experiments.

Based on the seven instance classes of the fixed-charge capacitated multi-commodity network design problem in Ghamlouche et al. (2003), we randomly generated 210 test instances of the

Table 5.1 Computational Performance of RO-C&CG and RS-C&CG Algorithms.

Class	$\lvert\mathcal{N}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{K}\rvert$	RO-C&CG opt%	g%	T	Im% mean	Im% max	RS-C&CG opt%	g%	T	Im% mean	Im% max
R4	10	60	10	100.0	0.0	0.2	6.4	29.7	100.0	0.0	0.3	37.8	100.0
R5	10	60	25	100.0	0.0	7.3	7.2	28.1	100.0	0.0	8.1	46.7	100.0
R6	10	60	50	73.3	0.4	9672.7	6.5	21.5	66.7	13.7	11520.1	46.1	85.5
R7	10	82	10	100.0	0.0	0.4	9.7	25.8	100.0	0.0	0.4	22.2	100.0
R8	10	83	25	100.0	0.0	11.7	11.8	23.1	100.0	0.0	12.3	48.7	85.4
R9	10	83	50	86.7	0.1	7117.9	6.9	11.9	100.0	0.0	1530.5	37.4	82.2
R10	20	120	40	100.0	0.0	489.5	9.0	26.7	90.0	0.5	4402.3	52.1	91.9
Mean				94.3	0.1	2471.4	8.2	29.7	93.8	1.9	2486.3	41.6	100.0

robust CTSNDP, which are available at `https://github.com/SSN0712/2024.06.26`. The numbers of nodes $|\mathcal{N}|$, arcs $|\mathcal{A}|$, and commodities $|\mathcal{K}|$ vary from 10 to 20, 60 to 120, and 10 to 50, respectively.

5.1. Algorithm Performance of RO-C&CG and RS-C&CG

In the first set of experiments, both RO-C&CG and the RS-C&CG were terminated when the running time exceeded eight hours or the optimality gap between the best upper and lower bounds found was below 0.01%. We solved the deterministic model DO by the Gurobi solver, and used its optimal objective value as the cost target $\mathcal{Z}_0$ in RS-C&CG. We set $\mu_\Gamma = \mu_z = 0.05$.

The computational results are presented in Table 5.1. For each class of the test instances and for each algorithm, we report the percentage of the instances solved to optimality in column opt%, the average optimality gap in column g% (defined as the percentage gap between the best upper and lower bounds found), the average computational time in CPU seconds in column T, and the mean and maximum of improvement ratio against the optimal deterministic solution in column Im%.

Table 5.1 shows the effectiveness of our RO-C&CG and RS-C&CG algorithms. Within the time limit, they both solve around 94% of the instances to optimality, and achieve optimality gaps of 0.1% and 2.6% on average, respectively. Columns Im% indicate that compared with the deterministic optimal solutions, the best upper bounds produced by our RO-C&CG and RS-C&CG algorithms significantly improve the objective values with respect to model RO and model RS by 8.2% and 41.6% on average and by 29.7% and 100.0% at maximum, respectively.

Table 5.1 confirms the computational tractability of both model RO and model RS that we derive for the robust CTSNDP, underscoring their practical usefulness. Notably, our study presents the first development of a C&CG-based algorithm for effectively solving a two-stage robust satisficing model, encouraging its future extensions to other problems that encompass uncertainties.

5.2. Solution Performance of Models RO and RS

Models RO and RS have distinct objectives for robustness. Following the approach in Bertsimas and Sim (2004) and Atamtürk and Zhang (2007), our second set of experiments evaluates the trade-offs for achieving the robustness. This involves comparing first-stage solutions obtained from models RO and RS based on their nominal, average, and worst-case performances.

- For the nominal performance, we calculated the total cost of each first-stage solution in the nominal scenario. For comparison, we solved model DO to obtain the minimum achievable total cost for the nominal scenario.
- For the average and the worst-case performances, we randomly generated a set Π of scenarios assumed to be uniformly distributed, and calculated the average and the worst-case total cost of each first-stage solution over these scenarios. For comparison, we computed the minimum achievable average total cost and the minimum achievable worst-case total cost by solving a stochastic programming model (SP) and a min-max optimization model (MM), which can both be formulated as an MILP by using our formulations of models RO and RS (see Appendix D).

Note that model DO assumes no travel time uncertainty, while models SP and MM presume knowledge of the travel time distribution, circumstances that are often unrealistic in practice.

For the tractability of models SP and MM, we set $|\Pi| = 200$. We used the Gurobi solver to solve models DO, SP, and MM. Our experiments focused on class R7, because its 30 instances were all solved to optimality for models RO, RS, DO, SP, and MM, while each of the classes R6, R8, R9, and R10 had some instances that could not be solved to optimality for models SP and MM within eight hours. For each instance in R7, we used RO-C&CG to solve model RO for each uncertainty budget $\Gamma \in \{1, 2, ..., 10\}$, and used RS-C&CG to solve model RS for each cost target $\mathcal{Z} = \lceil (1 + \mu_z) \cdot \mathcal{Z}_0 \rceil$ with $\mu_z \in \{0.02, 0.04, ..., 0.2\}$, where $\mathcal{Z}_0$ is set as the optimal objective value of model DO.

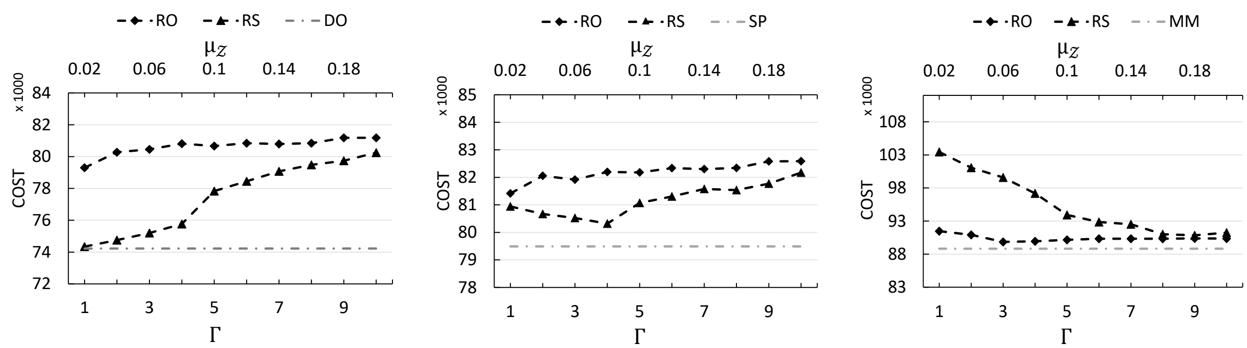
We first compare the total costs of the solutions obtained from models RO and RS in nominal scenarios with that of the optimal nominal solution obtained from model DO. The results are presented in Figure 1(a), where the total cost along the vertical axis is the mean across all instances in R7. It can be seen that solutions obtained from model RS exhibit better overall performance in the nominal scenario than those from model RO. When μ_z increases, the total nominal cost of the solutions from model RS gradually increases. In contrast, when Γ increases from 1, the total nominal cost of solutions obtained from model RO is often changed slightly.

For each instance in class R7, we generated 200 random scenarios to create Π for models SP and MM, and then calculated the average and the worst-case total costs of solutions obtained from models RO and RS over these random scenarios in Π, under different Γ and μ_z. We compare them with the minimum achievable average total cost obtained from model SP and the minimum achievable worst-case total cost obtained from model MM. The results are shown in Figures 1(b) and 1(c), where the total cost along the vertical axis is the mean across all the instances in R7.

Figure 1(b) shows that both model RO (with any Γ) and model RS (with any μ_z) can produce solutions of good average performance, within 4% from the minimum achievable average cost

Figure 1 Performance comparison of solutions from models RO and RS.

(a) Total costs in nominal scenario (b) Average total costs (c) Worst-case total costs



obtained from model SP, while solutions from model RS exhibit better average performance than model RO. Figure 1(c) shows that model RO (with any Γ) and model RS (with $\mu_z > 0.1$) can produce solutions of good worst-case performance, within 5% from the minimum achievable worst-case cost obtained from model MM, while solutions from model RO exhibit better worst-case performance than model RS. We can see that the impact of Γ on the average and the worst-case performances of model RO is not as significant as the impact of μ_z on these performances of model RS.

Our findings confirm the practical usefulness of model RO and model RS. Compared with model SP and model MM, they are computationally more tractable, and do not need the distribution information of travel times. They can achieve comparable average solution performance to that of model SP, and comparable worst-case solution performance to that of model MM, although model RS sometimes requires a proper parameter setting to achieve this. Compared with model RS, model RO has better worst-case solution performance, making it useful for conservative decision makers. Compared with model RO, model RS has better average performance, making it useful for decision makers who prioritize average performance but have limited distribution information about travel times. Moreover, the cost target of model RS is a more effective parameter for adjusting the trade-off involved in achieving the robustness guarantees for solutions obtained.

6. Conclusions

This paper studies a robust CTSNDP under travel time uncertainty, for which we derive several computationally tractable formulations and effective exact algorithms. It has established a strong foundation for future research: (i) There is great interest in enhancing our exact algorithms. One possible enhancement is to develop tailored branch-and-bound algorithms to solve those MILP models of both subproblems and master problems involved in our exact algorithm. (ii) As the first attempt at incorporating travel time uncertainty into robust service network design, we have developed robust optimization and robust satisficing models based on polyhedral uncertainty sets. Further exploration of alternative robust optimization approaches, such as the distributionally robust optimization approach, is an area of interest. (iii) Our robust optimization model and robust satisficing models, accompanied by their C&CG algorithms, provide a solid foundation that can be extended to tackle travel time uncertainty in other transportation problems.

References

Altın, A., Yaman, H., Pınar, M.Ç., 2011. The robust network loading problem under hose demand uncertainty: formulation, polyhedral analysis, and computations. INFORMS Journal on Computing 23, 75–89.

Atamtürk, A., Zhang, M., 2007. Two-stage robust network flow and design under demand uncertainty. Operations Research 55, 662–673.

Bertsimas, D., Brown, D.B., Caramanis, C., 2011. Theory and applications of robust optimization. SIAM Review 53, 464–501.

Bertsimas, D., Delarue, A., Jaillet, P., Martin, S., 2019. Travel time estimation in the age of big data. Operations Research 67, 498–515.

Bertsimas, D., Sim, M., 2004. The price of robustness. Operations Research 52, 35–53.

Boland, N., Hewitt, M., Marshall, L., Savelsbergh, M., 2017. The continuous-time service network design problem. Operations Research 65, 1303–1321.

Crainic, T.G., Hewitt, M., Toulouse, M., Vu, D.M., 2016. Service network design with resource constraints. Transportation Science 50, 1380–1393.

Cui, Z., Ding, J., Long, D.Z., Zhang, L., 2023. Target-based resource pooling problem. Production and Operations Management 32, 1187–1204.

Demir, E., Burgholzer, W., Hrušovskỳ, M., Arıkan, E., Jammernegg, W., Van Woensel, T., 2016. A green intermodal service network design problem with travel time uncertainty. Transportation Research Part B: Methodological 93, 789–807.

Fischetti, M., Monaci, M., 2009. Light robustness, in: Robust and online large-scale optimization. Springer, pp. 61–84.

Ghamlouche, I., Crainic, T.G., Gendreau, M., 2003. Cycle-based neighbourhoods for fixed-charge capacitated multicommodity network design. Operations Research 51, 655–667.

Hewitt, M., 2019. Enhanced dynamic discretization discovery for the continuous time load plan design problem. Transportation Science 53, 1731–1750.

Hewitt, M., Lehuédé, F., 2023. New formulations for the scheduled service network design problem. Transportation Research Part B: Methodological 172, 117–133.

Hrušovskỳ, M., Demir, E., Jammernegg, W., Van Woensel, T., 2018. Hybrid simulation and optimization approach for green intermodal transportation problem with travel time uncertainty. Flexible Services and Manufacturing Journal 30, 486–516.

Jaillet, P., Qi, J., Sim, M., 2016. Routing optimization under uncertainty. Operations Research 64, 186–200.

Koster, A.M., Kutschka, M., Raack, C., 2013. Robust network design: Formulations, valid inequalities, and computations. Networks 61, 128–149.

Lanza, G., Crainic, T.G., Rei, W., Ricciardi, N., 2021. Scheduled service network design with quality targets and stochastic travel times. European Journal of Operational Research 288, 30–46.

Layeb, S.B., Jaoua, A., Jbira, A., Makhlouf, Y., 2018. A simulation-optimization approach for scheduling in stochastic freight transportation. Computers & Industrial Engineering 126, 99–110.

Lium, A.G., Crainic, T.G., Wallace, S.W., 2009. A study of demand stochasticity in service network design. Transportation Science 43, 144–157.

Long, D.Z., Sim, M., Zhou, M., 2023. Robust satisficing. Operations Research 71, 61–82.

Marshall, L., Boland, N., Savelsbergh, M., Hewitt, M., 2021. Interval-based dynamic discretization discovery for solving the continuous-time service network design problem. Transportation Science 55, 29–51.

Nikolova, E., Stier-Moses, N.E., 2014. A mean-risk model for the traffic assignment problem with stochastic travel times. Operations Research 62, 366–382.

Pessoa, A.A., Poss, M., 2015. Robust network design with uncertain outsourcing cost. INFORMS Journal on Computing 27, 507–524.

Shu, S., Xu, Z., Baldacci, R., 2024. Incorporating holding costs in continuous-time service network design: New model, relaxation, and exact algorithm. Transportation Science 58, 279–556.

Soares, R., Marques, A., Amorim, P., Parragh, S.N., 2024. Synchronisation in vehicle routing: classification schema, modelling framework and literature review. European Journal of Operational Research 313, 817–840.

Wang, X., Crainic, T.G., Wallace, S.W., 2019. Stochastic network design for planning scheduled transportation services: The value of deterministic solutions. INFORMS Journal on Computing 31, 153–170.

Wang, Z., Qi, M., 2020. Robust service network design under demand uncertainty. Transportation Science 54, 676–689.

Wieberneit, N., 2008. Service network design for freight transportation: a review. OR spectrum 30, 77–112.

Yanıkoğlu, İ., Gorissen, B.L., den Hertog, D., 2019. A survey of adjustable robust optimization. European Journal of Operational Research 277, 799–813.

Zeng, B., Zhao, L., 2013. Solving two-stage robust optimization problems using a column-and-constraint generation method. Operations Research Letters 41, 457–461.

Zhou, M., Sim, M., Lam, S.W., 2022. Advance admission scheduling via resource satisficing. Production and Operations Management 31, 4002–4020.

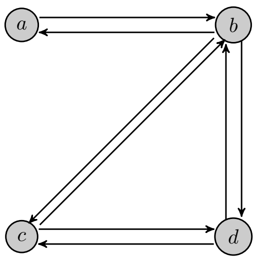
Appendix A: Glossary of Notation and Illustrative Examples

Table A.1 Glossary of notation.

Notation	Meaning
$\mathcal{D}$	(flat) network $\mathcal{D}=(\mathcal{N},\mathcal{A})$
$\mathcal{N}$	node set of network $\mathcal{D}$
$\mathcal{A}$	arc set of network $\mathcal{D}$
$\mathcal{K}$	set of commodities
o^k	origin of commodity $k\in\mathcal{K}$
d^k	destination of commodity $k\in\mathcal{K}$
q^k	demand of commodity $k\in\mathcal{K}$
τ_{ij}	travel time of arc $(i,j)\in\mathcal{A}$
c_{ij}^k	per-unit-of-flow cost of arc $(i,j)\in\mathcal{A}$ and commodity $k\in\mathcal{K}$
f_{ij}	fixed cost of arc $(i,j)\in\mathcal{A}$
u_{ij}	capacity of arc $(i,j)\in\mathcal{A}$
e^k	earliest available time of commodity $k\in\mathcal{K}$
l^k	latest arrival time of commodity $k\in\mathcal{K}$
h^k	per-unit-of-demand-and-time (in-storage holding) cost of commodity $k\in\mathcal{K}$ at a terminal
$\mathcal{P}=\{P^k\}_{k\in\mathcal{K}}$	routing plan with P^k representing a path for commodity $k\in\mathcal{K}$
$\mathcal{C}=\{C_r^\alpha\}_{\alpha\in\mathcal{A},r\in\{1,2,\dots,\lvert\mathcal{K}\rvert\}}$	consolidation plan with C_r^α subset of commodities and with r the consolidation index
$\mathcal{T}$	departure schedule
$f(\mathcal{P},\mathcal{C})$	total fixed cost and flow cost of solution $(\mathcal{P},\mathcal{C},\mathcal{T})$
$h(\mathcal{P},\mathcal{T})$	total holding cost of solution $(\mathcal{P},\mathcal{C},\mathcal{T})$
$g(\mathcal{P},\mathcal{T})$	total delay penalty of solution $(\mathcal{P},\mathcal{C},\mathcal{T})$
$\boldsymbol{\delta}$	vector of random variables $\delta_{\alpha r}$ for $\alpha\in\mathcal{A}$ and $r\in\{1,2,...,\lvert\mathcal{K}\rvert\}$
$\tilde{\tau}_{ij}$	uncertain travel time of arc $(i,j)\in\mathcal{A}$
$\tilde{\boldsymbol{\tau}}$	vector of uncertain travel times $\tilde{\tau}_{ij}$
$\bar{\tau}_{ij}$	nominal value of $\tilde{\tau}_{ij}$
$\hat{\tau}_{ij}$	maximum deviation of $\tilde{\tau}_{ij}$ with respect to nominal value $\bar{\tau}_{ij}$
$\mathcal{Z}$	prescribed target of the total two-stage cost
$\mathbb{U}$	support of vector $\boldsymbol{\delta}$
$\mathbb{U}(\Gamma)$	budgeted uncertainty set of vector $\boldsymbol{\delta}$ with Γ denoting a budget of uncertainty
$\mathbb{D}$	domain of all feasible solutions $(\mathcal{P},\mathcal{C},\mathcal{T})$
$\mathbb{F}$	domain of all nominal timely-implementable first-stage solutions
$\mathbb{T}(\mathcal{P},\mathcal{C},\tilde{\boldsymbol{\tau}})$	domain of departure schedule $\mathcal{T}$ with respect to solution $(\mathcal{P},\mathcal{C})$
$\boldsymbol{x}$	decision variables on routing plans
$\boldsymbol{z}$	decision variables on consolidation plans
$\boldsymbol{y}$	decision variables on numbers of vehicles
$\boldsymbol{v}$	decision variables on departure times
$\mathcal{Z}$, $\mathcal{Q}$	domains of decision variables
DO	consolidation-indexed formulation for deterministic CTSNDP
RO	robust optimization model for robust CTSNDP
RS	robust satisficing model for robust CTSNDP
RP$(\boldsymbol{x},\boldsymbol{z})$	max-min model defined by (3.25)–(3.36) for the worst-case second-stage cost
LP$(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}})$	inner minimization problem of model RP$(\boldsymbol{x},\boldsymbol{z})$ as a linear program
ROMILP	noncompact MILP defined in (4.14)-(4.17), a reformulation of model RO
RSMILP	noncompact MILP reformulation of model RS, with constraints (4.2)-(4.3) included
FO$(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$	model defined in (4.4) for the worst-case normalized cost deviation of $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$
SP	stochastic programming model in Appendix D to minimize expected total cost
MM	min-max optimization model in Appendix D to minimize worst-case total cost
$F_1(\boldsymbol{x},\boldsymbol{y})$	first-stage cost of solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$
$F_{RP}(\boldsymbol{x},\boldsymbol{z})$	worst-case second-stage cost of solution $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$
$F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}})$	optimal objective value of model LP$(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}})$, the minimum cost of $(\boldsymbol{x},\boldsymbol{z})$ under $\tilde{\boldsymbol{\tau}}$
$\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$	optimal objective value of model FO$(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$
$G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$	optimal objective value of model defined in (4.5)

Figure 2 **Illustrative examples for a CTSNDP instance, where (c) presents a flat solution that ships commodity 1 via arc (b,a), commodity 2 via arcs (d,b) and (b,a), and commodity 3 via arc (c,a), with commodities 1 and 3 consolidated on arc (b,a), and (d) presents a feasible solution based on the flat solution in (c) with departure and arrival times specified in the last two numbers within the brackets on the arcs.**

(a) Flat network $\mathcal{D}$



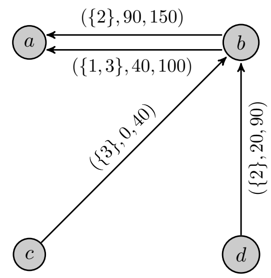
(b) Commodity data

k	o^k	d^k	q^k	e^k	l^k
1	b	a	25	0	160
2	d	a	30	20	180
3	c	a	40	0	180

(c) Flat solution

$\mathcal{P}$		$\mathcal{C}$	
k	P^k	α	$\mathcal{C}_r^\alpha$
1	(b,a)	(b,a)	$\{1,3\}$, $\{2\}$
2	(d,b,a)	(d,b)	$\{2\}$
3	(c,b,a)	(c,a)	$\{3\}$

(d) Feasible solution



Appendix B: Proof of Statements

B.1. Proof of Lemma 4.1

Recall that we slightly abuse the notation to define that $\sigma/\|\mathbf{0}\|_1 = 0$ for $\sigma = 0$, $\sigma/\|\mathbf{0}\|_1 = +\infty$ for $\sigma > 0$, and $\sigma/\|\mathbf{0}\|_1 = -\infty$ for $\sigma < 0$. Consider any $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) \in \mathcal{X}$ and any given $\hat{\rho}$.

First, if $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho}) > 0$, then according to (4.5), we have that $\max_{\boldsymbol{\delta}\in\mathbb{U}} \{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}\|_1\} > 0$, which implies that there exists a $\boldsymbol{\delta}^* \in \mathbb{U}$ with $F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}^*)) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}^*\|_1 > 0$. Thus,

$$\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) = \max_{\boldsymbol{\delta}\in\mathbb{U}} \frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z}}{\|\boldsymbol{\delta}\|_1} \geq \frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}^*)) - \mathcal{Z}}{\|\boldsymbol{\delta}^*\|_1} > \hat{\rho}$$

Second, if $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho}) < 0$, then we have that $\max_{\boldsymbol{\delta}\in\mathbb{U}} \{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}\|_1\} < 0$, which implies that $F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}\|_1 < 0, \quad \forall \boldsymbol{\delta} \in \mathbb{U}$. Thus,

$$\frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z}}{\|\boldsymbol{\delta}\|_1} < \hat{\rho}, \quad \forall \boldsymbol{\delta} \in \mathbb{U}.$$

Therefore, we obtain that

$$\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) = \max_{\boldsymbol{\delta}\in\mathbb{U}} \frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z}}{\|\boldsymbol{\delta}\|_1} < \hat{\rho}.$$

Third, if $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho}) = 0$, then we obtain that $\max_{\boldsymbol{\delta}\in\mathbb{U}} \{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}\|_1\} = 0$, which implies that there exists a $\boldsymbol{\delta}^* \in \mathbb{U}$ such that $F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}^*)) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}^*\|_1 = 0$, and $F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \hat{\rho}\|\boldsymbol{\delta}\|_1 \leq 0, \quad \forall \boldsymbol{\delta} \in \mathbb{U} \setminus \{\boldsymbol{\delta}^*\}$. Thus,

$$\frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}^*)) - \mathcal{Z}}{\|\boldsymbol{\delta}^*\|_1} = \hat{\rho}, \text{ and } \quad \frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z}}{\|\boldsymbol{\delta}\|_1} \leq \hat{\rho}, \text{ for all } \boldsymbol{\delta} \in \mathbb{U} \setminus \{\boldsymbol{\delta}^*\},$$

which implies that

$$\hat{\rho}^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) = \max_{\boldsymbol{\delta}\in\mathbb{U}} \frac{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z}}{\|\boldsymbol{\delta}\|_1} = \hat{\rho}.$$

Hence, Lemma 4.1 is proved.

B.2. Proof of Proposition 4.1

First, we need to establish Lemma B.1 below, which indicates that model LP($\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$) always has a feasible solution for each $(\boldsymbol{x},\boldsymbol{z})$ that satisfies constraints (3.17)–(3.24) and for each $\boldsymbol{\delta}\in\mathbb{U}$.

Lemma B.1 *For any $(\boldsymbol{x},\boldsymbol{z})$ that satisfies constraints (3.17)–(3.24), and for any $\boldsymbol{\delta}\in\mathbb{U}$, model LP($\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$) in (3.37)–(3.38) always has a feasible solution.*

Proof. For any given $(\boldsymbol{x},\boldsymbol{z})$ that satisfies constraints (3.17)–(3.24), it corresponds to a nominal timely-implementable flat solution $(\mathcal{P},\mathcal{C})$. Consider any $\boldsymbol{\delta}\in\mathbb{U}$ with the corresponding realized travel time $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$. For such $(\mathcal{P},\mathcal{C})$ and $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$, we first show as follows that there exists a departure schedule $\mathcal{T}$ such that constraints (2.1)–(2.4) are satisfied, from which we can then obtain a feasible solution to model LP($\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$).

For the nominal timely-implementable flat solution $(\mathcal{P},\mathcal{C})$, consider each commodity $k\in\mathcal{K}$ and its flat path P^k in $\mathcal{P}$ with an arc sequence denoted by $(a_1^k,...,a_{m^k}^k)$. For each $n\in\{1,2,\dots,m^k\}$, there must exist a consolidation $C_{r_n^k}^{a_n^k}\in\mathcal{C}$ for arc a_n^k with $r_n^k\in\{1,2,\cdots,|\mathcal{K}|\}$ such that $k\in C_{r_n^k}^{a_n^k}$. We can now construct a network $\mathcal{G}_{\mathcal{C}}=\{\mathcal{N}_{\mathcal{C}},\mathcal{A}_{\mathcal{C}}\}$ where each non-empty consolidation $C_r^\alpha\in\mathcal{C}$ corresponds to a node, denoted by $\langle\alpha,r\rangle$, in the node set $\mathcal{N}_{\mathcal{C}}$, and each pair of consolidations $C_{r_n^k}^{a_n^k}$ and $C_{r_{n+1}^k}^{a_{n+1}^k}$ for $k\in\mathcal{K}$ and $n\in\{1,...,m^k-1\}$ corresponds to an arc $(\langle a_n^k,r_n^k\rangle,\langle a_{n+1}^k,r_{n+1}^k\rangle)$ in the arc set $\mathcal{A}_{\mathcal{C}}$.

Since the flat solution $(\mathcal{P},\mathcal{C})$ is a nominal timely-implementable first-stage solution, there exists a departure schedule $\mathcal{T}$ which satisfies (2.1)–(2.4) with nominal travel times $\overline{\boldsymbol{\tau}}$. According to $\mathcal{T}$, for each consolidation $C_r^\alpha\in\mathcal{C}$ of arc $\alpha=(\nu,\nu')\in\mathcal{A}$ we can obtain its corresponding departure time from node ν, which is denoted by $t_{\alpha,r}$. For each pair of consolidations $C_{r_n^k}^{a_n^k}$ and $C_{r_{n+1}^k}^{a_{n+1}^k}$ with $k\in\mathcal{K}$ and $n\in\{1,...,m^k-1\}$, the departure time of $C_{r_n^k}^{a_n^k}$ from node ν_n^k plus the nominal value $\overline{\tau}_{a_n^k}$ of travel time of arc a_n^k must be less than or equal to the departure time of $C_{r_{n+1}^k}^{a_{n+1}^k}$ from node ν_{n+1}^k. Thus, by the definition of $\mathcal{G}_{\mathcal{C}}=\{\mathcal{N}_{\mathcal{C}},\mathcal{A}_{\mathcal{C}}\}$, we obtain that

$$t_{\alpha,r}+\overline{\tau}_\alpha\le t_{\alpha',r'},\ \forall\ (\langle\alpha,r\rangle,\langle\alpha',r'\rangle)\in\mathcal{A}_{\mathcal{C}}.$$

This, together with $\overline{\tau}_\alpha>0$ for all $\alpha\in\mathcal{A}$, implies that $\mathcal{G}_{\mathcal{C}}$ must be an acyclic network, and thus has a topological ordering of nodes in $\mathcal{N}_{\mathcal{C}}$, denoted by $(\langle\alpha_1,r_1\rangle,\langle\alpha_2,r_2\rangle,\dots,\langle\alpha_{|\mathcal{N}_{\mathcal{G}}|},r_{|\mathcal{N}_{\mathcal{G}}|}\rangle)$.

Next, consider each possible realized travel time $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$ with any $\boldsymbol{\delta}\in\mathbb{U}$. For $n=1,2,\dots,|\mathcal{N}_{\mathcal{G}}|$, we can set the departure time of consolidation $C_{r_n}^{\alpha_n}$, denoted by $\hat{t}_{\alpha_n,r_n}$, iteratively as follows: $\hat{t}_{\alpha_1,r_1}=\max_{k\in\mathcal{K}}e^k$ and $\hat{t}_{\alpha_n,r_n}=\hat{t}_{\alpha_{n-1},r_{n-1}}+\max_{(i,j)\in\mathcal{A}}\{\overline{\tau}_{ij}+\hat{\tau}_{ij}\}$ for $n=2,3,\dots,|\mathcal{N}_{\mathcal{C}}|$. Thus, it can be seen that for each commodity $k\in\mathcal{K}$, $\hat{t}_{\alpha_1^k,r_1^k}\ge\hat{t}_{\alpha_1,r_1}=\max\limits_{k\in\mathcal{K}}e^k\ge e^k$ and $\hat{t}_{\alpha_{n+1}^k,r_{n+1}^k}\ge\hat{t}_{\alpha_n^k,r_n^k}+\max\limits_{(i,j)\in\mathcal{A}}\{\overline{\tau}_{ij}+\hat{\tau}_{ij}\}\ge\hat{t}_{\alpha_n^k,r_n^k}+\tilde{\tau}_{\alpha_n^k}$ for $n=1,\dots,m^k-1$.

Thus, by setting the departure time of commodity k for node ν_n^k to be equal to $\hat{t}_{\alpha_n^k,r_n^k}$, for $n=1,2,\dots,m^k$ and $k\in\mathcal{K}$, we obtain a plan $\hat{\mathcal{T}}$ which satisfies the constraints (2.1), (2.2) and (2.4) under the travel time $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$. From such a departure schedule $\hat{\mathcal{T}}$, we can obtain the values of variables v_{ij}^k, b_{ijr}, w_i^k, and s^k according to their definitions, which form a feasible solution to model LP($\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$). Hence, Lemma B.1 is proved. □

Recall that $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ is defined by a linear program in (3.37)–(3.38). As in Section 4.1.1, let β_i^k, γ^k, ψ^k, η_{ij}^k, θ_{ijr}^k, ξ_{ijr}^k, and λ_i^k denote the dual variables associated with its constraints (3.26)–(3.32), respectively.

Let Ω indicate the feasible domain of its dual, which is a convex polyhedron defined by some linear constraints. By Lemma B.1 and the strong duality theorem, the optimal objective value of LP$(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}})$ equals that of its dual linear program.

Accordingly, we can replace the LP formulation of $F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ with its dual to reformulate the model defined in (4.5) for $G(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \hat{\rho})$ as the following nonlinear optimization model:

$$\begin{aligned} G(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \hat{\rho}) = \max\ & F_1(\boldsymbol{x}, \boldsymbol{y}) + \Big[\sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^k - \lambda_i^k) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^k - \lambda_i^k) \big) \cdot \tilde{\tau}_{jir} \\ & - \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} (M_1 x_{ij}^k) \cdot \eta_{ij}^k + \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} [M_1 (z_{ijr}^k - 1)] \cdot (\theta_{ijr}^k + \xi_{ijr}^k) \\ & + \sum_{k\in\mathcal{K}} e^k \cdot (\gamma^k - \lambda_{o^k}^k) + \sum_{k\in\mathcal{K}} l^k \cdot (\lambda_{d^k}^k - \psi^k) \Big] - \mathcal{Z} - \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \hat{\rho} |\delta_{ijr}| \qquad \text{(B.1)} \\ \text{s.t.}\quad & (\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \boldsymbol{\eta}, \boldsymbol{\theta}, \boldsymbol{\xi}, \boldsymbol{\lambda}) \in \Omega, \qquad \text{(B.2)} \\ & \tilde{\tau}_{ijr} = \overline{\tau}_{ij} + \hat{\tau}_{ij} \delta_{ijr}, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}, \qquad \text{(B.3)} \\ & -1 \le \delta_{ijr} \le 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}. \qquad \text{(B.4)} \end{aligned}$$

Next, to further prove Proposition 4.1, for any given $(\boldsymbol{x}, \boldsymbol{z})$ and $\hat{\rho}$, consider any optimal solution $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}^*, \boldsymbol{\delta}^*)$ of the optimization model defined in (B.1)–(B.4). By fixing $(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \boldsymbol{\eta}, \boldsymbol{\theta}, \boldsymbol{\xi}, \boldsymbol{\lambda}) = (\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*)$, the nonlinear optimization model defined in (B.1)–(B.4) reduces to the following nonlinear model on $\boldsymbol{\delta}$, denoted as model $\mathbf{S}_1$.

$$\begin{aligned} [\mathbf{S}_1]\ \max\quad & \sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \Big\{ \hat{\tau}_{jir} \Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k*} - \lambda_i^{k*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k*} - \lambda_i^{k*}) \Big) \cdot \delta_{jir} - \hat{\rho} \cdot |\delta_{jir}| \Big\} \\ s.t.\quad & -1 \le \delta_{ijr} \le 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\} \end{aligned}$$

We can see that $\boldsymbol{\delta}^*$ is an optimal solution of model $\mathbf{S}_1$. For any optimal solution $\hat{\boldsymbol{\delta}}$ of model $\mathbf{S}_1$, $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}(\hat{\boldsymbol{\delta}}), \hat{\boldsymbol{\delta}})$ forms a feasible solution of the optimization model in (B.1)–(B.4), and it has the same objective value as that of $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}^*, \boldsymbol{\delta}^*)$. Thus, $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}(\hat{\boldsymbol{\delta}}), \hat{\boldsymbol{\delta}})$ is also an optimal solution to the optimization model in (B.1)–(B.4).

Consider any optimal solution $\hat{\boldsymbol{\delta}}$ to model $\mathbf{S}_1$. Due to the optimality of $\hat{\boldsymbol{\delta}}$, it can be seen that for any $(j,i) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$, if $\hat{\delta}_{jir} > 0$, then $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k*} - \lambda_i^{k*} + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k*} - \lambda_i^{k*}) \Big) \ge 0$, and that if $\hat{\delta}_{jir} < 0$, then $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k*} - \lambda_i^{k*} + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k*} - \lambda_i^{k*}) \Big) \le 0$. This is because otherwise, $\hat{\boldsymbol{\delta}}$ cannot be an optimal solution to model $\mathbf{S}_1$, as we can increase its objective value by changing the sign of each $\hat{\delta}_{jir}$ with $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k*} - \lambda_i^{k*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k*} - \lambda_i^{k*}) \Big) \cdot \hat{\delta}_{jir} < 0$ to its opposite. Thus, we obtain that $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k*} - \lambda_i^{k*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k*} - \lambda_i^{k*}) \Big) \cdot \hat{\delta}_{jir} \ge 0$ for all $(j,i) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$. Accordingly, model $\mathbf{S}_1$ is equivalent to the following maximization LP, denoted as model $\mathbf{S}_2$:

$$\begin{aligned} [\mathbf{S}_2]\ \max\quad & \sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \Big\{ \Big[\Big| \hat{\tau}_{jir} \Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k*} - \lambda_i^{k*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k*} - \lambda_i^{k*}) \Big) \Big| - \hat{\rho} \Big] \cdot \delta_{jir}^+ \Big\} \\ s.t.\quad & 0 \le \delta_{ijr}^+ \le 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\} \end{aligned}$$

From any optimal solution $\boldsymbol{\delta}^{+}$ of model $\mathbf{S}_2$, we can derive an optimal solution of model $\mathbf{S}_1$ by setting $\delta_{jir} = \delta^{+}_{jir}$ if $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z^k_{jir}(\beta^{k*}_i - \lambda^{k*}_i + \sum_{k\in\mathcal{K}^d_i} z^k_{jir}(\psi^{k*} - \lambda^{k*}_i)\Big) \geq 0$, and setting $\delta_{jir} = -\delta^{+}_{jir}$ if $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z^k_{jir}(\beta^{k*}_i - \lambda^{k*}_i + \sum_{k\in\mathcal{K}^d_i} z^k_{jir}(\psi^{k*} - \lambda^{k*}_i)\Big) < 0$, for each $(j,i)\in\mathcal{A}$ and $r\in\{1,2,...,|\mathcal{K}|\}$, so that their objective function values are the same.

For model $\mathbf{S}_2$, its constraint matrix associated with $\delta^{+}_{ijr} \leq 1$ for all $(i,j)\in\mathcal{A}$ and $r\in\{1,2,...,|\mathcal{K}|\}$ is totally unimodular, as it contains only one entry of 1 in each column. Thus, the feasible solution region of model $\mathbf{S}_2$ is an integral polytope. There must exists an integral optimal solution to model $\mathbf{S}_2$ with $\delta^{+}_{ijr}\in\{0,1\}$ for each $(i,j)\in\mathcal{A}$ and $r\in\{1,2,...,|\mathcal{K}|\}$. This implies that there exists an optimal solution $\boldsymbol{\delta}$ to model $\mathbf{S}_1$ with $\delta_{ijr}\in\{-1,0,1\}$ for each $(i,j)\in\mathcal{A}$ and $r\in\{1,2,...,|\mathcal{K}|\}$. Such an solution $\boldsymbol{\delta}$ must also be optimal for the model in (B.1)–(B.4) and the model in (4.5).

Moreover, consider such an optimal solution $\boldsymbol{\delta}'$ to the model in (4.5) that satisfies $\delta'_{ijr}\in\{-1,0,1\}$ for all $(i,j)\in\mathcal{A}$ and $r\in\{1,2,...,|\mathcal{K}|\}$. Suppose there exist $(i,j)\in\mathcal{A}$ and $r\in\{1,2,\cdots,|\mathcal{K}|\}$ with $\sum_{k\in\mathcal{K}} h^k q^k \hat{\tau}_{ij} \leq \hat{\rho}$ and $\delta'_{ijr} = -1$. By changing only the value of δ'_{ijr} from -1 to 0, we can obtain a new scenario in $\mathbb{U}$, which is denoted by $\boldsymbol{\delta}''$. Under $\boldsymbol{\delta}''$, the travel time for the r-th consolidation through arc (i,j) is increased by $\hat{\tau}_{ij}$, resulting in the decrease of the total holding cost by $\sum_{k\in\mathcal{K}} h^k q^k \hat{\tau}_{ij}$ at maximum. This implies that $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}'')) - F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}')) \geq -\sum_{k\in\mathcal{K}} h^k q^k \hat{\tau}_{ij}$. Thus, noting that $\|\boldsymbol{\delta}''\|_1 - \|\boldsymbol{\delta}'\|_1 = -1$, it can be seen that the difference between the objective value of model in (4.5) under $\boldsymbol{\delta}''$ and that under $\boldsymbol{\delta}'$ is at least $\hat{\rho} - \sum_{k\in\mathcal{K}} h^k q^k \hat{\tau}_{ij}$, which, due to $\sum_{k\in\mathcal{K}} h^k q^k \hat{\tau}_{ij} \leq \hat{\rho}$, must be non-negative. Thus, $\boldsymbol{\delta''}$ must also be an optimal solution to the model in (4.5). By repeating this iteratively, we can obtain an optimal solution $\boldsymbol{\delta}$ to the model in (4.5) that satisfies both (i) and (ii) of Proposition 4.1. The proof is completed.

B.3. Proof of Proposition 4.2

By Proposition 4.1, constraints (B.4) can be replaced with $\delta_{ijr}\in\{-1,0,1\}$ for all $(i,j)\in\mathcal{A}$ and $r\in\{1,2,...,|\mathcal{K}|\}$, and with $\delta_{ijr}\in\{0,1\}$ for all $(i,j)\in\mathcal{A}(\hat{\rho})$ and $r\in\{1,2,...,|\mathcal{K}|\}$ where $\mathcal{A}(\hat{\rho}) = \{(i,j)\in\mathcal{A} : \sum_{k\in\mathcal{K}} h^k q^k \hat{\tau}_{ij} \leq \hat{\rho}\}$. By (B.3) we have that $\tilde{\tau}_{ijr}\in\{\overline{\tau}_{ijr} - \hat{\tau}_{ijr}, \overline{\tau}_{ijr}, \overline{\tau}_{ijr} + \hat{\tau}_{ijr}\}$, which, together with $\overline{\tau}_{ijr}\in\mathbb{N}_{>0}$, $\hat{\tau}_{ijr}\in\mathbb{N}_0$ and $\overline{\tau}_{ijr} > \hat{\tau}_{ijr}$, implies that $\tilde{\tau}_{ijr}\in\mathbb{N}_{>0}$. Moreover, let a new variable $\hat{\varphi}_{jir}$ represent each nonlinear term $\Big(\sum_{k\in\mathcal{K}_i} z^k_{jir}(\beta^k_i - \lambda^k_i) + \sum_{k\in\mathcal{K}^d_i} z^k_{jir}(\psi^k - \lambda^k_i)\Big)\cdot\tilde{\tau}_{jir} - \hat{\rho}|\delta_{ijr}|$. We replace each integer variable δ_{jir} with three new binary variables $\hat{\zeta}_{jir,-1}$, $\hat{\zeta}_{jir,0}$ and $\hat{\zeta}_{jir,1}$, which are used to indicate whether δ_{ijr} equals -1, 0 and 1, respectively. Let $\tilde{\tau}_{ijr,-1} = \overline{\tau}_{ijr} - \hat{\tau}_{ijr}$, $\tilde{\tau}_{ijr,0} = \overline{\tau}_{ijr}$ and $\tilde{\tau}_{ijr,1} = \overline{\tau}_{ijr} + \hat{\tau}_{ijr}$. Accordingly, the linear constraints (4.8)–(4.11) can be derived. Thus, the nonlinear optimization model in (B.1)–(B.4) for $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\hat{\rho})$ can be reformulated to the maximization MILP model shown in Proposition 4.2. The proof is completed.

B.4. Proof of Lemma 4.2

Consider any $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})\in\mathcal{X}$. By Lemma 4.1, if $\rho_l \leq \rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, we have that $\max_{\boldsymbol{\delta}\in\mathbb{U}}\{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \rho_l\|\boldsymbol{\delta}\|_1\} = G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l) \geq 0$. Note that $\boldsymbol{\delta}(\rho_l)$ indicates a realization of $\boldsymbol{\delta}$ such that $F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l))) - \mathcal{Z} - \rho_l\|\boldsymbol{\delta}(\rho_l)\|_1 = \max_{\boldsymbol{\delta}\in\mathbb{U}}\{F_1(\boldsymbol{x},\boldsymbol{y}) + F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})) - \mathcal{Z} - \rho_l\|\boldsymbol{\delta}\|_1\}$. Since

$\max_{\boldsymbol{\delta}\in\mathbb{U}}\{F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))-\mathcal{Z}-\rho_l\|\boldsymbol{\delta}\|_1\}\geq 0$, we obtain that $F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l)))-\mathcal{Z}-\rho_l\|\boldsymbol{\delta}(\rho_l)\|_1\geq 0$, which implies that

$$\rho_l'=\frac{F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l)))-\mathcal{Z}}{\|\boldsymbol{\delta}(\rho_l)\|_1}\geq\rho_l.$$

Hence, $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l')=\max_{\boldsymbol{\delta}\in\mathbb{U}}\{F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))-\mathcal{Z}-\rho_l'\|\boldsymbol{\delta}\|_1\}\geq F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l)))-\mathcal{Z}-\rho_l'\|\boldsymbol{\delta}(\rho_l)\|_1=0$. Thus, by Lemma 4.1, we obtain that $\rho_l'\leq\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. Lemma 4.2 is proved.

B.5. Proof of Theorem 4.1

To prove statement (i) of Theorem 4.1, consider each iteration n of Algorithm 1. Let $\rho_l^{(n)}$ denote the value of ρ_l updated in Step 4. Algorithm 1 solves $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l^{(n)})$ in Step 5, derives its optimal solution $\boldsymbol{\delta}(\rho_l^{(n)})$ of $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l^{(n)})$ by (4.12), and computes the value of $\rho_l'^{(n)}$ from $\boldsymbol{\delta}(\rho_l^{(n)})$ by $\rho_l'^{(n)}=(F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l^{(n)})))-\mathcal{Z})/\|\boldsymbol{\delta}(\rho_l^{(n)})\|_1$. If Algorithm 1 does not terminate at iteration n, then $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l^{(n)})>0$, which, together with Lemma 4.1, implies $\rho_l^{(n)}<\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. Thus, by Lemma 4.2, we have that

$$\rho_l^{(n)}<\rho_l'^{(n)}\leq\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}).\tag{B.5}$$

By the definition of $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l'^{(n)})$ and $\rho_l'^{(n)}$, we have that

$$G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l'^{(n)})\geq F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l^{(n)})))-\mathcal{Z}-\rho_l'^{(n)}\|\boldsymbol{\delta}(\rho_l^{(n)})\|_1=0.\tag{B.6}$$

Next, consider each iteration $m\geq n+1$. If Algorithm 1 does not terminate at iteration m, we have that

$$\rho_l^{(m)}\geq\rho_l'^{(n)},\tag{B.7}$$

$$G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l^{(m)})=F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l^{(m)})))-\mathcal{Z}-\rho_l^{(m)}\|\boldsymbol{\delta}(\rho_l^{(m)})\|_1>0,\tag{B.8}$$

$$F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l^{(n)})))-\mathcal{Z}-\rho_l^{(m)}\|\boldsymbol{\delta}(\rho_l^{(n)})\|_1\leq 0,\tag{B.9}$$

where (B.7) and (B.8) are implied by Step 4 of Algorithm 1 and Lemma 4.1, and (B.9) is implied by (B.6) and (B.7). By (B.5) and (B.7) we obtain that $\rho_l^{(n)}<\rho_l^{(m)}$. This, together with (B.8) and (B.9), implies that $\boldsymbol{\delta}(\rho_l^{(n)})$ and $\boldsymbol{\delta}(\rho_l^{(m)})$ are not equal.

By Proposition 4.1, each $\boldsymbol{\delta}$ derived by (4.12) satisfies that $\delta_{ijr}\in\{-1,0,1\}$ for all $(i,j)\in\mathcal{A}$ and $r\in\{1,2,\cdots,|\mathcal{K}|\}$. Therefore, as there are a finite number of such $\boldsymbol{\delta}$, Algorithm 1 must terminate in a finite number of iterations. Moreover, when Algorithm 1 terminates, we have that $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l)=0$. By Lemma 4.1, we obtain that Algorithm 1 returns $\rho_l=\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, and accordingly, $\boldsymbol{\delta}(\rho_l)$ is the corresponding worst-case scenario for $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$. Hence, the first statement of Theorem 4.1 is proved.

The statement (ii) of Theorem 4.1 follows directly from the property of bisection search. For any $\epsilon>0$, Algorithm 1 only needs at most $\lceil\log_2((\rho_h^{(0)}-\rho_l^{(0)})/\epsilon)\rceil$ iterations to ensure $\rho_h-\rho_l\leq\epsilon$. At each iteration of Algorithm 1, by Lemma 4.1 and Lemma 4.2 we note that $\rho_l\leq\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})\leq\rho_h$ and $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l)\geq 0$. Thus, after $\lceil\log_2((\rho_h^{(0)}-\rho_l^{(0)})/\epsilon)\rceil$ iterations, we have $\rho_l\leq\rho^*(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})\leq\rho_h\leq\rho_l+\epsilon$. By $G(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\rho_l)\geq 0$ we also have $(F_1(\boldsymbol{x},\boldsymbol{y})+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}(\rho_l)))-\mathcal{Z}\geq\rho_l\|\boldsymbol{\delta}(\rho_l)\|_1$. Hence, the statement (ii) of Theorem 4.1 is also proved.

B.6. Proof of Theorem 4.2

At each iteration of Algorithm 2, UB and LB are updated by solving the corresponding master problem and subproblem, while a new worst-case scenario $\boldsymbol{\delta}$ in $\mathbb{U}$ is obtained and added into the scenario subset Λ. Algorithm 2 stops when $UB = LB$.

First, we show that Algorithm 2 returns an optimal solution to model RS if it terminates with $UB = LB$. As model RSMILP(Λ) is a relaxation of model RS, the value of LB, which equals the optimal objective value of model RSMILP(Λ), is a valid lower bound on the optimal objective value of model RS. As UB is the worst-case normalized cost deviation of a first-stage solution $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}})$, it provides a valid upper bound on the optimal objective value of model RS. Thus, when $UB = LB$, $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}})$ forms an optimal solution to model RS.

Next, we show that Algorithm 2 must terminate with $UB = LB$ in a finite number of iterations. We note that at each iteration n, if the worst-case scenario $\boldsymbol{\delta}^{(n)}$ identified in Step 3 of Algorithm 2 is not in the current scenario subset Λ, it will be added to Λ. According to Proposition 4.1, $\boldsymbol{\delta}^{(n)}$ satisfies that $\delta_{ijr}^{(n)} \in \{-1, 0, 1\}$ for all $(i,j) \in \mathcal{A}$ and $r \in \{1, 2, \cdots, |\mathcal{K}|\}$, and has a finite number of possible values. Therefore, in a finite number of iterations, $\boldsymbol{\delta}^{(n)}$ identified in Step 3 of Algorithm 2 must be included in the current scenario set Λ. In such a situation, both LB and UB must be equal to the optimal objective value of the current master problem, implying that $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}})$ forms an optimal solution to model RS. This completes the proof of Theorem 4.2.

B.7. Proof of Proposition 4.3

Recall that $F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ is defined by a linear program in (3.37)–(3.38), which, by Lemma B.1, always has a feasible solution. Similar to our reformulation of $G(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \hat{\rho})$, we can replace the LP formulation of $F_{LP}(\boldsymbol{x}, \boldsymbol{z}, \tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ with its dual, thereby reformulating the inner minimization problem of the max-min model RP($\boldsymbol{x}, \boldsymbol{z}$) of $F_{RP}(\boldsymbol{x}, \boldsymbol{z})$ defined in (3.25)–(3.36). The reformulation takes the form of a nonlinear optimization model with a bi-linear objective function, as shown below:

$$
\begin{aligned}
F_{RP}(\boldsymbol{x}, \boldsymbol{z}) = \max \quad & \sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^k - \lambda_i^k) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^k - \lambda_i^k) \Big) \cdot \tilde{\tau}_{jir} \\
& - \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} (M_1 x_{ij}^k) \cdot \eta_{ij}^k + \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} [M_1 (z_{ijr}^k - 1)] \cdot (\theta_{ijr}^k + \xi_{ijr}^k) \\
& + \sum_{k\in\mathcal{K}} e^k \cdot (\gamma^k - \lambda_{o^k}^k) + \sum_{k\in\mathcal{K}} l^k \cdot (\lambda_{d^k}^k - \psi^k) \qquad \text{(B.10)} \\
\text{s.t.} \quad & (\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \boldsymbol{\eta}, \boldsymbol{\theta}, \boldsymbol{\xi}, \boldsymbol{\lambda}) \in \Omega, \qquad \text{(B.11)} \\
& \tilde{\tau}_{ijr} = \overline{\tau}_{ij} + \hat{\tau}_{ij} \delta_{ijr}, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}, \qquad \text{(B.12)} \\
& -1 \le \delta_{ijr} \le 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}, \qquad \text{(B.13)} \\
& \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} |\delta_{ijr}| \le \Gamma. \qquad \text{(B.14)}
\end{aligned}
$$

Proposition B.1 below indicates that the domain of each variable δ_{ijr} can be restricted to $\{-1, 0, 1\}$ without changing the optimal objective value of the nonlinear optimization model above.

Proposition B.1 *There exists an optimal solution to the nonlinear optimization model defined in (B.10)–(B.14) such that* $\delta_{ijr} \in \{-1, 0, 1\}$ *for each* $(i,j) \in \mathcal{A}$ *and* $r \in \{1, 2, ..., |\mathcal{K}|\}$.

Proof. For any given $(\boldsymbol{x}, \boldsymbol{z})$, consider any optimal solution $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}^*, \boldsymbol{\delta}^*)$ of the nonlinear optimization model defined in (B.10)–(B.14). By fixing $(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \boldsymbol{\eta}, \boldsymbol{\theta}, \boldsymbol{\xi}, \boldsymbol{\lambda}) = (\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*)$, the model defined in (B.10)–(B.14) reduces to the following nonlinear model on $\boldsymbol{\delta}$, denoted as model $\mathbf{R}_1$.

$$
\begin{aligned}
[\mathbf{R}_1] \quad \max \quad & \sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \Big\{ \hat{\tau}_{jir} \Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*}) \Big) \cdot \delta_{jir} \Big\} \\
s.t. \quad & -1 \le \delta_{ijr} \le 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}, \\
& \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} |\delta_{ijr}| \le \Gamma.
\end{aligned}
$$

Accordingly, $\boldsymbol{\delta}^*$ must be an optimal solution to model $\mathbf{R}_1$. Moreover, for any optimal solution $\hat{\boldsymbol{\delta}}$ to model $\mathbf{R}_1$, $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}(\hat{\boldsymbol{\delta}}), \hat{\boldsymbol{\delta}})$ forms a feasible solution to the nonlinear optimization model defined in (B.10)–(B.14), with the same objective value as that of $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}^*, \boldsymbol{\delta}^*)$. Thus, $(\boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \boldsymbol{\psi}^*, \boldsymbol{\eta}^*, \boldsymbol{\theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \tilde{\boldsymbol{\tau}}(\hat{\boldsymbol{\delta}}), \hat{\boldsymbol{\delta}})$ is also an optimal solution to the model defined in (B.10)–(B.14).

Consider any optimal solution $\hat{\boldsymbol{\delta}}$ to model $\mathbf{R}_1$. Due to the optimality of $\hat{\boldsymbol{\delta}}$, it can be seen that for any $(j,i) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$, if $\hat{\delta}_{jir} > 0$, then $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*} + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*})\Big) \ge 0$, and that if $\hat{\delta}_{jir} < 0$, then $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*} + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*})\Big) \le 0$. This is because otherwise, $\hat{\boldsymbol{\delta}}$ cannot be an optimal solution to model $\mathbf{R}_1$, as we can increase its objective value by changing the sign of each $\hat{\delta}_{jir}$ with $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*})\Big) \cdot \hat{\delta}_{jir} < 0$ to its opposite. Thus, we obtain that $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*}) + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*})\Big) \cdot \hat{\delta}_{jir} \ge 0$ for all $(j,i) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$. Accordingly, model $\mathbf{R}_1$ is equivalent to the following maximization LP, denoted as model $\mathbf{R}_2$:

$$
\begin{aligned}
[\mathbf{R}_2] \quad \max \quad & \sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \Big\{ \Big| \hat{\tau}_{jir} \Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*} + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*}) \Big) \Big| \cdot \delta_{jir}^+ \Big\} \\
s.t. \quad & \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \delta_{ijr}^+ \le \Gamma, \\
& 0 \le \delta_{ijr}^+ \le 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1, 2, ..., |\mathcal{K}|\}.
\end{aligned}
$$

From any optimal solution $\boldsymbol{\delta}^+$ to model $\mathbf{R}_2$, we can derive an optimal solution to model $\mathbf{R}_1$ by setting $\delta_{jir} = \delta_{jir}^+$ if $\hat{\tau}_{jir}\Big(\sum_{k\in\mathcal{K}_i} z_{jir}^k (\beta_i^{k^*} - \lambda_i^{k^*} + \sum_{k\in\mathcal{K}_i^d} z_{jir}^k (\psi^{k^*} - \lambda_i^{k^*})\Big) \ge 0$, and setting $\delta_{jir} = -\delta_{jir}^+$ otherwise, for each $(j,i) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$, so that their objective values are the same.

For model $\mathbf{R}_2$, its constraint matrix associated with $\sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \delta_{ijr}^+ \le \Gamma$ and $\delta_{ijr}^+ \le 1$ for all $(i,j) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$ is totally unimodular, as it contains two entries of 1 in each column. This implies that with an integral Γ, the feasible solution region of model $\mathbf{R}_2$ is an integral polytope. Thus, there exists an integral optimal solution to model $\mathbf{R}_2$ with $\delta_{ijr}^+ \in \{0, 1\}$ for each $(i,j) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$. This implies that there exists an optimal solution $\boldsymbol{\delta}$ to model $\mathbf{R}_1$ with $\delta_{ijr} \in \{-1, 0, 1\}$ for each $(i,j) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$. Therefore, there exists an optimal solution to the nonlinear optimization model defined in (B.10)–(B.14) that satisfies $\delta_{ijr} \in \{-1, 0, 1\}$ for each $(i,j) \in \mathcal{A}$ and $r \in \{1, 2, ..., |\mathcal{K}|\}$. Hence, Proposition B.1 is proved. $\square$

Continuing Proof of Proposition 4.3 We can now prove Proposition 4.3 as follows. By Proposition B.1, constraints (B.13) can be replaced with $\delta_{ijr} \in \{-1,0,1\}$ for all $(i,j) \in \mathcal{A}$ and $r \in \{1,2,...,|\mathcal{K}|\}$. By (B.12) we have that $\tilde{\tau}_{ijr} \in \{\overline{\tau}_{ijr} - \hat{\tau}_{ijr}, \overline{\tau}_{ijr}, \overline{\tau}_{ijr} + \hat{\tau}_{ijr}\}$, which, together with $\overline{\tau}_{ijr} \in \mathbb{N}_{>0}$, $\hat{\tau}_{ijr} \in \mathbb{N}_0$ and $\overline{\tau}_{ijr} > \hat{\tau}_{ijr}$, implies that $\tilde{\tau}_{ijr} \in \mathbb{N}_{>0}$. Moreover, we introduce a new variable φ_{jir} to represent each nonlinear term $\left(\sum_{k\in\mathcal{K}_i} z^k_{jir}(\beta^k_i - \lambda^k_i) + \sum_{k\in\mathcal{K}^d_i} z^k_{jir}(\psi^k - \lambda^k_i)\right) \cdot \tilde{\tau}_{jir}$. We then replace each integer variable δ_{jir} with three new binary variables $\zeta_{jir,-1}$, $\zeta_{jir,0}$ and $\zeta_{jir,1}$, which are used to indicate whether δ_{ijr} equals -1, 0 and 1, respectively. Let $\tilde{\tau}_{ijr,-1} = \overline{\tau}_{ijr} - \hat{\tau}_{ijr}$, $\tilde{\tau}_{ijr,0} = \overline{\tau}_{ijr}$ and $\tilde{\tau}_{ijr,1} = \overline{\tau}_{ijr} + \hat{\tau}_{ijr}$. Accordingly, the following linear constraints can be derived for the newly introduced variables, where M_3 is a sufficiently large constant.

$$\zeta_{ijr,-1} + \zeta_{ijr,0} + \zeta_{ijr,1} = 1, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \tag{B.15}$$

$$-M_3(1-\zeta_{jir,\ell}) \le \varphi_{jir} - \Big(\sum_{k\in\mathcal{K}_i} z^k_{jir}(\beta^k_i - \lambda^k_i) + \sum_{k\in\mathcal{K}^d_i} z^k_{jir}(\psi^k - \lambda^k_i)\Big)\tilde{\tau}_{jir,\ell} \le M_3(1-\zeta_{jir,\ell}),$$
$$\forall\ (j,i) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \ell = \{-1,0,1\}, \tag{B.16}$$

$$\zeta_{ijr,\ell} \in \{0,1\}, \quad \forall\ (i,j) \in \mathcal{A}, r \in \{1,2,...,|\mathcal{K}|\}, \ell = \{-1,0,1\}. \tag{B.17}$$

Here, constraints (B.15) ensure that exactly one of the three variables $\zeta_{jir,-1}$, $\zeta_{jir,0}$ and $\zeta_{jir,1}$ equals 1, constraints (B.16) ensure that each variable φ_{jir} equals $\left(\sum_{k\in\mathcal{K}_i} z^k_{jir}(\beta^k_i - \lambda^k_i) + \sum_{k\in\mathcal{K}^d_i} z^k_{jir}(\psi^k - \lambda^k_i)\right) \cdot \tilde{\tau}_{jir}$, and constraints (B.17) define the domain of the variables $\zeta_{ijr,-1}$, and $\zeta_{ijr,0}$, and $\zeta_{ijr,1}$. Thus, constraints $(B.14)$ can be replaced with the following linear constraint:

$$\sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} (\zeta_{ijr,-1} + \zeta_{ijr,1}) \le \Gamma. \tag{B.18}$$

Therefore, the nonlinear optimization model defined by (B.10)–(B.14) for $F_{RP}(\boldsymbol{x},\boldsymbol{z})$ can be further reformulated to the following maximization MILP model:

$$\begin{aligned} F_{RP}(\boldsymbol{x},\boldsymbol{z}) = \max \quad & \sum_{(j,i)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} \varphi_{jir} - \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} (M_1 x^k_{ij}) \cdot \eta^k_{ij} + \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} \sum_{r=1}^{|\mathcal{K}|} [M_1(z^k_{ijr} - 1)] \cdot (\theta^k_{ijr} + \xi^k_{ijr}) \\ & + \sum_{k\in\mathcal{K}} e^k \cdot (\gamma^k - \lambda^k_{o^k}) + \sum_{k\in\mathcal{K}} l^k \cdot (\lambda^k_{d^k} - \psi^k) \\ \text{s.t.} \quad & (\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\psi},\boldsymbol{\eta},\boldsymbol{\theta},\boldsymbol{\xi},\boldsymbol{\lambda}) \in \Omega,\ \text{(B.15) – (B.17) and (B.18).} \end{aligned}$$

Hence, Proposition 4.3 is proved. $\square$

Appendix C: Adapting Solutions to Dynamic Uncertainty Revelation

The two-stage formulation of the robust CTSNDP introduced in Section 2.2 assumes that before the actual departure schedule is determined in the second-stage, all actual travel times are revealed. In Remark 2.1, we claim that this two-stage formulation can be adapted to cases under the dynamic uncertainty revelation. We will now explain why this claim is correct.

Consider any first-stage solution $(\mathcal{P},\mathcal{C})$ of a routing plan and a consolidation plan. Under dynamic uncertainty revelation, the actual travel time of each consolidated shipment on an arc $(i,j) \in \mathcal{A}$ is only revealed after its arrival at node j. One possible approach to determine the departure schedule is to apply a *reactive policy* in which for each $(i,j)\in\mathcal{A}$, every consolidated shipment on arc (i,j) departs from node i as soon as all the commodities for the shipment have arrived at node i. Consequently, the departure times of consolidated shipments depend only on the actual travel times realized in the previous part of their transport, and not on the unrevealed future travel times.

Consider any possible scenario $\boldsymbol{\delta}$, which determines travel times $\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})$. We now show that the departure schedule obtained by the reactive policy introduced above achieves the minimum second stage cost under $\boldsymbol{\delta}$. On the one hand, the reactive policy ensures that for each arc $(i,j)\in\mathcal{A}$, every consolidated shipment on arc (i,j) departs from node i as soon as all the commodities for the shipment have arrived at node i, ensuring that all commodities arrive at their destinations at the earliest possible time. Therefore, the total delay penalty must be minimized. On the other hand, for each commodity k, let $\tilde{T}_k(\mathcal{P},\mathcal{C},\boldsymbol{\delta})$ indicate the total travel time of commodity k under $(\mathcal{P},\mathcal{C})$ and $\boldsymbol{\delta}$. Under the reactive policy, it can be seen that the total in-storage holding time equals $\max\{l_k-e_k-\tilde{T}_k(\mathcal{P},\mathcal{C},\boldsymbol{\delta}),0\}$, which achieves the minimum total in-storage holding time. Thus, the reactive policy achieves a minimum total in-storage holding cost that equals $h^k\max\{l_k-e_k-\tilde{T}_k(\mathcal{P},\mathcal{C},\boldsymbol{\delta}),0\}$.

Hence, the departure schedule derived from the reactive policy achieves the minimum total second-stage cost for each possible $\boldsymbol{\delta}$. Thus, solutions to the RO-CTSNDP and the RS-CTSNDP can be adapted by the reactive policy to cases under the dynamic uncertainty revelation without increasing their objective values.

Appendix D: Models SP and MM

Given a set Π of scenarios assumed to be uniformly distributed, we can formulate a stochastic programming model as follows, aiming to minimize the expected total cost over all possible scenarios in Π.

$$[\text{SP}] \quad \min_{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\overline{\boldsymbol{v}},\overline{\boldsymbol{b}})\in\mathcal{X}} \ \sum_{\boldsymbol{\delta}\in\Pi}\frac{1}{|\Pi|}\Big(\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}}(c_{ij}^k q^k)x_{ij}^k+\sum_{(i,j)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|}f_{ij}y_{ijr}+F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))\Big)$$

It utilizes the second-stage cost $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$, which is defined by a minimization LP in (3.37)-(3.38) for our models RO and RS. As a result, model SP is a minimization MILP.

Similarly, given Π we can formulate a min-max optimization model as follows, aiming to minimizing the worst-case total cost over all possible scenarios in Π.

$$[\text{MM}] \quad \min_{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\overline{\boldsymbol{v}},\overline{\boldsymbol{b}})\in\mathcal{X}}\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}}(c_{ij}^k q^k)x_{ij}^k+\sum_{(i,j)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|}f_{ij}y_{ijr}+\ \max_{\boldsymbol{\delta}\in\Pi}F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})).$$

Model MM can be further reformulated as follows:

$$\begin{aligned}\min_{\Phi,\ (\boldsymbol{x},\boldsymbol{y},\boldsymbol{z},\overline{\boldsymbol{v}},\overline{\boldsymbol{b}})\in\mathcal{X}} \ &\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}}(c_{ij}^k q^k)x_{ij}^k+\sum_{(i,j)\in\mathcal{A}}\sum_{r=1}^{|\mathcal{K}|}f_{ij}y_{ijr}+\Phi\\ \text{s.t.}\ \ &\Phi\ge F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta})),\ \forall\boldsymbol{\delta}\in\Pi.\end{aligned}$$

It can be seen that the above reformulation of model MM is a minimization MILP, because $F_{LP}(\boldsymbol{x},\boldsymbol{z},\tilde{\boldsymbol{\tau}}(\boldsymbol{\delta}))$ is defined by a minimization LP in (3.37)-(3.38).