The Pickup and Delivery Problem with Time Windows and Incompatibility Constraints in Cold Chain Transportation

Faheng Deng, Hu Qin, Jiliu Li
School of Management, Huazhong University of Science and Technology, Wuhan 430074, China
{dengfaheng@hotmail.com, tigerqin1980@qq.com, lijiliu@hust.edu.cn}

Chun Cheng*
Institute of Supply Chain Analytics, Dongbei University of Finance and Economics, Dalian 116025, China
*Corresponding author. chun.cheng@polymtl.ca

This study investigates a new variant of the pickup and delivery problem with time windows (PDPTW) applied in cold chain transportation, which quantifies the effect of time on the quality of perishable products. Multiple commodities with incompatibility constraints are considered, where some types of products cannot be transported in a vehicle simultaneously due to their different properties and requirements for storage temperatures. The aim is to determine vehicles' pickup and delivery routes as well as their departure times from the depot such that the travel cost and refrigeration cost of vehicles and the quality decay cost of products are minimized. We formulate this problem as a set partitioning model, which is solved exactly by a tailored branch-and-price (B&P) algorithm. To tackle the asymmetry issue arising from the pricing problem of the B&P framework, we develop a novel asymmetric bidirectional labeling algorithm. Benchmark instance sets based on real-world statistical data and classic PDPTW instance sets are first generated for this problem. Numerical results show that our B&P algorithm can solve most instances to optimality in an acceptable time frame. Moreover, our results demonstrate that integrating the refrigeration and quality decay costs into the objective function can significantly lower the total cost of cold chain transportation activities, compared to the widely adopted objective function minimizing only the travel cost.

Key words: cold chain transportation; perishable product; pickup and delivery; incompatibility constraints; branch-and-price

History: Accepted to Transportation Science on July 9, 2022

1. Introduction

Cold chain transportation (CCT) refers to the transportation process of temperature-sensitive products like fruits and vegetables along the supply chain, which plays a significant role in cold chain logistics. In China, we have witnessed the growing popularity of CCT over the past few years, e.g., the fresh food logistics of JD.com and Hema Fresh of Alibaba, which is mainly implemented in two modes. One is the trunk transportation with a relatively long time horizon like three days or more, and the other is the urban distribution that often needs to be completed within a day. Unlike normal-temperature transportation, CCT involves more complex cost components, that is, besides the travel cost, we also need to consider the refrigeration cost resulting from using refrigerated vehicles. In addition to supporting vehicle movement, the energy of a refrigerated vehicle is also consumed by the cooling machine to maintain an appropriate temperature range for carried products. Refrigeration costs could be very high since the continuous operation of the cooling machine requires lasting power inputs until the transportation process is over. Data from vehicle manufacturers show that the fuel consumption per 100 kilometers of a medium-size refrigerated vehicle is up to one-and-a-half times that of its normal-temperature counterpart. For an electric refrigerated vehicle, the recharge mileage with the cooling machine on is also significantly shorter than with the cooling machine off.

In CCT, besides vehicle-related costs, the quality decay cost of perishable products is also a key factor. In China, the cost associated with the quality loss of fruits and vegetables is up to 100 billion RMB per year (Chai 2016). In long-distance trunk transportation, vegetables and fruits are liable to decay due to delays, causing economic losses to suppliers. For example, one type of vegetables with a purchase price of 4 RMB/kg can be sold at 5 RMB/kg when fresh. However, once their freshness reduces and staleness is discernible, they may only be sold at 4.2 RMB/kg. As time goes on, their values continue to decline and may become lower than the purchase price. Public data show that the deterioration rate of perishable products is about 10% or more, and some even reach 30% during the transportation process (Cnwest 2015). The qualities of perishable commodities affect not only suppliers' costs but also customers' satisfaction, especially those who shop online. Thus, effective operations of CCT are significant to the success of fresh food e-commerce.

Due to the rapid expansion of e-commerce and the improved consumption level, multi-product transportation is another challenge faced by cold chain logistics companies. First, it is difficult to guarantee commodity qualities while simultaneously transporting multiple products with different optimal storage temperatures (Stellingwerf et al. 2021). For example, at -20 degrees Celsius, the nutrition value of vegetables will be damaged while pork can be preserved well. Table 1 summarizes the temperature requirements of different products in cold chain logistics. We can observe that different temperature ranges are associated with different commodities, and thus refrigerated vehicles with different temperatures are required. Second, incompatibilities could arise due to different product properties, e.g., food and chemical products cannot usually be transported together. Therefore, incompatibility constraints should also be taken into account when performing CCT activities. Note that not all products are perishable in a cold chain system; for example, frozen beef can be preserved well for around three months. Thus, no quality decay cost is associated with stable products as their shelf lives far exceed the time horizon, which indicates that we also need to consider the pickup/delivery priority of products based on their shelf lives, in order to minimize the total cost of a CCT system.

Table 1 **The temperature requirements of different products in cold chain logistics (Shanghai Municipal Quality and Technical Supervision Bureau 2007)**

Temperature requirement	Product category
$0°C \sim 4°C$	Chilled poultry meat, aquatic product, non-dairy cream cake, juice, and yogurt
$0°C \sim 7°C$	Vegetable, fruit, pasteurized milk, refrigerated egg, egg liquid, and cooked meat
$\leq -18°C$	Frozen poultry meat, frozen aquatic product, frozen beverage, and quick-frozen vegetable
$-23°C \sim -25°C$	Ice cream and popsicle
$\leq -50°C$	Tuna fish

1.1. Our Contributions

This paper studies a pickup and delivery problem with time windows and incompatibility constraints (PDPTW-IC) in CCT, which is motivated by the practical transportation activities of a Chinese third-party logistics (3PL) company. This company runs refrigerated vehicles to serve transportation requests during a time horizon. Each request includes an origin-destination pair, where a given type of product is first picked up from the origin and then delivered to the destination. Each request is associated with one type of product, and there are multiple requests with different (perishable or stable) products. Requests with perishable products should be served as soon as possible to control the quality decay cost. However, requests with stable products are only required to be served within the time windows. We determine vehicles' pickup and delivery routes and their departure times from the depot to minimize the total cost. We consider our work makes the following contributions to the literature:

(1) We introduce a new variant of the pickup and delivery problem with time windows (PDPTW), i.e., PDPTW-IC, where we explicitly consider the incompatibility constraints among multiple commodities, which commonly exist in real-world CCT activities whereas are often ignored in prior studies. Based on the characteristics of CCT, we propose a comprehensive objective function, including the travel cost, the refrigeration cost, and the quality decay cost.

(2) We construct a set partitioning model for PDPTW-IC, which is solved by a tailored branch-and-price (B&P) algorithm. To tackle the asymmetry issue arising from the pricing problem of the B&P framework, we develop a novel asymmetric bidirectional labeling algorithm and prove the corresponding dominance rules. To the best of our knowledge, this paper develops the first exact algorithm for solving routing problems in cold chain logistics.

(3) We generate benchmark instance sets based on real-world statistical data and known instance sets in the literature, which will be available to the research community and thus motivate more algorithmic contributions for this type of problem.

(4) We conduct extensive numerical tests to evaluate the proposed B&P algorithm and the new objective function. Results show that our algorithm can solve most instances efficiently.

Moreover, the new objective function can significantly lower the total cost, compared to the widely adopted objective function minimizing only the travel cost.

The rest of this paper is structured as follows. Section 2 reviews related literature. Section 3 describes the problem and constructs the mathematical model. Section 4 introduces the bidirectional labeling algorithm for the pricing problem. Section 5 presents the branching strategy. Section 6 reports computational results, which is followed by conclusions in Section 7.

2. Literature Review

This section reviews related studies on PDPTW and CCT. A summary of papers on CCT is presented in Table 2.

The PDPTW has many applications, e.g., door-to-door passenger transportation, one-to-one freight transportation, and urban courier services, which can be formulated as a two-index model (Ropke et al. 2007, Furtado et al. 2017), a three-index model (Cordeau 2006, Ropke and Cordeau 2009), and a set partitioning model (Ropke and Cordeau 2009, Baldacci et al. 2011a). Several exact methods have been proposed to solve the PDPTW model, including the branch-and-cut algorithm (Cordeau 2006, Ropke et al. 2007, Factorovich et al. 2020), the branch-price-and-cut (BPC) algorithm (Ropke and Cordeau 2009, Cherkesly et al. 2015, Veenstra et al. 2017, Bettinelli et al. 2019), and set partitioning formulation-based algorithms (Baldacci et al. 2011a). Costa et al. (2019) provide a broad overview of BPC algorithms applied to vehicle routing problems (VRPs), where a subsection reviews related contributions on the PDPTW. Pessoa et al. (2020) propose a BPC solver incorporating the best techniques in the literature for generic VRPs. To the best of our knowledge, the authors have developed the state-of-the-art exact algorithm for the PDPTW. Heuristic methods are also developed, e.g., the adaptive large neighborhood search (Pisinger and Ropke 2007), the simulated annealing algorithm (Li and Lim 2008, Wang et al. 2015), the hybrid discrete particle swarm optimization method (Goksal et al. 2013), and the tabu search algorithm (Goeke 2019). For more details of PDPTW and its variants, see the review paper by Parragh et al. (2008a,b), Cordeau et al. (2008) and Qin et al. (2021).

Incompatibility constraints are common in real-world transportation problems. For example, foods and chemical products should be transported separately, and hazardous materials might dangerously react if presented in the same vehicle. To tackle the incompatibility issue, companies can consider utilizing heterogeneous dedicated vehicles to serve conflictive requests. However, this may require an enormous fleet size, especially when the number of requests is large, leading to a waste of resources. In the literature, some authors have incorporated incompatibility constraints into VRPs. For example, Manerba and Mansini (2015) consider the multi-vehicle traveling purchaser problem with pairwise incompatibility constraints (MVTPP-PIC), where multiple pairs of products cannot be transported on a vehicle simultaneously. A branch-and-cut algorithm is developed to solve the problem. Gendreau et al. (2016) study a special case of MVTPP-PIC, where the demand for each product is unitary. The authors develop a column generation (CG) approach for the problem. Factorovich et al. (2020) present a one-to-one pickup and delivery problem with only a single vehicle. In this case, the authors add an extra incompatibility constraint to impose that incompatible pairs of requests cannot be served by the vehicle simultaneously. Ceselli et al. (2009) use a CG algorithm to solve the rich VRP with incompatibility constraints. Battarra et al. (2009) consider a multi-trip VRP with incompatibility constraints. They minimize the number of used vehicles and develop an iterative algorithm.

In recent years, cold chain transportation of perishable goods has received increasing attention in the context of VRPs. Focusing on a single type of perishable product, Hsu et al. (2007) extend the vehicle routing problem with time windows (VRPTW) by considering stochastic travel times affecting the spoilage rate of products. Osvald and Stirn (2008) consider the distribution of fresh vegetables, where perishability is described by a linear model. The problem is modeled as a VRPTW with time-dependent travel times, which is solved by a tabu search-based heuristic approach. Based on the minimum loss of perishable goods, Qi and Hu (2020) study a VRP in emergency cold chain logistics, where fuel costs, refrigeration costs, and cargo damage costs are considered. The authors assume that refrigeration costs and cargo damage costs increase linearly with time. A hybrid heuristic algorithm is developed. Stellingwerf et al. (2021) focus on a quality-driven VRP in fresh food transportation. They propose a time- and temperature-dependent kinetic model to describe the quality decay realistically, where the temperature is an essential factor affecting product quality. The authors consider four types of objectives (i.e., minimizing product decay, carbon dioxide (CO_2) emissions, cost, and maximum decay) and compare their impacts on product quality.

In another line of research, some authors have studied multi-product distribution in cold chain logistics. For example, Chen et al. (2009) unify the production scheduling problem and the VRPTW in a single framework, where suppliers manufacture perishable products and sell to retailers with stochastic demands. Products are assumed to deteriorate at constant rates once manufactured. Different products have different deterioration rates, so those with lower decay rates are scheduled to be manufactured earlier. Product prices depend on the time differences between manufacturing and delivery. The authors formulate the problem as a nonlinear programming model, where the production scheduling part is solved by a constrained Nelder–Mead method, and the distribution part is solved by a heuristic algorithm. Amorim and Almada-Lobo (2014) construct a multi-objective VRP model to measure the relationship between distribution

Table 2 A summary of cold chain transportation related works

Authors	Problem	Component of objective function	Commodity	Incompatibility constraints?	Approach for quality decay	Departure time as decision variable?	Solution method
Hsu et al. (2007)	VRPTW	Fixed cost; transportation cost; inventory cost; energy cost; penalty cost	Single	Not applicable	Randomly decrease over time	Yes	Heuristic
Osvald and Stirn (2008)	VRPTW	Travel distance; travel time; delay cost; quality decay cost	Single	Not applicable	Linear decay over time	No	Heuristic
Qi and Hu (2020)	VRPTW	Fuel cost; refrigeration cost; cargo damage cost	Single	Not applicable	Linear decay over time	No	Heuristic
Stellingwerf et al. (2021)	VRP	Product decay; CO_2 emissions; transportation cost; maximum decay	Single	Not applicable	Time- and temperature-dependent kinetic model	No	Commercial solver
Chen et al. (2009)	Production scheduling and VRPTW	Total profit	Multiple	No	Product-specific decay rate	No	Heuristic
Amorim and Almada-Lobo (2014)	VRPTW	Distribution cost; freshness of product	Multiple	No	Linear decay over time	No	Heuristic
Song and Ko (2016)	VRP	Customer satisfaction	Multiple	No	Product-specific decay rate	No	Heuristic
This paper	PDPTW	Travel cost; refrigeration cost; quality decay cost	Multiple	Yes	Linear decay over time	Yes	Exact

plans and cost-freshness trade-offs. The authors assume that the shelf lives of perishable products decrease linearly with time. They develop an ϵ-constraint method and a hybrid evolutionary approach to solve small-size and large-size instances, respectively. Song and Ko (2016) address a VRP for food delivery using both refrigerated and general-type vehicles. The objective is to maximize customer satisfaction, which depends on the freshness of delivered products. When starting a trip, the freshness of each product is assumed to be perfect, whereas it reduces over elapsed travel times. The products delivered by refrigerated vehicles have higher levels of freshness than those delivered by general-type vehicles. To solve the problem, the authors derive a priority-based heuristic algorithm. Numerical results confirm that refrigerated vehicles have better performance in perishable product delivery compared to general-type vehicles.

Based on reviewed papers and Table 2, we find that (1) although multi-product delivery in CCT is studied by some authors, the incompatibility constraints among products are often ignored, whereas these constraints commonly exist in real-world CCT activities; (2) heuristic algorithms are often used to solve routing problems in CCT. To the best of our knowledge, no exact algorithm has been developed.

3. Problem Description and Mathematical Formulation

In this section, we describe the problem and introduce the quality decay function. We then construct a set partitioning model for PDPTW-IC.

3.1. Problem Description

Our problem is defined on a complete directed graph $G = (V, A)$, where $V = \{0, \ldots, 2n+1\}$ is the set of nodes. Nodes 0 and $2n+1$ denote the starting and ending depots, while subsets $V_p = \{1, \ldots, n\}$ and $V_d = \{n+1, \ldots, 2n\}$ denote pickup and delivery nodes, respectively. Each request i is associated with a pickup node i and a delivery node $i+n$. $A = \{(i, j) : \forall i \in V \setminus \{2n+1\}, j \in V \setminus \{0\}, i \neq j\}$ is the set of arcs. A fleet of $|K|$ identical refrigerated vehicles with capacity Q is available. We define customer set as $V_c = V_p \cup V_d$. Each customer has a demand q_i ($q_i > 0, \forall i \in V_p$ and $q_i = -q_{i-n}, \forall i \in V_d$) and must be visited exactly once. The demands at depots are $q_0 = q_{2n+1} = 0$. Each node $i \in V$ is associated with a hard time window $[e_i, l_i]$, where e_i and l_i are the earliest and latest times to start the service at node i, respectively. We let $e_0 = e_{2n+1} = 0$, $l_0 = l_{2n+1} = T$, where T is the time horizon. Each arc $(i, j) \in A$ is associated with a travel distance d_{ij} and a travel time σ_{ij}, where the latter also includes the service time at node i. The service times at the depots are set to 0. We assume that travel times satisfy the triangle inequality and that travel cost and refrigeration cost are proportional to travel distance and time, respectively. Let ρ_0 be the travel cost per unit distance and ρ_1 be the refrigeration cost per unit time. For each request, only one type of commodity is involved, but this type of commodity may be included in multiple requests. Let φ_i be the commodity type included in request i.

Different from traditional PDPTWs, our problem requires that commodities transported on a vehicle must be compatible at any time of the transportation process—traveling on the roads or waiting at customers. We introduce operators $\oplus$ and $\ominus$ to respectively denote compatibility and incompatibility between two commodities or a commodity and a set. For example, $a \oplus b$ represents that a and b are compatible. Besides vehicle capacity, time window, and incompatibility constraints, a feasible route r also needs to satisfy the pairing and precedence constraint, i.e., route r can visit the delivery node $i+n$ only after visiting the pickup node $i \in V_p$. We can then remove arcs $(i+n, i), \forall i \in V_p$ from set A.

3.2. Quantification of Quality Decay

In cold chain logistics, although there are many approaches to measure the quality decay, most are too complicated and require a large number of input parameters, which are often unavailable in the transportation process. In this study, we extend the linear model proposed by Osvald and Stirn (2008) to estimate the quality decay, as their model requires few parameters. Moreover, the authors indicate that a loss of 20% of quality can be related to a transportation process, denoting three cases: (1) 20% of transported quantity is completely decayed, and 80% is in perfect conditions; (2) the whole transported quantity is evenly decayed so that a product could be sold only at 80% of its original price; or (3) a combination of the first two cases. For completely decayed products, they are often disposed of without any return; thus, case (1) is equivalent to selling all quantities at 80% of the original prices. Based on these backgrounds, we assume that the quality decay of perishable products leads to a decrease in their market price. This price decrease depends on four parameters: ς', ς'', P', and P'', as represented by Figure 1, where the limited lifespan of each perishable product is divided into three stages. At the moment of harvest (point ς), perishable products are in perfect condition. From time ς to time ς', although the quality is reducing, there are no discernible changes; thus, a product can be sold at a good price P' at this stage. The quality decay continues and becomes visible in the second stage, causing the selling price to drop linearly with time. At time ς'', the product can only be sold at a very low price P''. After time ς'', the product becomes unacceptable and will be disposed of without any return. We consider the time duration $\varsigma' - \varsigma$ as the *fresh period* of products because there are no discernible changes. From time ς to time e_0, harvested products are collected and taken to pickup customers, and we consider this time duration as the *harvest period*. When everything is ready, our pickup and delivery process begins at e_0. Note that if the *harvest period* is shorter than the *fresh period*, products have no discernible changes from e_0 to ς'.

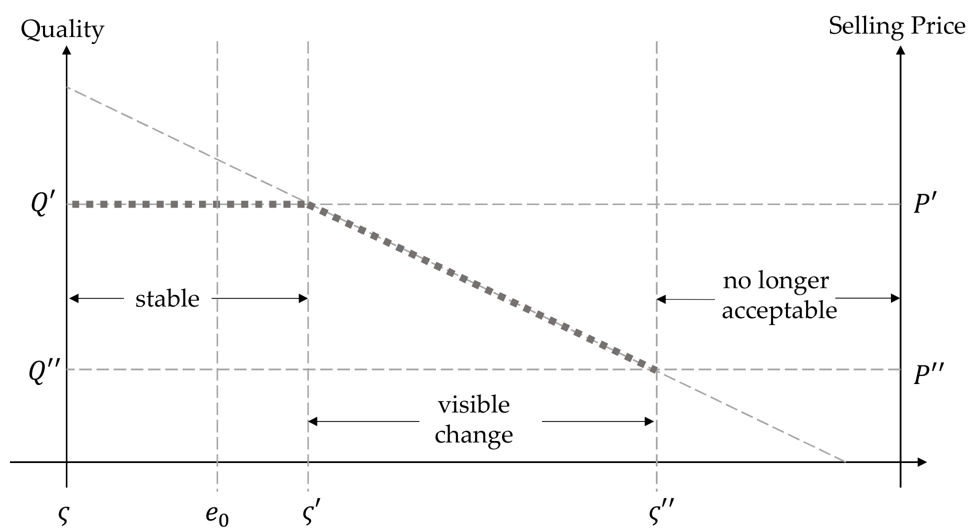

Figure 1 A quality decay model extended from Osvald and Stirn (2008)

Although cold chain logistics has a very high requirement of transportation timeliness, in real-world applications, perishable products are seldom delivered to customers before time ς'. And it is a common phenomenon that products decay visibly before delivery. Moreover, once a product becomes unacceptable, it might be disposed of directly and will not be delivered to customers anymore. Thus, we assume that the pickup and delivery process ends before time ς''. In addition, as mentioned before, not all cold chain products experience quality loss, i.e., the qualities of some products are stable under refrigerated conditions. Thus, no quality decay cost is associated with stable products. We define $\mathcal{P}$ as the set of all product types, among which some types of products are perishable, denoted as set $\overline{\mathcal{P}}$. Then the rest products in set $\mathcal{P} \setminus \overline{\mathcal{P}}$ are stable. Let t_i be the service starting time (SST) at node i and t_0 be the departure time of a vehicle from depot 0. The quality decay cost $\Lambda(i)$ of request i is calculated as

$$\Lambda(i) = q_i \frac{\max\{t_{i+n} - \varsigma'_{\varphi_i}, 0\}}{\varsigma''_{\varphi_i} - \varsigma'_{\varphi_i}} (P'_{\varphi_i} - P''_{\varphi_i}) \quad \forall \varphi_i \in \overline{\mathcal{P}},\ i \in V_p, \tag{1}$$

where $\varsigma''_{\varphi_i} - \varsigma'_{\varphi_i}$ represents the time span of visible changes of commodity φ_i, and P'_{φ_i} and P''_{φ_i} are the commodity's market prices at times ς'_{φ_i} and ς''_{φ_i}, respectively. If $\varphi_i \in \mathcal{P} \setminus \overline{\mathcal{P}}$, then no decay cost is associated with request i. For notational simplicity, we let

$$\pi_{i+n} = q_i \frac{P'_{\varphi_i} - P''_{\varphi_i}}{\varsigma''_{\varphi_i} - \varsigma'_{\varphi_i}}, \quad \forall \varphi_i \in \overline{\mathcal{P}},\ i \in V_p. \tag{2}$$

If $\varphi_i \in \mathcal{P} \setminus \overline{\mathcal{P}}$, we set $\pi_{i+n} = 0$.

Figure 2 illustrates a feasible solution to the PDPTW-IC, where $\varphi_3, \varphi_4 \in \overline{\mathcal{P}}$ and $\varphi_1, \varphi_2 \in \mathcal{P} \setminus \overline{\mathcal{P}}$. Meanwhile, φ_1 and φ_2 are incompatible. For each customer, the time-related parameters are given in the square brackets, where the first two numbers denote the time window, and the third number is the SST at that customer. For the ending depot 9, the last two numbers in the square bracket respectively denote the returning times of vehicles 1 and 2. Other parameters are omitted here because the main purpose of this example is to demonstrate the special features of our problem. As $\varphi_1 \ominus \varphi_2$, vehicle 1 can serve request 2 only after finishing request 1. Since no quality decay cost is associated with these two requests, vehicle 1 should leave depot 0 as late as possible to reduce the waiting times at customers and thus minimize the refrigeration cost. In contrast, as $\varphi_3 \oplus \varphi_4$, vehicle 2 can first visit all the pickup customers and then the delivery customers as long as other side constraints can be satisfied. Since φ_3 and φ_4 are perishable, vehicle 2 should leave depot 0 as early as possible to control the quality decay cost; however, to minimize the refrigeration cost, vehicle 2 is expected to leave the depot as late as possible. To balance these two types of cost and minimize their sum, the optimal departure time for vehicle 2 is 8:43 in this example. We will show how to deal with these two conflicting objective components in Section 4.

3.3. A Set Partitioning Model

The PDPTW-IC consists of designing vehicle routes and determining the departure time of each route from the depot to serve customers, such that the travel cost, refrigerated cost, and quality decay cost are minimized. To solve PDPTW-IC instances efficiently, this section introduces a set partitioning model, based on which a B&P algorithm is developed in the next section.

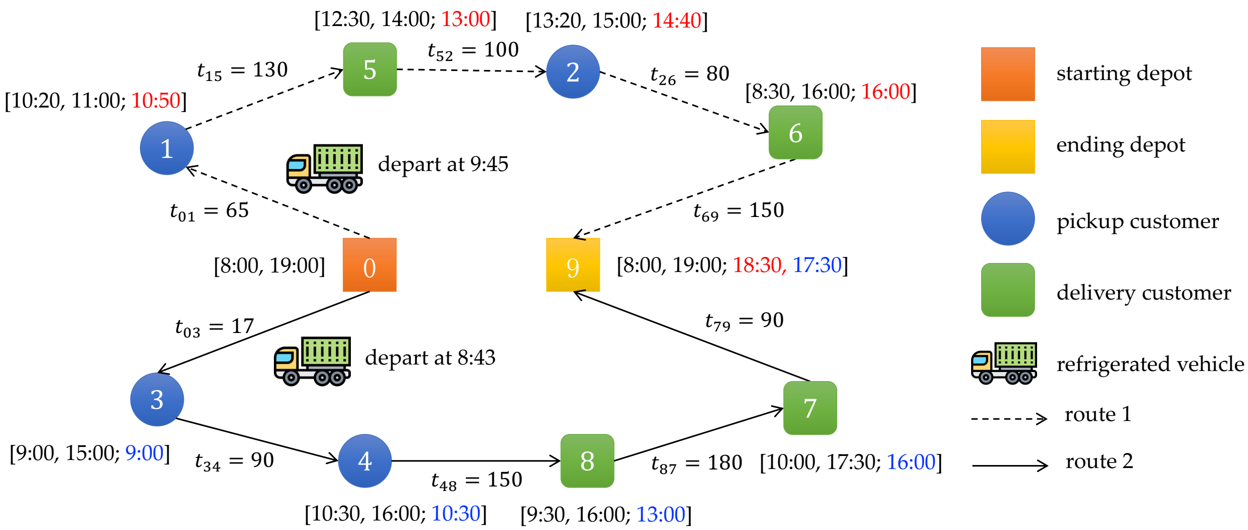

Figure 2 **Illustration of a feasible solution to the PDPTW-IC**

Let $\mathcal{R}$ be the set of all feasible routes and c_r be the cost of route $r \in \mathcal{R}$. For each $r \in \mathcal{R}$, we denote d_r as its total travel distance and assume the last customer in this route is j. Then c_r can be computed as

$$c_r = \rho_0 d_r + \rho_1 (t_j - t_0) + \sum_{i \in \Delta_r} \Lambda(i), \tag{3}$$

where Δ_r is the set of pickup customers in route r and the three terms represent the travel cost, the refrigeration cost, and the quality decay cost, respectively.

A binary parameter a_{ir} equals to 1 if request $i \in V_p$ is served by route r and 0 otherwise. Let θ_r be a binary variable equalling to 1 if route $r \in \mathcal{R}$ is selected and 0 otherwise. The set partitioning model is then constructed as

$$\min \sum_{r \in \mathcal{R}} c_r \theta_r \tag{4}$$

$$\text{s.t.} \quad \sum_{r \in \mathcal{R}} a_{ir} \theta_r = 1 \qquad \forall i \in V_p, \tag{5}$$

$$\sum_{r \in \mathcal{R}} \theta_r \leq |K|, \tag{6}$$

$$\theta_r \in \{0, 1\} \qquad \forall r \in \mathcal{R}. \tag{7}$$

The objective function (4) minimizes the total cost of selected routes. Constraints (5) force that each request is served exactly once. Constraints (6) ensure that at most $|K|$ routes are selected in the final solution. Constraints (7) are integrality constraints.

Due to the large size of set $\mathcal{R}$, it is impractical to enumerate all the feasible routes, even for small-size instances. Thus, we utilize a CG algorithm (Desaulniers et al. 2005) to solve the linear relaxation problem (LRP) at each node of the branch-and-bound (B&B) tree, leading to a B&P

method. In the CG algorithm framework, a restricted master problem (RMP) is obtained by considering the LRP with a reduced subset $\mathcal{R}' \subseteq \mathcal{R}$, which can be efficiently solved to optimality by the simplex method. After solving the RMP, dual solutions are obtained and passed to a pricing problem to check if any route $r' \in \mathcal{R}$ with a negative reduced cost exists. If so, route r' is added to set $\mathcal{R}'$, and the RMP is solved again. Otherwise, the current solution is optimal for the LRP.

We let $\mu_i, i \in V_p$ and μ_0 be the dual variables associated with constraints (5) and (6), respectively. We set $\mu_{2n+1} = 0$ and $\mu_i = 0, \forall i \in V_d$. Then the reduced cost $\bar{c}_r$ of route $r \in \mathcal{R}$ is

$$\bar{c}_r = c_r - \sum_{i \in V} a_{ir}\mu_i = c_r - \sum_{i \in V_r} \mu_i, \tag{8}$$

where V_r is the set of nodes in route r.

4. The Pricing Problem

The pricing problem of PDPTW-IC is an elementary shortest path problem with resource and incompatibility constraints, which is NP-hard. To efficiently solve it, a tailored bidirectional labeling algorithm is developed, which has been proved to be more efficient than the monodirectional labeling algorithm (Righini and Salani 2006, Gschwind et al. 2018). We generate labels in both forward and backward directions and combine them to form completed routes. We first introduce the functions for calculating the refrigeration and quality decay costs of a route in Section 4.1, which will be used in the bidirectional labeling algorithm. We then present the forward and backward labeling algorithms in Sections 4.2 and 4.3, respectively. The label joining procedure is introduced in Section 4.4. We finally present a *ng*-route relaxation technique in Section 4.5 to further accelerate the labeling algorithm.

4.1. Forward and Backward Cost Functions

Besides customer visiting sequences, the refrigeration and quality decay costs are also affected by vehicle departure times (from the depot) derived from some functions' limits. To minimize the quality decay cost, vehicles are expected to leave the depot as early as possible; however, to minimize the refrigeration cost, vehicles are expected to start trips as late as possible (to reduce waiting times at customers) as long as time window constraints can be met. These conflicting objective components lead to asymmetric function extensions, which are explained in the following.

Take Figure 3(a) as an example, suppose that a backward partial route is extended from node $2n+1$ to node p_1. Given a SST t_{p_1} at node p_1, there are many feasible values for the SST t_{d_1} at its successor node d_1, as long as the time window constraint at d_1 can be satisfied. To minimize the refrigeration cost, t_{d_1} is expected to be close to t_{p_1}, which also holds for the objective minimizing the decay cost. However, this analysis process does not apply to the forward extension. Specifically, suppose that a forward partial route is extended from node 0 to node d_2, as shown in

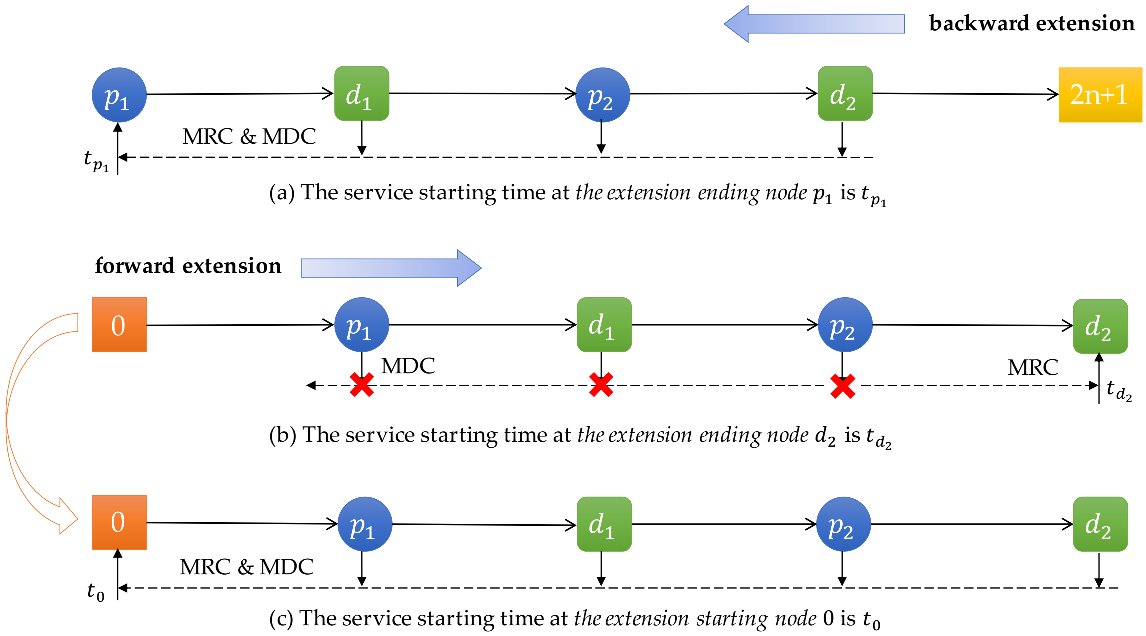

Figure 3 Asymmetric extensions (MRC: minimize refrigeration cost; MDC: minimize quality decay cost)

Figure 3(b). Given a SST t_{d_2} at node d_2, for its predecessor node p_2, to minimize the refrigeration cost, the SST t_{p_2} at p_2 is expected to be close to t_{d_2} (then the route duration will be shorter); however, to minimize the quality decay cost, t_{p_2} is expected to be close to e_0 (i.e., pickup and deliver commodities as early as possible). Thus, it is difficult to decide a time t_{p_2} for node p_2 such that the refrigeration cost and quality decay cost are simultaneously minimized. Similar analyses also apply to nodes d_1 and p_1. Instead, as shown in Figure 3(c), given the SST t_0 at node 0, for any node in the forward partial route, its SST is expected to be close to t_0, which can simultaneously optimize both cost components. To conclude, the specific characteristics of PDPTW-IC make our label extension methods and B&P algorithm different from those developed in the literature.

4.1.1. Forward Cost Functions. For a forward partial or completed route $r^f = (v_0, v_1, \dots, v_m = i)$ beginning at node $v_0 = 0$ and ending at node i, let α and β be the earliest and latest SST at node i, respectively, such that the time window constraints of customers in route r^f are satisfied. We define three functions of t_0 as follows:

- $\delta(t_0)$: the SST at the ending node i, whose range is restricted to $[\alpha, \beta]$ to ensure the time window feasibility of customers in r^f;
- $\tau^d(t_0)$: the quality decay cost of route r^f;
- $\tau^f(t_0)$: the sum of the refrigeration and quality decay costs of route r^f.

Note that $\tau^d(t_0)$ is an auxiliary cost function that helps simplify the derivation process of function $\tau^f(t_0)$. The following theorems and corollary indicate that the three functions admit piecewise linear convex expressions.

Theorem 1 *$\delta(t_0)$ is a piecewise linear nondecreasing convex function expressed as*

$$\delta(t_0) = \max\{\omega + t_0,\ \alpha\}, \tag{9}$$

where ω is the total travel time of route r^f.

Proof. See Appendix A.1 in the Online Supplement. □

Theorem 2 *$\tau^d(t_0)$ is a piecewise linear nondecreasing convex function.*

Proof. See Appendix A.2 in the Online Supplement. □

Corollary 1 *$\tau^f(t_0)$ is a piecewise linear convex function.*

Proof. See Appendix A.3 in the Online Supplement. □

To help understand these three functions, we give their visualizations in Figure 4.

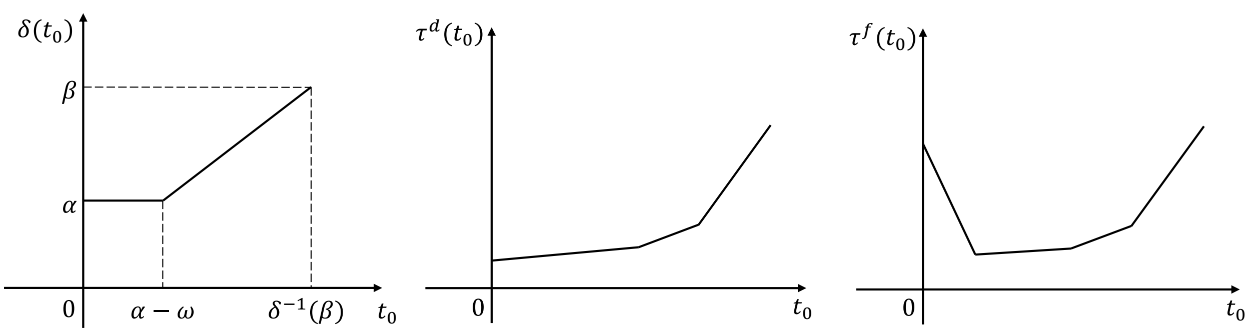

Figure 4 Visualizations of functions $\delta(t_0)$, $\tau^d(t_0)$, and $\tau^f(t_0)$.

4.1.2. Backward Cost Functions. For a backward partial or completed route $r^b = (v_0, v_1, \ldots, v_m = i)$ beginning at node $v_0 = 2n+1$ and ending at node i, we define two functions of t_i as follows:

- $\tau^b(t_i)$: the sum of the refrigeration and quality decay costs of route r^b;
- $\tau^c(t_i)$: the cost evaluation function of route r^b, equalling to $\tau^b(t_i) + \rho_1 t_i$.

Similar to the forward cost function $\tau^d(t_0)$, $\tau^c(t_i)$ is also an auxiliary function as it is more straightforward to derive it than $\tau^b(t_i)$.

Theorem 3 *$\tau^c(t_i)$ is a piecewise linear nondecreasing convex function.*

Proof. See Appendix A.4 in the Online Supplement. □

Based on the definition of $\tau^c(t_i)$, we can obtain $\tau^b(t_i) = \tau^c(t_i) - \rho_1 t_i$. The visualizations of functions $\tau^c(t_i)$ and $\tau^b(t_i)$ can refer to those of $\tau^d(t_0)$ and $\tau^f(t_0)$ as presented in Figure 4, respectively. After having the cost functions, we can rewrite the reduced cost $\bar{c}_r$ of route $r \in \mathcal{R}$ in (8) as

$$\bar{c}_r = \rho_0 d_r + \min_{t_0 \in [\alpha,\beta]} \tau^b(t_0) - \sum_{i \in V_r} \mu_i. \tag{10}$$

The *min* term means that we set the departure time from the depot as the one minimizing $\tau^b(t_0)$.

Besides the asymmetric function extensions introduced earlier, we note that there also exists asymmetry in the definitions of forward and backward functions. Specifically, $\tau^d(t_0)$ only accounts for costs paid in the forward part of a route, while $\tau^c(t_i)$ also accounts for the term $\rho_1 t_i$ that is already paid in the forward part of a route.

4.2. Forward Labeling Algorithm

This section first defines the label associated with a forward route and then presents the method for label extension. Finally, we introduce the dominance rules.

4.2.1. Label Definition and Extension. A forward route $r^f = (v_0, v_1, \ldots, v_m = i)$ is represented by a label $L = (i, \Gamma, U, C, \omega, \alpha, \beta, \mathcal{D}, \mathcal{F})$, which includes the following information:

- i: the current node of the label;
- Γ: the set of nodes that have not been visited by route r^f;
- U: the set of unfinished requests, i.e., the set of pickup customers that have been served by route r^f but their delivery customers have not been served;
- C: the difference between the accumulated dual value and the accumulated travel cost;
- ω: the total travel time of route r^f;
- $[\alpha, \beta]$: the domain of the SST at node i to ensure the time window feasibility of all nodes in r^f;
- $\mathcal{D} = \{\mathcal{D}_h = (\bar{u}_h, \bar{g}_h)\}_{h=1,\ldots,|\mathcal{D}|}$: the parameters used to express function $\tau^d(t_0)$ (see the proof of Theorem 2);
- $\mathcal{F} = \{\mathcal{F}_h = (u_h, g_h)\}_{h=1,\ldots,|\mathcal{F}|}$: the parameters used to express function $\tau^f(t_0)$ (see the proof of Corollary 1).

We denote $\Psi(U) = \{\varphi_v | v \in U\}$ as the set of commodities on board and $W(U) = \sum_{v \in U} q_v$ as the load on board. Label L can be extended to node $j \in \Gamma$ only if $W(U) + q_j \leq Q$ and $\max\{e_j, \alpha + \sigma_{ij}\} \leq \max\{e_j, \min\{l_j, \beta + \sigma_{ij}\}\}$, ensuring the vehicle capacity and time window constraints are respected. Meanwhile, a legal extension must satisfy one of the following conditions:

$$j \in V_p \wedge (\varphi_j \oplus \Psi(U)), \tag{11}$$

$$j \in V_d \wedge (v_{j-n} \in U), \tag{12}$$

$$j = 2n + 1 \wedge U = \varnothing. \tag{13}$$

Constraint (11) means that if j is a pickup node, then the compatibility constraint must be satisfied. Constraint (12) imposes that a route can visit a delivery node only if the corresponding pickup node has been visited. Constraint (13) indicates that the vehicle can return to the ending depot only after finishing all assigned requests.

The resources associated with the new label $L' = (j, \Gamma', U', C', \omega', \alpha', \beta', \mathcal{D}', \mathcal{F}')$ extended from L are calculated as follows:

$$\Gamma' = \Gamma \setminus \{j\} \setminus \mathcal{A} \setminus \mathcal{B} \tag{14}$$

$$C' = C + \mu_j - \rho_0 d_{ij} \tag{15}$$

$$\omega' = \omega + \sigma_{ij} \tag{16}$$

$$\alpha' = \max\{e_j,\ \alpha + \sigma_{ij}\} \tag{17}$$

$$\beta' = \max\{e_j,\ \min\{l_j,\ \beta + \sigma_{ij}\}\} \tag{18}$$

if j is the ending depot $2n+1$:

$$\mathcal{F}' = \mathcal{F} \tag{19}$$

if j is a pickup customer:

$$U' = U \cup \{j\} \tag{20}$$

$$\mathcal{D}' = \mathcal{D} \tag{21}$$

$$\mathcal{F}' = \{(\bar{u}'_h - \rho_1,\ \rho_1\alpha' + \bar{g}'_h)\}_{\mathcal{D}'_h \in \mathcal{D}'} \cup \{(\bar{u}'_h,\ \bar{g}'_h + \rho_1\omega')\}_{\mathcal{D}'_h \in \mathcal{D}'} \tag{22}$$

if j is a delivery customer :

$$U' = U \setminus \{j - n\} \tag{23}$$

$$\mathcal{D}' = \{(\bar{u}_h + \pi_j,\ \pi_j(\omega' - \varsigma'_j) + \bar{g}_h)\}_{\mathcal{D}_h \in \mathcal{D}} \cup \{(\bar{u}_h,\ \bar{g}_h + \pi_j(\alpha' - \varsigma'_j))\}_{\mathcal{D}_h \in \mathcal{D}} \cup \mathcal{D} \tag{24}$$

$$\mathcal{F}' = \{(\bar{u}'_h - \rho_1,\ \rho_1\alpha' + \bar{g}'_h)\}_{\mathcal{D}'_h \in \mathcal{D}'} \cup \{(\bar{u}'_h,\ \bar{g}'_h + \rho_1\omega')\}_{\mathcal{D}'_h \in \mathcal{D}'}, \tag{25}$$

where $\mathcal{D}'_h = (\bar{u}'_h, \bar{g}'_h)$ is the h^{th} element of set $\mathcal{D}'$. In Equation (14), $\mathcal{A}$ and $\mathcal{B}$ are the sets of unreachable customers from the new label L'. Specifically, traveling directly from node j to $v \in \mathcal{A}$ will violate the time window at customer v. While traveling to $v \in \mathcal{B}$ will violate the combination of the incompatibility constraints and time window constraints. Namely, if $z \in U'$ and $\varphi_z \ominus \varphi_v$, then visiting node v by the route associated with label L' will require the vehicle to deliver commodity φ_z first, i.e., visiting node $z+n$ before node v. In this case, we can label node v as unreachable if its time window constraint cannot be satisfied. Mathematically, sets $\mathcal{A}$ and $\mathcal{B}$ are defined as

$$\mathcal{A} = \{v \in V_p | \alpha' + \sigma_{jv} > l_v\}, \tag{26}$$

$$\mathcal{B} = \{v \in V_p | \exists z \in U' \wedge (\varphi_z \ominus \varphi_v),\ \max\{\alpha' + \sigma_{j,z+n}, e_{z+n}\} + \sigma_{z+n,v} > l_v\}. \tag{27}$$

When defining these two sets, we do not consider the delivery customer of request v because we can set $l_v := \min\{l_v, l_{v+n} - \sigma_{v,v+n}\}$, $\forall v \in V_p$ to ensure $l_v + \sigma_{v,v+n} \leq l_{v+n}$ (Ropke and Cordeau 2009). This setting will not affect the feasible solutions of the original problem. The derivation of Equation (26) is straightforward; thus, we only detail how Equation (27) is obtained. Specifically, for label L', we assume customer $v \in V_p$ satisfies the conditions in set $\mathcal{B}$. To prove that v is unreachable for label L', we consider two cases: (i) Customer $z+n$ is not visited yet, i.e., commodity φ_z is still on board. Then we cannot visit customer v due to its associated commodity φ_v is incompatible with φ_z. (ii) Customer $z+n$ has already be visited. Let α'' be the earliest SST at customer $z+n$, then we must have $\alpha'' \geq \max\{\alpha' + \sigma_{j,z+n},\ e_{z+n}\}$, where the first term of the right-hand side is obtained by letting the vehicle travel directly from customers j to $z+n$ since visiting other intermediate nodes before $z+n$ will increase the travel time due to the triangle inequality. Subsequently, we can obtain $\alpha'' + \sigma_{z+n,v} \geq \max\{\alpha' + \sigma_{j,z+n},\ e_{z+n}\} + \sigma_{z+n,v} > l_v$, making customer v unreachable. The idea of defining unreachable sets comes from Feillet et al. (2004). The logic behind it is that when extending to a new label, we remove the nodes that cannot be visited in any extension of the corresponding path, either because they have been visited or because of resource constraints. This idea has the potential for more successful dominance and thus speeding up the labeling process.

After updating sets $\mathcal{D}$ and $\mathcal{F}$, an AERASE procedure detailed in Algorithm 1 is applied to erase redundant lines in these sets. In row 7, the expression $g_i \geq g_j$ denotes the redundant relation between two lines. In row 16, the expression $u_i\zeta + g_i \geq u_k\zeta + g_k$ means that if the intersection point ζ of lines $u_it + g_i$ and $u_jt + g_j$ lies above line $u_kt + g_k$, then line $u_kt + g_k$ is redundant. The redundant relations among lines are illustrated in Figure 5.

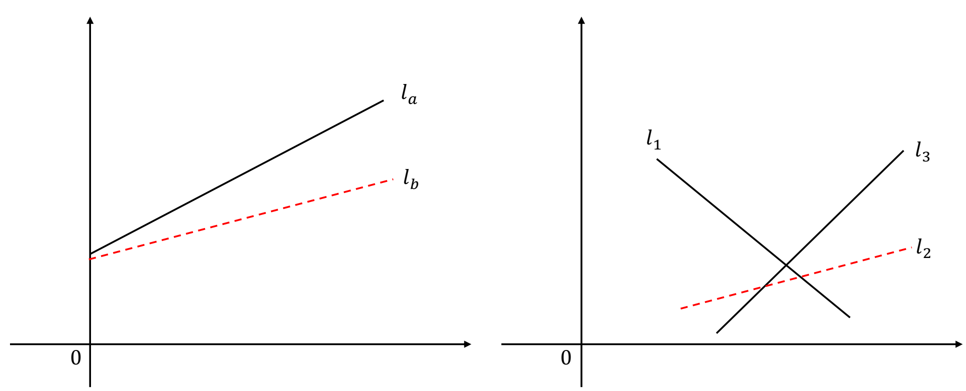

Figure 5 **Visualizations of redundant lines (l_b and l_2) in sets $\mathcal{D}$ and $\mathcal{F}$**

For the forward partial route $r^f = (0)$, since $\tau^d(t_0) = 0$, we can simply set $\mathcal{D} = \{(0,0)\}$. When conducting the forward extension operation, a special case may arise. That is, the range of function $\delta(t_0)$ reduces to a single value (i.e., $\alpha = \beta$) at some nodes. Let $[\alpha_{v_k}, \beta_{v_k}]$ be the range of the SST at node v_k $(k \in [0, m])$. For two successive nodes i and j of a route where $\alpha_i < \beta_i$, the special case

Algorithm 1 The pseudocode of the AERASE procedure

Input: Set A for calculating the forward or backward cost function.
Sort the element $(u, g) \in A$ by u in nondecreasing order. If u is identical, we sort the elements by g.
Initialize set $C_L = \emptyset$.
for $i = |A|, \ldots, 2$ **do**
  if $(u_i, g_i) \notin C_L$ **then**
    for $j = i-1, \ldots, 1$ **do**
      if $((u_j, g_j) \notin C_L) \wedge (g_i \geq g_j)$ **then**
        $C_L := C_L \cup \{(u_j, g_j)\}$.
$A := A \setminus C_L$ and $C_L = \emptyset$.
for $i = 1, \ldots, |A|-2$ **do**
  if $(u_i, g_i) \notin C_L$ **then**
    for $j = |A|, \ldots, i+2$ **do**
      if $(u_j, g_j) \notin C_L$ **then**
        for $k = j-1, \ldots, i+1$ **do**
          Let $\zeta = (g_i - g_j)/(u_j - u_i)$.
          if $((u_k, g_k) \notin C_L) \wedge (u_i\zeta + g_i \geq u_k\zeta + g_k)$ **then**
            $C_L := C_L \cup \{(u_k, g_k)\}$.
$A := A \setminus C_L$.
Output: Set A after removing redundant lines.

happens when: (1) $l_i + \sigma_{ij} \leq e_j$, then the SST at node j can only be e_j regardless of when node i is served; (2) $\alpha_i + \sigma_{ij} = l_j$, then the SST at node j can only be l_j and the range of the SST at node i should be updated to $[\alpha_i, \alpha_i]$. Moreover, we also need to update the ranges of nodes before i, to respect the time window constraints at j and its successors. Note that if the SST at a node is fixed, the time schedules of its successors are also fixed.

4.2.2. Dominance Rules. To accelerate the labeling algorithm, dominance rules are often utilized to eliminate unpromising labels. Before moving forward, we first give some definitions. Let $\mathbb{P}(L)$ be the partial path associated with label L. A feasible partial path that can be used to extend $\mathbb{P}(L)$ is called as *a feasible completion* for label L. Let $\Omega(L)$ be the set of all feasible completions of label L. For a completion $\xi \in \Omega(L)$, we use $\mathbb{P}(L) \otimes \xi$ to denote a feasible completed route obtained by combining $\mathbb{P}(L)$ and ξ, and $\bar{c}(\mathbb{P}(L) \otimes \xi)$ is the resulting reduced cost. For two labels L_1 and L_2 ending at the same node, we use $L_1 \prec L_2$ to indicate that label L_1 dominates label L_2. The standard dominance rule introduced by Dabia et al. (2013) is presented in Proposition 1.

Proposition 1 *For labels L_1 and L_2 ending at the same node, $L_1 \prec L_2$ if*

(1) $\Omega(L_2) \subseteq \Omega(L_1)$,

(2) $\bar{c}(\mathbb{P}(L_2) \otimes \xi) \geq \bar{c}(\mathbb{P}(L_1) \otimes \xi), \forall \xi \in \Omega(L_2)$.

Condition (1) means that L_1 can combine with more feasible completions to form a completed route than L_2. As the departure time from depot 0 needs to be decided in the forward extension, it is not straightforward to check this condition for our dominance rule. On the other hand, since the SST t_i at the current node i affects the feasibility of the completions due to the time window constraints, the larger the range of t_i, the more completions can be combined with the path of the corresponding label. Since we have recorded the range of t_i, i.e., $[\alpha, \beta]$ in the forward extension, one necessary condition to ensure condition (1) is $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$. Therefore, to derive the forward dominance rule for our problem, we first use $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$ together with other necessary conditions to guarantee condition (1). If we can ensure condition (2) holds for any $t_i \in [\alpha_2, \beta_2]$, we are done. To do so, we must calculate the reduced costs of $\mathbb{P}(L_1)$ and $\mathbb{P}(L_2)$. From Equation (8), to compute the reduced cost, we first need to know the total cost of a forward route. However, since the argument of the forward cost functions is t_0 (not t_i), we must derive a function, say $\eta(t_i)$, to compute the forward cost under the condition that t_i (instead of t_0) is given.

For a forward route $r^f = (v_0, v_1, \ldots, v_m = i)$, function $\eta(t_i)$ denotes its refrigeration and quality decay costs, which will be used in both the forward dominance rules and the label joining procedure. The logic behind defining $\eta(t_i)$ is that to use $\tau^f(t_0)$ to compute the forward cost, we first need to transform t_i to t_0 via a function. Note that besides $\alpha < \beta$, the definition of $\eta(t_i)$ also applies to the special case $\alpha = \beta$. For brevity purpose, we provide the detailed derivation of $\eta(t_i)$ in Appendix B in the Online Supplement. Based on the preceding analyses, we derive another dominance rule in Proposition 2, which is more straightforward than Proposition 1 as it compares the accumulated costs and resource consumptions associated with two labels.

Proposition 2 *(Forward Dom) For labels L_1 and L_2 ending at the same node, $L_1 \prec L_2$ if*

(1) $\Gamma_2 \subseteq \Gamma_1$,

(2) $U_1 \subseteq U_2$,

(3) $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$,

(4) $\eta_1(t_i) - C_1 \leq \eta_2(t_i) - C_2, \forall t_i \in [\alpha_2, \beta_2]$.

Proof. See Appendix A.5 in the Online Supplement. □

The dominance rule in Proportion 2 requires the comparison of two complex functions, i.e., $\eta_1(t_i)$ and $\eta_2(t_i)$, which is difficult to be implemented. Since the expressions of $\delta(t_0)$ and $\tau^f(t_0)$ are already known, we propose another dominance rule in Proposition 3 that is easier to handle.

Proposition 3 *(Forward Dom*) For labels L_1 and L_2 ending at the same node, $L_1 \prec L_2$ if*

(1) $\Gamma_2 \subseteq \Gamma_1$,

(2) $U_1 \subseteq U_2$,

(3) $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$,

(4) $\tau_2^f(t_0) - \bar{\tau}_1^f(t_0) \geq C_2 - C_1, \forall t_0 \in [\alpha_2 - \omega_2, \beta_2 - \omega_2]$ *if* $\alpha_2 < \beta_2$,

(5) $\eta_1(\alpha_2) - C_1 \leq \eta_2(\alpha_2) - C_2$,

where $\bar{\tau}_1^f(t_0) = \tau_1^f(t_0 + \omega_2 - \omega_1)$.

Proof. See Appendix A.6 in the Online Supplement. □

We choose time as the resource bounding for label extensions and denote Θ^f (Θ^b) as the threshold for stopping forward (backward) extension. The process of generating forward labels is given in Algorithm 2, where $\mathcal{L}$ is the set of labels that have not been extended and set $\bar{\mathcal{L}}_i$ stores labels ending at customer $i \in V_c$ that are not dominated. With a slight abuse of notation, we denote $\bar{c}(L)$ as the reduced cost associated with label L, $\alpha(L)$ as the earliest SST time at the ending node of label L, and $\Gamma(L)$ as the set of nodes that have not been extended by label L. In lines 17 and 20, Proposition 3 is used to prune dominated labels.

4.3. Backward Labeling Algorithm

For a backward route $r^b = (v_0, v_1, \ldots, v_m = i)$, we define the corresponding label as $L = (i, \Gamma, U, C, \alpha, \beta, \mathcal{C})$. Attributes i, Γ, C, α, and β have the same meanings as their counterparts in the forward label. The definitions of other attributes are as follows:

- U: the set of delivery customers that have been served by route r^b but their pickup customers have not been served;
- $\mathcal{C} = \{\mathcal{C}_h = (\hat{u}_h, \hat{g}_h)\}_{h=1,\ldots,|\mathcal{C}|}$: the parameters used to express function $\tau^c(t_i)$ (see the proof of Theorem 3).

We only record $\tau^c(t_i)$ in the labeling process, because we can get $\tau^b(t_i)$ easily once $\tau^c(t_i)$ is known and it is sufficient to only use $\tau^c(t_i)$ in the backward dominance. Label L can be extended to node $j \in \Gamma$ only if $W(U) + q_j \geq -Q$ and $e_j \leq \min\{l_j,\ \beta - \sigma_{ji}\}$, guaranteeing the vehicle capacity and time window constraints are respected. Moreover, a legal extension should also meet one of the following three constraints:

$$j \in V_d \wedge (\varphi_j \oplus \Psi(U)), \tag{28}$$

$$j \in V_p \wedge (v_{j+n} \in U), \tag{29}$$

$$j = 0 \wedge U = \varnothing, \tag{30}$$

which have similar meanings as their counterparts (11)–(13).

Algorithm 2 ForwardLabeling(Θ^f): The process of generating forward labels

Define L_k: a label ending at node k; R_s: the set of labels associated with a completed route.
Initialize $L_0 = (0, V \setminus \{0\}, \emptyset, 0, 0, e_0, l_0, \{(0,0)\}, \emptyset)$; $\mathcal{L} = \{L_0\}$; $\bar{\mathcal{L}}_i = \emptyset, \forall i \in V_c$.
while $\mathcal{L} \neq \emptyset$ **do**
  $N_c = \emptyset$; select the first element L_k of set $\mathcal{L}$ and update $\mathcal{L} := \mathcal{L} \setminus \{L_k\}$.
  if $k = 2n+1$ **then**
    if $\bar{c}(L_k) < 0$ **then**
      $R_s := R_s \cup \{L_k\}$.
  else if $\alpha(L_k) < \Theta^f$ **then**
    for $j \in \Gamma(L_k)$ **do**
      if label L_k can be extended to node j **then**
        Generate label L_j by extending L_k to j and update $\mathcal{L} := \mathcal{L} \cup \{L_j\}$.
        if $j \neq 2n+1$ **then**
          $N_c := N_c \cup \{j\}$, $\bar{\mathcal{L}}_j := \bar{\mathcal{L}}_j \cup \{L_j\}$.
  for $i \in N_c$ **do**
    $D = \emptyset$; select the last element L_l of set $\bar{\mathcal{L}}_i$.
    for $L_i \in \bar{\mathcal{L}}_i$ **do**
      if $L_i \prec L_l$ **then**
        $D := D \cup \{L_l\}$.
        Break;
      else if $L_l \prec L_i$ **then**
        $D := D \cup \{L_i\}$.
    $\bar{\mathcal{L}}_i := \bar{\mathcal{L}}_i \setminus D$.

The resources associated with the new label $L' = (j, \Gamma', U', C', \alpha', \beta', \mathcal{C}')$ extended from L are calculated as follows:

$$C' = C + \mu_j - \rho_0 d_{ji} \tag{31}$$

$$\alpha' = e_j \tag{32}$$

$$\beta' = \min\{l_j, \beta - \sigma_{ji}\} \tag{33}$$

if j is a pickup customer or the starting depot 0**:**

$$U' = U \setminus \{j+n\} \tag{34}$$

$$\mathcal{C}' = \{(0,\ \hat{u}_h e_i + \hat{g}_h)\}_{\mathcal{C}_h \in \mathcal{C}} \cup \{(\hat{u}_h,\ \hat{u}_h \sigma_{ji} + \hat{g}_h)\}_{\mathcal{C}_h \in \mathcal{C}} \tag{35}$$

if j is a delivery customer:

$$U' = U \cup \{j\} \tag{36}$$

$$\mathcal{C}' = \{(\hat{u}_h + \pi_j,\ \hat{u}_h \sigma_{ji} + \hat{g}_h - \pi_j \varsigma'_j)\}_{\mathcal{C}_h \in \mathcal{C}} \cup \{(\hat{u}_h,\ \hat{u}_h \sigma_{ji} + \hat{g}_h)\}_{\mathcal{C}_h \in \mathcal{C}}$$

$$\cup \{(\pi_j,\ \hat{u}_h e_i + \hat{g}_h - \pi_j \varsigma'_j)\}_{\mathcal{C}_h \in \mathcal{C}} \cup \{(0,\ \hat{u}_h e_i + \hat{g}_h)\}_{\mathcal{C}_h \in \mathcal{C}}. \tag{37}$$

The update of Γ' is the same as its forward counterpart (14). Sets $\mathcal{A}$ and $\mathcal{B}$ are defined as

$$\mathcal{A} = \{v \in V_d | \beta' - \sigma_{vj} < e_v\}, \tag{38}$$

$$\mathcal{B} = \{v \in V_d | \exists z \in U' \wedge (\varphi_z \ominus \varphi_v), \min\{\beta' - \sigma_{z-n,j}, l_{z-n}\} - \sigma_{v,z-n} < e_v\}. \tag{39}$$

We note that the backward extension starts from the first node after depot $2n+1$ and ends at depot 0. The update of set $\mathcal{C}$ can refer to the proof process of Theorem 3 and the AERASE procedure is applied after its update. When $r^b = (v_0, v_1)$, according to $\tau^c(t_{v_1}) = (\rho_1 + \pi_{v_1}) t_{v_1}$, we can simply set $\mathcal{C} = \{(\rho_1 + \pi_{v_1}, 0)\}$. The dominance rule used in backward labeling is presented in Proposition 4. The process of generating backward labels is similar to that for producing forward labels as presented in Algorithm 2.

Proposition 4 *(Backward Dom) For labels L_1 and L_2 ending at the same node i, $L_1 \prec L_2$ if*

(1) $\Gamma_2 \subseteq \Gamma_1$,

(2) $U_1 = U_2$,

(3) $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$,

(4) $\tau_1^c(t_i) - C_1 \le \tau_2^c(t_i) - C_2, \forall t_i \in [\alpha_2, \beta_2]$.

Proof. See Appendix A.7 in the Online Supplement. □

4.4. Label Joining

When forward and backward labels are symmetric, we can directly sum two cost functions (i.e., $F_{p_2}(t_{p_2})$ and $B_{p_2}(t_{p_2})$ in Figure 6) to obtain the cost of a completed route. However, when labels are asymmetric, transformations are needed before the joining operation. Take the case in Figure 6 as an example, for the backward cost function τ^b, its argument is the SST t_{p_2} at the joining point p_2. Whereas the argument of the forward cost function τ^f is the SST t_0 at depot 0; thus, we need to use function $\phi(t_{p_2})$ to derive t_0 first given the SST at node p_2 is t_{p_2}, i.e., $t_0 = \phi(t_{p_2})$. Then we can sum functions $\tau^f(\phi(t_{p_2}))$ and $\tau^b(t_{p_2})$ (as they are both functions of variable t_{p_2}) to obtain the cost of a completed route.

Now suppose we have a forward label L^f and a backward label L^b ending at the same customer i. If i is a pickup customer, let $U'_1 = \{v_d | v_d = v_p + n, \forall v_p \in U^f \setminus \{i\}\}$ and $U'_2 = U^b$. If i is a delivery customer, let $U'_1 = \{v_d | v_d = v_p + n, \forall v_p \in U^f\}$ and $U'_2 = U^b \setminus \{i\}$. Labels L^f and L^b can be joined together at node i if the following constraints are satisfied:

$$\mathbb{P}(L^f) \cap \mathbb{P}(L^b) = \{i\}, \tag{40}$$

$$\max\{\alpha^f, \alpha^b\} \le \min\{\beta^f, \beta^b\}, \tag{41}$$

$$U'_1 = U'_2. \tag{42}$$

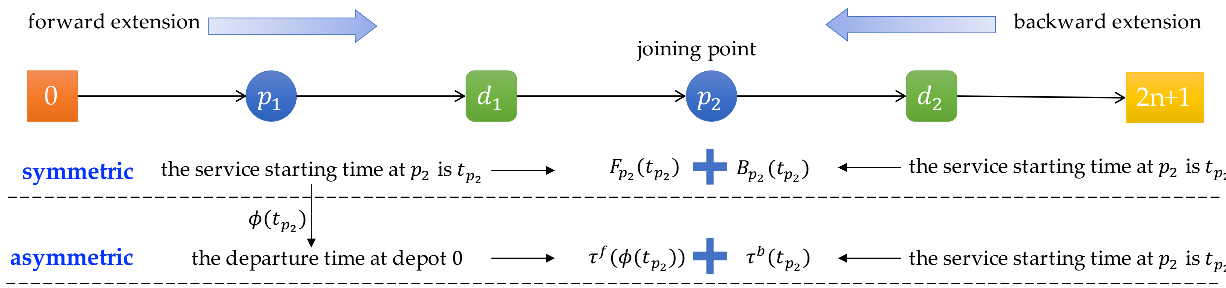

Figure 6 **The symmetric and asymmetric label joining operations**

The reduced cost $\bar{c}(r)$ of a completed route r is the minimum of the sum of the forward and backward reduced costs, i.e.,

$$\bar{c}(r) = \min_{\max\{\alpha^f,\alpha^b\} \le t_i \le \min\{\beta^f,\beta^b\}} \{\eta(t_i) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} - C^f - C^b + \mu_i, \tag{43}$$

where $\mathbb{1}(\cdot)$ is an indicator function. Let

$$F(t_i) = \eta(t_i) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d), \quad \forall t_i \in [\max\{\alpha^f,\alpha^b\}, \min\{\beta^f,\beta^b\}]. \tag{44}$$

To find the minimum of $F(t_i)$, we only discuss the case where $\max\{\alpha^f,\alpha^b\} < \min\{\beta^f,\beta^b\}$, as $F(t_i)$ is fixed when $\max\{\alpha^f,\alpha^b\} = \min\{\beta^f,\beta^b\}$. When $\max\{\alpha^f,\alpha^b\} < \min\{\beta^f,\beta^b\}$, we must have $\alpha^f < \beta^f$. Note that $\delta^{-1}(t_i) = t_i - \omega$, $\forall t_i \in (\alpha^f, \beta^f]$. To analyze $F(t_i)$, we restate the definition of function $\eta(t_i)$ (see equations (B.3) and (B.4) in the Online Supplement) as follows:

$$\eta(t_i) = \tau^f(\phi(t_i)) = \begin{cases} \tau^f(t_0') & \text{if } t_i = \alpha^f, \quad (45a) \\ \tau^f(t_i - \omega) & \text{if } t_i > \alpha^f. \quad (45b) \end{cases}$$

where

$$t_0' = \arg \min_{t_0 \in [0,\ \alpha^f - \omega]} \tau^f(t_0). \tag{46}$$

From the definition we know that $\tau^f(t_0') \le \tau^f(\alpha^f - \omega)$, so $t_i = \alpha^f$ could be a discontinuity point of function $\eta(t_i)$. The graph of $\tau^f(t_i - \omega)$ can be obtained by shifting the graph of $\tau^f(t_0)$ to the right. In this way, we can get the graph of $\eta(t_i)$ as shown in Figure 7(a). The visualization of $\tau^b(t_i)$ is given in Figure 7(b). Since both $\eta(t_i)$ and $\tau^b(t_i)$ are piecewise linear convex functions, $F(t_i)$ is also a piecewise linear convex function as shown in Figure 7(c). Note that if $\tau^f(t_0') < \tau^f(\alpha^f - \omega)$, $t_i = \alpha^f$ would also be a discontinuity point of function $F(t_i)$. By expanding $\eta(t_i)$ and $\tau^b(t_i)$, we get the following proposition.

Proposition 5 *When $t_i \in (\alpha^f, \min\{\beta^f,\beta^b\}]$,*

$$F(t_i) = \max_{\mathcal{F}_h \in \mathcal{F}} \left\{ \max_{\mathcal{C}_h \in \mathcal{C}} \{(u_h + \hat{u}_h - \rho_1 - \pi_i \mathbb{1}(i \in V_d))t_i + g_h + \hat{g}_h - u_h\omega\} \right\}. \tag{47}$$

Proof. See Appendix A.8 in the Online Supplement. □

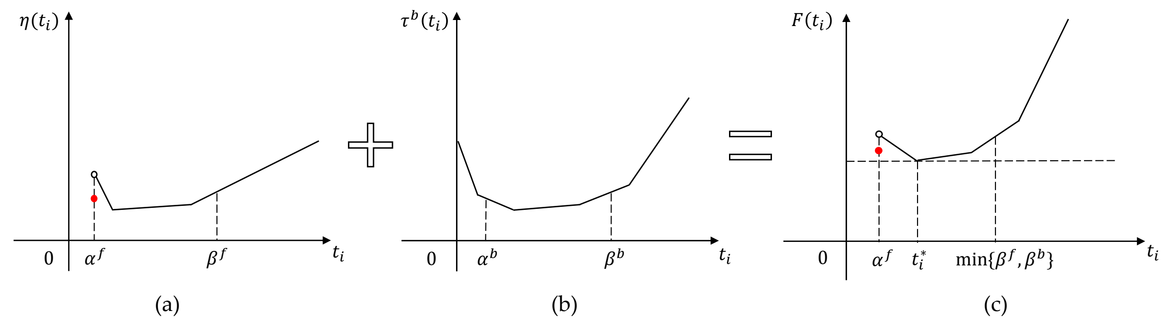

Figure 7 **Visualizations of functions** $\eta(t_i)$, $\tau^b(t_i)$, **and** $F(t_i)$

Up to now, we have known the expression and graph of function $F(t_i)$; thus, we can get its minimum on the interval $[\max\{\alpha^f, \alpha^b\}, \min\{\beta^f, \beta^b\}]$ and also the value of $\bar{c}(r)$.

4.5. Ng-Route Relaxation

Besides the bidirectional search, we also use the ng-route relaxation introduced by Baldacci et al. (2011b) to accelerate the labeling algorithm. The ng-route concepts relies on the definition of a ng-set $\mathcal{N}_i$ for each customer $i \in V_c$. A ng-set $\mathcal{N}_i$ contains the closest λ customers to i and node i itself. A cycle $(v_1, v_2, \ldots, v_k, v_1)$ is called an ng-cycle if $v_1 \in \bigcap_{i'=1,\ldots,k} \mathcal{N}_{v_{i'}}$. A route that does not contain any ng-cycle is an ng-route.

In our labeling algorithm, the ng-route relaxation allows each request to be visited more than once under two limitations: (1) the pickup node cannot be revisited before its delivery node is visited, and (2) the precedence constraints associated with the request must be respected. For each customer $i \in V_c$, we define two ng-sets $\mathcal{N}_i^f$ and $\mathcal{N}_i^b$ for the forward and backward extensions, respectively. Distance is used as the criterion to define the proximity of customers. The distance $dist(i,j)$ of two requests is defined as

$$dist(i,j) = \min\{\sigma_{ij}, \sigma_{i,j+n}, \sigma_{i+n,j}, \sigma_{i+n,j+n}\}, \forall i,j \in V_p.$$

Then, for each request $i \in V_p$, the pickup customers of the λ closest requests and i itself will be added to $\mathcal{N}_i^f$. We define $\mathcal{N}_i^f = \mathcal{N}_{i-n}^f, \forall i \in V_d$ and $\mathcal{N}_i^b = \{j+n | j \in \mathcal{N}_i^f\}, \forall i \in V_c$.

For a forward label L^f and its associated route $\mathbb{P}(L^f) = (v_0, v_1, ..., v_m)$, let $ng(L^f)$ be the subset of requests such that an ng-cycle is prohibited when extending L^f. Then $ng(L^f)$ can be written as

$$ng(L^f) = \left\{ i \in V_p \middle| \exists k \in \{1, ..., m\} \text{ such that } n + i = v_k \text{ and } i \in \bigcap_{i'=k}^{m} \mathcal{N}_{v_{i'}}^f \right\}.$$

For a backward label L^b and its associated route $\mathbb{P}(L^b) = (v_0, v_1, ..., v_m)$, $ng(L^b)$ is defined as

$$ng(L^b) = \left\{ i \in V_d \middle| \exists k \in \{1, ..., m\} \text{ such that } i - n = v_k \text{ and } i \in \bigcap_{i'=k}^{m} \mathcal{N}_{v_{i'}}^b \right\}.$$

When using the ng-route relaxation in the pricing problem, for the forward extension, we need to update condition (11) and set $\Gamma^{f'}$ in (14) as

$$j \in V_p \wedge (\varphi_j \oplus \Psi(U^f)) \wedge (j \notin U^f) \quad \text{and} \quad \Gamma^{f'} = V_c \cup \{2n+1\} \setminus ng(L^{f'}) \setminus \mathcal{A}^f \setminus \mathcal{B}^f,$$

respectively. Similarly, for the backward extension, we update condition (28) and set $\Gamma^{b'}$ as

$$j \in V_d \wedge (\varphi_j \oplus \Psi(U^b)) \wedge (j \notin U^b) \quad \text{and} \quad \Gamma^{b'} = V_c \cup \{0\} \setminus ng(L^{b'}) \setminus \mathcal{A}^b \setminus \mathcal{B}^b,$$

respectively. Meanwhile, the joining condition (40) ensuring elementary routes is not needed.

5. Branching Strategy

We use the best-first strategy to explore the B&B tree, that is, priority is given to an unexplored tree node whose father node has the smallest lower bound. In our B&P algorithm, we use the following hierarchical branching scheme. First, we branch on the number of used vehicles. Let $(\tilde{\theta}_r)(r \in \mathcal{R})$ be an optimal solution of the LRP and $\tilde{m} = \sum_{r \in \mathcal{R}} \tilde{\theta}_r$ be the number of used vehicles in the optimal solution. If $\tilde{m}$ is fractional, we generate two children nodes by forcing $\sum_{r \in \mathcal{R}} \theta_r \leq \lfloor \tilde{m} \rfloor$ and $\sum_{r \in \mathcal{R}} \theta_r \geq \lceil \tilde{m} \rceil$, respectively. If $\tilde{m}$ is integer, we branch on a fractional arc. Let the total flow through arc $(i,j) \in A$ in the optimal solution be $\tilde{x}_{ij} = \sum_{r \in \mathcal{R}} \tilde{\gamma}_{ijr} \tilde{\theta}_r$, where $\tilde{\gamma}_{ijr}$ denotes whether arc (i,j) is traversed by route r. If there exist multiple arcs with fractional values of $\tilde{x}_{ij}$, we then branch on arc (i^*, j^*) whose value of $\tilde{x}_{i^* j^*}$ is the closest to 0.5. Two children nodes are generated by forbidding and enforcing vehicles to travel through arc (i^*, j^*) in the labeling algorithm, respectively. We can implement the former case by deleting arc (i^*, j^*). For the latter case, we delete arcs $(i^*, l), l \neq j^*$ and $(l, j^*), l \neq i^*$.

6. Computational Experiments

In this section, we introduce the instances and evaluate the B&P algorithm and the new objective function. Our algorithm was coded in Java programming language using ILOG CPLEX 12.10 as the solver. All parameters in the solver were set to their default values. The RMPs in the CG framework were solved by the simplex algorithm. All experiments were run on a laptop with a 2.5 GHz Intel(R) Core(TM) i5-7300HQ processor and 24 GB of memory under the macOS system using a single thread. The time limit on each execution of the B&P algorithm was set to 7200 seconds, and all the computing times in this section are reported in seconds.

6.1. Instances

We generate PDPTW-IC instances based on the PDPTW instances proposed by Li and Lim (2008), which originate from the Solomon (1987) VRPTW instances by randomly pairing up customers in the same route. Like Solomon's instance classification, the PDPTW instances also have six classes: LC1, LC2, LR1, LR2, LRC1, and LRC2. Customers are clustered in LC instances while randomly distributed in LR instances. A subset of customers are clustered, and the others are randomly distributed in LRC instances. The LC1, LR1, and LRC1 instances have a short time horizon, while the LC2, LR2, and LRC2 instances have a long time horizon. All instances have more than 50 requests, and some customers are dummy nodes for coupling purposes.

We generate three instance sets with different time horizons, where *Set Short* and *Set Normal* instances are adapted from the LRC1 and LC1 instances, respectively. These two instance sets are used to simulate short-distance and long-distance urban distributions, respectively. In particular, we set four hours for Set Short and one day for Set Normal. The *Set Large* instances are modified from the LC2 instances, which have a long time horizon—around three days. To simulate long-distance trunk transportation, the travel distances between nodes in Set Large instances are tripled. In each instance set, the number of customers takes values from the set {10, 20, 30, 40, 50, 60, 80}. Based on the transportation cost of refrigerated vehicles provided by the 3PL company, we set $\rho_0 = 1.2$ and $\rho_1 = 0.4$.

Each PDPTW-IC instance includes information on customer, vehicle, and commodity. For a customer, its location, demand, and time window are known. For a vehicle, its travel speed and capacity are known. We set node coordinates, time windows, service times, and vehicle speeds the same as those in Li and Lim (2008). We also keep the same pickup-delivery pairs. For each of their instances, we rank the pairs based on the order the pickup customer appears. Then we generate instances with $n = 10, \dots, 80$ customers by selecting the first $n/2$ pairs. We enlarge customer demands and vehicle capacities ten times to reflect real-world statistical data. For each commodity, its perishability or stability, price, and (in)compatibility are known. We mainly consider four types of commodities: fruit and vegetable, poultry and meat, medical supply, and manufactured food. We assume fruit and vegetable are perishable while the others are stable under cold chain conditions. In each instance, the proportion of perishable commodities is around 30%. When collecting the data, we found that the *harvest period* was quite long, especially for the imported fruits and vegetables, which was normally greater than the *fresh period*. As a result, most perishable products began to decay visibly before time e_0. Thus, we set $\varsigma'_\varphi = 0$, $\forall \varphi \in \overline{\mathcal{P}}$ in the following experiments. We also assume that a perishable product at time ς'' can only be sold at a half price of that at time ς'. The price differences and time spans of visible changes for all perishable products are detailed in Table C1 in the Online Supplement, which are based on the local market data in Wuhan, China. Based on some preliminary tests, we set the neighborhood size λ of the ng-route relaxation to 15. The details of instances and solutions are also available at `https://github.com/dengfaheng/PDPTW-IC-INST`.

6.2. Evaluation of Label Extension Method

As suggested before, the forward extension is quite different from the backward extension in our labeling algorithm. Thus, our first attempt is to evaluate the efficiency of different extension methods, including only forward extension (FE), only backward extension (BE), and bidirectional extension (BidE). The FE (BE) can be implemented by setting $\Theta^f = l_0 + 1, \Theta^b = e_0 - 1$ ($\Theta^f = e_0 - 1, \Theta^b = l_0 + 1$) and the BidE can be achieved by setting $\Theta^f = \Theta^b = l_0/2$. We conduct experiments by selecting instances of different sizes from each set. Results are reported in Table 3, where columns n and UB denote the number of customers and the optimal value of each instance. Columns *F.T*, *B.T*, and *Bid.T* represent the computing times of using FE, BE, and BidE, respectively.

Table 3 Performance comparisons among the three label extension methods

	Set Short				Set Normal				Set Large			
n	UB	F.T	B.T	Bid.T	UB	F.T	B.T	Bid.T	UB	F.T	B.T	Bid.T
10	411.66	0.19	0.01	0.10	994.47	0.01	0.01	0.01	4330.39	0.01	0.01	0.01
10	420.29	0.03	0.01	0.02	920.60	0.02	0.01	0.01	2512.64	0.02	0.01	0.01
20	725.58	0.05	0.01	0.01	1271.89	0.01	0.01	0.01	4681.68	0.01	0.01	0.01
20	723.05	0.12	0.04	0.04	1405.97	1.40	0.93	0.88	5174.13	0.13	0.10	0.27
30	1188.09	0.12	0.12	0.14	1792.24	0.02	0.05	0.02	6001.87	0.08	0.43	0.08
30	1061.70	0.05	0.06	0.04	1858.94	0.14	0.16	0.06	5819.13	1.90	1.04	0.64
40	1878.65	0.74	0.04	0.03	2542.21	0.02	0.03	0.02	7724.19	0.23	0.84	0.30
40	1419.37	0.21	0.16	0.06	2544.52	4.02	15.62	6.62	8025.10	2.40	3.33	3.40
50	2237.43	0.39	0.38	0.34	3324.36	0.02	0.04	0.02	10909.69	0.08	0.98	0.17
50	1753.96	0.10	0.06	0.05	3029.92	0.29	1.49	0.17	10625.60	10.72	6.46	5.58
60	2740.03	0.18	0.25	0.18	4699.65	0.09	0.51	0.10	12549.44	0.17	15.06	2.90
60	2346.77	0.14	0.11	0.07	3932.81	6.21	18.15	1.84	12415.15	334.21	94.51	31.14
80	3648.41	0.14	0.26	0.13	6197.25	7.92	185.31	2.50	15554.95	1323.33	N/A	344.42
Sum	-	2.46	1.51	1.21	-	20.17	222.32	12.26	-	1673.29	122.78	388.93

N/A: No feasible solution is provided within the 7200-second time limit.

From Table 3, we observe that all the selected instances in Set Short are solved to optimality and the time differences among the three methods are minor. However, for Set Normal, although the three methods can provide optimal solutions for all the instances, the total computing time of BidE is shorter than those of FE and BE. For instances in Set Large, BE cannot provide a feasible solution for the 80-customer instance within the 7200-second time limit. Although FE can produce optimal solutions for all the selected instances, its computing time is much higher than that of BidE, i.e., 1673.29 seconds against 388.93 seconds. From the results, we can conclude that the bidirectional label extension method is more efficient in solving PDPTW-IC instances, particularly for large-size instances with a long time horizon. Therefore, in the following sections, we use BidE to perform our experiments.

6.3. Evaluation of the B&P Algorithm

In this section, we first report the results of our B&P algorithm for all the instances, which are summarized in Table 4 and the detailed results are presented in Tables C2–C4 in Appendix C. In Table 4, column *Total* denotes the total number of instances. Column *Solved* reports the number of instances that are solved to optimality. For these optimally solved instances, the average computing time and the average number of explored nodes in the B&B tree are reported in columns *Time* and *Node*, respectively.

Table 4 A summary of results generated by our B&P algorithm for all the instances

	Set Short				Set Normal				Set Large			
n	Total	Solved	Time	Node	Total	Solved	Time	Node	Total	Solved	Time	Node
10	8	8	0.05	1.25	9	9	0.01	1.00	8	8	0.02	1.00
20	8	8	0.20	2.25	9	9	0.62	6.56	8	8	0.15	1.00
30	8	8	0.69	9.75	9	9	1.65	1.00	8	8	36.12	1.25
40	8	8	7.92	173.25	9	9	98.79	18.78	8	8	108.42	2.00
50	8	8	749.37	17130.00	9	9	30.97	1.67	8	8	332.71	1.50
60	8	8	664.43	2832.00	9	8	180.90	24.25	8	7	450.26	1.57
80	8	8	1488.31	1186.75	9	5	126.38	1.00	8	2	2930.68	1.00
Sum	56	56	2910.98	21335.25	63	58	439.31	54.25	56	49	3858.36	9.32

■ Note that results in columns *Time* and *Node* are the average results of instances that are solved to optimality.

For Set Short, all the 56 instances are solved to optimality by B&P. This is because instances in this set have relatively narrow time windows and thus are the most tractable in comparison with instances in other sets. Moreover, we observe that the B&B process has explored a large number of nodes for instances in this set. For Set Normal, 58 out of 63 instances are solved to optimality with much fewer explored nodes. The instances in Set Large are the most difficult due to their wider time windows and longer time horizons, and thus 7 out of 56 instances cannot be optimally solved. Overall, the B&P algorithm can produce optimal solutions for 163 out of 175 instances within an acceptable time frame. Tables C2–C4 further show that for most instances, the lower bound at the root node is very tight and B&P can find the optimal solution by only exploring the root node, proving the effectiveness of our set partitioning model and B&P algorithm.

Table 5 reports the performance of dominance rules at the root node of the B&B tree, where columns *Label* and *Dominated* are the average numbers of generated and dominated labels, respectively. Column *Ratio (%)* provides the percentage of dominated labels, computed as *Ratio (%)* = *Dominated/Label*×*100*. We observe that 52.51% of labels can be dominated for Set Short instances, and the ratio increases to 70.47% for Set Large instances. For these dominated labels, we can directly discard them and also not need to consider their extensions, reducing the overall computing time of our B&P algorithm.

Table 5 Performance of dominance rules at the root node of the B&B tree

	Set Short			Set Normal			Set Large		
n	Label	Dominated	Ratio (%)	Label	Dominated	Ratio (%)	Label	Dominated	Ratio (%)
10	977	357	36.53	203	72	35.21	820	394	48.08
20	3685	1755	47.62	3696	1636	44.27	8733	5719	65.49
30	8002	4104	51.28	26726	16677	62.40	136613	105296	77.08
40	10457	5768	55.16	92656	59317	64.02	157579	122506	77.74
50	9720	5513	56.71	83562	52655	63.01	611115	480879	78.69
60	24419	14442	59.14	89326	52959	59.29	934360	726194	77.72
80	132254	80863	61.14	336025	168042	50.01	2036989	1395653	68.52
Avg.	27073	16114	52.51	90313	50194	54.03	555173	405234	70.47

6.4. Analyses of Cost Components and Objective Functions

In this section, we analyze some important features of PDPTW-IC. For all the optimally solved instances in each set, we calculate the average proportions of travel cost ($TC\%$), refrigeration cost ($RC\%$), and quality decay cost ($DC\%$) in the total cost, which are reported in Figure 8. From the results, we get some interesting observations:

(1) *In short-distance urban distribution (Set Short), a majority (64%) of the total cost comes from the travel cost, and only 11% is the quality decay cost.* This observation makes sense because instances in this set have a short time horizon, and thus products are picked up from and delivered to customers quickly. Consequently, perishable products can keep a high level of freshness when delivered, leading to a low quality decay cost.

(2) *In long-distance urban distribution (Set Normal) and trunk transportation (Set Large), the refrigeration cost becomes the most crucial part, and the quality decay cost also doubles.* As mentioned earlier, both traveling and waiting contribute to the refrigeration cost. For Set Normal instances, the high refrigeration cost mainly comes from the long waiting times at customers. For Set Large instances, since we have tripled the travel distances between nodes, the waiting times at customers are relatively shorter; thus, the proportion of the refrigeration cost is not as high as that in Set Normal instances. As waiting at customers or on the roads (due to traffic congestions) is quite common in real-world transportation activities, it is vital to take the refrigeration cost into consideration when making scheduling decisions for refrigerated vehicles. In addition, as the time horizon becomes longer, the quality decay cost also increases due to the limited lifespans of perishable products.

To the best of our knowledge, most 3PL companies in China take travel costs as the first or even the only consideration when planning logistics activities, which may increase the refrigeration and quality decay costs, leading to a high total cost. To validate the importance of a comprehensive objective function, we compare results under different objectives. For this purpose, we solve all the optimally solved instances again but with the objective of minimizing the travel cost, which

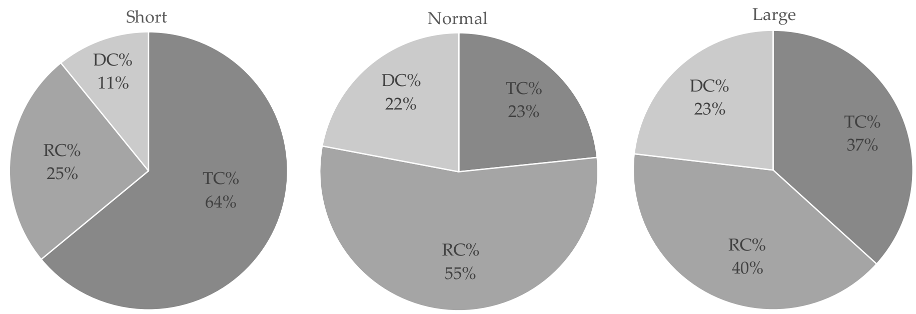

Figure 8 **The average TC%, RC%, and DC% of all optimally solved instances in each set**

can be implemented by setting $\rho_1 = 0$ and $\pi_i = 0, \forall i \in V_d$. We then fix the scheduling decisions and calculate the corresponding refrigeration cost, quality decay cost, and total cost. Detailed results under this objective function can be found in Tables C5–C7 in Appendix C. Comparisons of the total cost under different objective functions are reported in Table 6. Column *TRDC* gives the average result under our proposed objective function, and column *TCO* presents the average result with the objective of minimizing the travel cost. Columns *Rd (%)* and *#Rd* provide the percentage and quantity differences, respectively. They are calculated as *Rd (%) = (TRDC-TCO)/TCO* $\times$*100* and *#Rd = TRDC-TCO*. Thus, a negative value of *Rd (%)* or *#Rd* means that our proposed objective function has a lower total cost.

Table 6 **Comparisons of the total cost under different objective functions**

	Set Short				Set Normal				Set Large			
n	TRDC	TCO	Rd (%)	#Rd	TRDC	TCO	Rd (%)	#Rd	TRDC	TCO	Rd (%)	#Rd
10	390.78	422.54	-7.35	-31.76	926.12	1135.93	-19.06	-209.81	2768.07	3101.32	-10.71	-333.25
20	658.56	701.64	-5.74	-43.08	1287.17	1479.01	-12.86	-191.84	4194.32	5186.47	-19.15	-992.15
30	964.72	1027.16	-6.08	-62.44	1737.21	2009.55	-13.35	-272.34	4975.83	6338.93	-20.77	-1363.10
40	1416.32	1488.88	-4.69	-72.56	2430.93	2836.70	-14.03	-405.77	6778.71	8247.54	-17.74	-1468.83
50	1741.68	1828.88	-4.54	-87.20	3055.27	3589.80	-14.68	-534.54	8393.24	9977.40	-15.84	-1584.16
60	2127.90	2239.58	-4.76	-111.68	4122.18	4822.36	-14.29	-700.17	10813.65	13115.14	-17.66	-2301.48
80	2722.79	2863.87	-4.73	-141.07	5768.50	6617.01	-12.78	-848.51	15554.95	17802.88	-12.63	-2247.93
Avg.	1431.82	1510.36	-5.41	-78.54	2761.06	3212.91	-14.43	-451.85	7639.83	9109.95	-16.36	-1470.13

From Table 6, we observe that our comprehensive objective function can help reduce the total cost for all the instance sets. In particular, as the time horizon becomes longer, cost reduction becomes more significant, from 5.41% for Set Short to 16.36% for Set Large. For 3PL companies, the CCT activity is performed several times per week. Hence, the accumulated cost reduction over an entire year could be significant. To conclude, besides the travel cost, refrigeration and quality decay costs are also important factors that should be considered when planning CCT activities, especially long-distance transportation activities.

7. Conclusions

This study introduces a new cold chain transportation problem extended from the classic PDPTW, where multiple commodities with incompatibility constraints are considered. For the new variant, besides travel costs, refrigeration costs of vehicles and quality decay costs of perishable products are also explicitly incorporated into the objective function. To minimize the total cost, we focus on determining the optimal pickup and delivery routes and the vehicle departure times from the depot. We develop a set partitioning model for the problem, which is solved by a tailored B&P algorithm. To tackle the asymmetry issue arising from the pricing problem of the B&P framework, we develop a novel asymmetric bidirectional labeling algorithm. Benchmark instances based on real-world statistical data and known instance sets in the literature are first generated for this problem. Extensive numerical results show that the bidirectional labeling algorithm has better performance than the monodirectional labeling algorithm and that our B&P algorithm can solve most instances to optimality in an acceptable time frame. In addition, our proposed objective function can help reduce the total cost, compared to the widely adopted objective function minimizing only the travel cost.

Future research could be conducted from two aspects: (1) In this paper, we assume that the travel times between nodes are constant, whereas they are closely related to traffic conditions in real-world applications. Thus, considering time-dependent travel times can result in a more practical problem. (2) More powerful algorithms can be developed to solve large-size instances with a long time horizon more efficiently. For example, we can consider setting dynamic half-way points (i.e., values of parameters Θ^f and Θ^b) as in Tilk et al. (2017) to further improve the efficiency of the bidirectional labeling algorithm. We can also develop heuristic or metaheuristic algorithms.

Acknowledgements

This research was partially supported by the National Key R&D Program of China (Grant 2018YFB1700600) and the National Natural Science Foundation of China [Grants 72101049, 71971090, 71821001]. The authors are thankful to the associate editor and two anonymous referees for their helpful comments.

References

Amorim, Pedro, Bernardo Almada-Lobo. 2014. The impact of food perishability issues in the vehicle routing problem. *Computers & Industrial Engineering* **67** 223–233.

Baldacci, Roberto, Enrico Bartolini, Aristide Mingozzi. 2011a. An exact algorithm for the pickup and delivery problem with time windows. *Operations Research* **59**(2) 414–426.

Baldacci, Roberto, Aristide Mingozzi, Roberto Roberti. 2011b. New Route Relaxation and Pricing Strategies for the Vehicle Routing Problem. *Operations Research* **59**(5) 1269–1283.

Battarra, Maria, Michele Monaci, Daniele Vigo. 2009. An adaptive guidance approach for the heuristic solution of a minimum multiple trip vehicle routing problem. *Computers & Operations Research* **36**(11) 3041–3050.

Bettinelli, Andrea, Valentina Cacchiani, Teodor Gabriel Crainic, Daniele Vigo. 2019. A branch-and-cut-and-price algorithm for the multi-trip separate pickup and delivery problem with time windows at customers and facilities. *European Journal of Operational Research* **279**(3) 824–839.

Ceselli, Alberto, Giovanni Righini, Matteo Salani. 2009. A column generation algorithm for a rich vehicle-routing problem. *Transportation Science* **43** 56–69.

Chai, Xiaozhou. 2016. Barcode solutions help companies reduce food distribution losses. *Logistics and Material Handling* **8** 138–138.

Chen, Huey Kuo, Che Fu Hsueh, Mei Shiang Chang. 2009. Production scheduling and vehicle routing with time windows for perishable food products. *Computers & Operations Research* **36**(7) 2311–2319.

Cherkesly, Marilene, Guy Desaulniers, Gilbert Laporte. 2015. Branch-price-and-cut algorithms for the pickup and delivery problem with time windows and last-in-first-out loading. *Transportation Science* **49**(4) 752–766.

Cnwest. 2015. Why is fresh online shopping "not fresh"? cold chain distribution becomes a "pain point". https://society.huanqiu.com/article/9CaKrnJMvlp. Accessed April 18, 2021.

Cordeau, Jean-François. 2006. A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research* **54**(3) 573–586.

Cordeau, Jean-François, Gilbert Laporte, Stefan Ropke. 2008. Recent models and algorithms for one-to-one pickup and delivery problems. *The vehicle routing problem: latest advances and new challenges*. Springer, 327–357.

Costa, Luciano, Claudio Contardo, Guy Desaulniers. 2019. Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science* **53**(4) 946–985.

Dabia, Said, Stefan Ropke, Tom Van Woensel, Ton De Kok. 2013. Branch and price for the time-dependent vehicle routing problem with time windows. *Transportation Science* **47**(3) 380–396.

Desaulniers, Guy, Jacques Desrosiers, Marius M Solomon. 2005. *Column generation*, vol. 5. Springer.

Factorovich, Pablo, Isabel Méndez-Díaz, Paula Zabala. 2020. Pickup and delivery problem with incompatibility constraints. *Computers & Operations Research* **113** 104805.

Feillet, Dominique, Pierre Dejax, Michel Gendreau, Cyrille Gueguen. 2004. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. *Networks* **44**(3) 216–229.

Furtado, Maria Gabriela S., Pedro Munari, Reinaldo Morabito. 2017. Pickup and delivery problem with time windows: A new compact two-index formulation. *Operations Research Letters* **45**(4) 334–341.

Gendreau, Michel, Daniele Manerba, Renata Mansini. 2016. The multi-vehicle traveling purchaser problem with pairwise incompatibility constraints and unitary demands: A branch-and-price approach. *European Journal of Operational Research* **248**(1) 59–71.

Goeke, Dominik. 2019. Granular tabu search for the pickup and delivery problem with time windows and electric vehicles. *European Journal of Operational Research* **278**(3) 821–836.

Goksal, Fatma Pinar, Ismail Karaoglan, Fulya Altiparmak. 2013. A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. *Computers & Industrial Engineering* **65**(1) 39–53.

Gschwind, Timo, Stefan Irnich, Ann-Kathrin Rothenbächer, Christian Tilk. 2018. Bidirectional labeling in column-generation algorithms for pickup-and-delivery problems. *European Journal of Operational Research* **266**(2) 521–530.

Hsu, Chaug-Ing, Sheng-Feng Hung, Hui-Chieh Li. 2007. Vehicle routing problem with time-windows for perishable food delivery. *Journal of Food Engineering* **80**(2) 465–475.

Li, Haibing, Andrew Lim. 2008. A metaheuristic for the pickup and delivery problem with time windows. *International Journal on Artificial Intelligence Tools* **12**(02) 173–186.

Manerba, Daniele, Renata Mansini. 2015. A branch-and-cut algorithm for the multi-vehicle traveling purchaser problem with pairwise incompatibility constraints. *Networks* **65** 139–154.

Osvald, Ana, Lidija Zadnik Stirn. 2008. A vehicle routing algorithm for the distribution of fresh vegetables and similar perishable food. *Journal of Food Engineering* **85**(2) 285–295.

Parragh, Sophie N, Karl F Doerner, Richard F Hartl. 2008a. A survey on pickup and delivery problems Part I: Transportation between customers and depot. *Journal Für Betriebswirtschaft* **58**(1) 21–51.

Parragh, Sophie N, Karl F Doerner, Richard F Hartl. 2008b. A survey on pickup and delivery problems Part II: Transportation between pickup and delivery locations. *Journal Für Betriebswirtschaft* **58**(2) 81–117.

Pessoa, Artur, Ruslan Sadykov, Eduardo Uchoa, François Vanderbeck. 2020. A generic exact solver for vehicle routing and related problems. *Mathematical Programming* **183**(1) 483–523.

Pisinger, David, Stefan Ropke. 2007. A general heuristic for vehicle routing problems. *Computers & Operations Research* **34**(8) 2403–2435.

Qi, Chengming, Lishuan Hu. 2020. Optimization of vehicle routing problem for emergency cold chain logistics based on minimum loss. *Physical Communication* **40** 101085.

Qin, Hu, Xinxin Su, Teng Ren, Zhixing Luo. 2021. A review on the electric vehicle routing problems: Variants and algorithms. *Frontiers of Engineering Management* **8**(3) 370–389.

Righini, Giovanni, Matteo Salani. 2006. Symmetry helps: Bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints. *Discrete Optimization* **3**(3) 255–273.

Ropke, Stefan, Jean François Cordeau. 2009. Branch and cut and price for the pickup and delivery problem with time windows. *Transportation Science* **43**(3) 267–286.

Ropke, Stefan, Jean-François Cordeau, Gilbert Laporte. 2007. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks* **49**(4) 258–272.

Shanghai Municipal Quality and Technical Supervision Bureau. 2007. Technology and management specification of cold chain logistics (DB 31/T 388-2007). http://www.bjqtwl.com/html/shejiguifan/2014/0417/76.html, accessed April 18, 2021.

Solomon, Marius M. 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research* **35**(2) 254–265.

Song, Byung Duk, Young Dae Ko. 2016. A vehicle routing problem of both refrigerated- and general-type vehicles for perishable food products delivery. *Journal of Food Engineering* **169** 61–71.

Stellingwerf, Helena M, Leendert HC Groeneveld, Gilbert Laporte, Argyris Kanellopoulos, Jacqueline M Bloemhof, Behzad Behdani. 2021. The quality-driven vehicle routing problem: Model and application to a case of cooperative logistics. *International Journal of Production Economics* **231** 107849.

Tilk, Christian, Ann-Kathrin Rothenbächer, Timo Gschwind, Stefan Irnich. 2017. Asymmetry matters: Dynamic half-way points in bidirectional labeling for solving shortest path problems with resource constraints faster. *European Journal of Operational Research* **261**(2) 530–539.

Veenstra, Marjolein, Marilène Cherkesly, Guy Desaulniers, Gilbert Laporte. 2017. The pickup and delivery problem with time windows and handling operations. *Computers & Operations Research* **77** 127–140.

Wang, Chao, Dong Mu, Fu Zhao, John W. Sutherland. 2015. A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows. *Computers & Industrial Engineering* **83** 111–122.

Online Supplement

A. Proofs of Statements

This section provides completed proofs for theorems, propositions, and corollary presented in the main paper. Before giving the proofs, we first introduce the generic form of a piecewise linear convex function. In a two-dimensional coordinate system, a line can be expressed as $y = \tilde{u}x + \tilde{g}$, where $\tilde{u}$ and $\tilde{g}$ represent the slope and intercept, respectively. A piecewise linear convex function can be expressed as a generic form $F(x) = \max_{h \in H} \{\tilde{u}_h x + \tilde{g}_h\}$, where $H = \{1, 2, \ldots, |H|\}$ is the index set of lines and $\tilde{u}_{h-1} < \tilde{u}_h$, $\tilde{g}_{h-1} > \tilde{g}_h, \forall h \in \{2, \ldots, |H|\}$.

A.1. Proof of Theorem 1

We prove it by mathematical induction. Let k be the index of set $\{0, 1, \ldots, m+1\}$.

When $k = 0$, $r^f = (v_0)$. Then $\delta(t_0) = \max\{0 + t_0,\ e_0\}$ and $\delta(t_0) \in [e_0, l_0]$, satisfying relation (9).

We assume Theorem 1 holds for $k = m$. For the corresponding route $r^f = (v_0, v_1, \ldots, v_m = i)$, we denote the SST at node i as $\delta(t_0) = \max\{\omega + t_0,\ \alpha\}$. When $k = m+1$, the corresponding route is $r^f = (v_0, v_1, \ldots, v_m = i, v_{m+1} = j)$. we derive the range of the SST $\delta'(t_0)$ at node j as

$$\alpha' = \max\left\{e_j,\ \alpha + \sigma_{ij}\right\}, \quad \beta' = \max\{e_j,\ \min\{l_j,\ \beta + \sigma_{ij}\}\}.$$

If $\alpha' \le \beta'$, it is a feasible extension and we can get

$$\begin{aligned}
\delta'(t_0) &= \max\{e_j,\ \delta(t_0) + \sigma_{ij}\} \\
&= \max\{e_j,\ \max\{\omega + t_0,\ \alpha\} + \sigma_{ij}\} \\
&= \max\{e_j,\ \max\{\omega + t_0 + \sigma_{ij}\ ,\ \alpha + \sigma_{ij}\}\} \\
&= \max\{e_j,\ \omega + t_0 + \sigma_{ij},\ \alpha + \sigma_{ij}\} \\
&= \max\{\omega + \sigma_{ij} + t_0,\ \max\{e_j,\ \alpha + \sigma_{ij}\}\} \\
&= \max\{\omega + \sigma_{ij} + t_0,\ \alpha'\}.
\end{aligned}$$

Let $\omega' = \omega + \sigma_{ij}$, then $\delta'(t_0) = \max\{\omega' + t_0,\ \alpha'\}$, satisfying relation (9). □

A.2. Proof of Theorem 2

We prove it by mathematical induction. Let k be the index of set $\{0, 1, \ldots, m+1\}$.

When $k = 0$, route r^f only includes the starting depot 0. Then the quality decay cost is 0, and $\tau^d(t_0)$ can be written as $\tau^d(t_0) = \max_{h \in \{1\}} \{0 \cdot t_0 + 0\}$. Thus, Theorem 2 holds.

We assume Theorem 2 holds for $k = m$ and denote the quality decay cost of the corresponding route $r^f = (v_0, v_1, \ldots, v_m = i)$ as $\tau^d(t_0) = \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h\}$. When $k = m+1$, if node i can be extended to node $v_{m+1} = j$, we need to consider two cases for calculating the quality decay cost $\tau^{d'}(t_0)$ of route $r^f = (v_0, v_1, \ldots, v_m = i, v_{m+1} = j)$ given the SST $\delta'(t_0)$ at node j. For notational simplicity, let $\varsigma'_j \triangleq \varsigma'_{\varphi_j}$ in the following.

(1) If j is a pickup node or the ending depot, then

$$\tau^{d'}(t_0) = \tau^d(t_0) + 0 = \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h\}.$$

(2) If j is a delivery node, then

$$\begin{aligned}
\tau^{d'}(t_0) &= \tau^d(t_0) + \pi_j \max\{\delta'(t_0) - \varsigma'_j,\ 0\} \\
&= \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h\} + \pi_j \max\{\max\{\omega' + t_0,\ \alpha'\} - \varsigma'_j,\ 0\} \\
&= \max_{h \in \bar{\mathcal{H}}} \left\{\bar{u}_h t_0 + \bar{g}_h + \max\{\max\{\pi_j(\omega' - \varsigma'_j) + \pi_j t_0,\ \pi_j(\alpha' - \varsigma'_j)\},\ 0\}\right\} \\
&= \max_{h \in \bar{\mathcal{H}}} \left\{\max\{\max\{\pi_j(\omega' - \varsigma'_j) + \bar{g}_h + (\pi_j + \bar{u}_h)t_0,\ \bar{u}_h t_0 + \bar{g}_h + \pi_j(\alpha' - \varsigma'_j)\},\ \bar{u}_h t_0 + \bar{g}_h\}\right\} \\
&= \max_{h \in \bar{\mathcal{H}}} \left\{\pi_j(\omega' - \varsigma'_j) + \bar{g}_h + (\pi_j + \bar{u}_h)t_0,\ \bar{u}_h t_0 + \bar{g}_h + \pi_j(\alpha' - \varsigma'_j),\ \bar{u}_h t_0 + \bar{g}_h\right\}.
\end{aligned}$$

Thus, Theorem 2 holds for $k = m + 1$. □

A.3. Proof of Corollary 1

We use Theorems 1 and 2 to prove Corollary 1. After obtaining $\tau^d(t_0)$ and $\delta(t_0)$ of route $r^f = (v_0, v_1, \ldots, v_m = i)$, we can let $\tau^d(t_0) = \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h\}$ and calculate $\tau^f(t_0)$ as

$$\begin{aligned}
\tau^f(t_0) &= \tau^d(t_0) + \rho_1(\delta(t_0) - t_0) \\
&= \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h\} + \rho_1 \left(\max\{\omega + t_0,\ \alpha\} - t_0\right) \\
&= \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h\} + \max\{\rho_1 \omega,\ \rho_1 \alpha - \rho_1 t_0\} \\
&= \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \bar{g}_h + \max\{\rho_1 \omega,\ \rho_1 \alpha - \rho_1 t_0\}\} \\
&= \max_{h \in \bar{\mathcal{H}}} \{\max\{\bar{u}_h t_0 + \rho_1 \omega + \bar{g}_h,\ (\bar{u}_h - \rho_1)t_0 + \rho_1 \alpha + \bar{g}_h\}\} \\
&= \max_{h \in \bar{\mathcal{H}}} \{\bar{u}_h t_0 + \rho_1 \omega + \bar{g}_h,\ (\bar{u}_h - \rho_1)t_0 + \rho_1 \alpha + \bar{g}_h\},
\end{aligned}$$

satisfying the form of a piecewise linear convex function. □

A.4. Proof of Theorem 3

We prove it by mathematical induction. Let k be the index of set $\{1, \ldots, m+1\}$.

When $k = 1$, $r^b = (v_0, v_1)$ and v_1 must be a delivery node. Then we have

$$\tau^c(t_{v_1}) = (\rho_1 + \pi_{v_1})t_{v_1},\ t_{v_1} \in [e_{v_1}, \min\{l_{v_1}, l_{v_0} - \sigma_{v_1 v_0}\}].$$

Thus, Theorem 3 holds for $k = 1$.

We assume Theorem 3 holds for $k = m$. For the corresponding route $r^b = (v_0, v_1, \ldots, v_m = i)$, we denote the cost evaluation function as $\tau^c(t_i) = \max_{h \in \hat{\mathcal{H}}} \{\hat{u}_h t_i + \hat{g}_h\}, t_i \in [\alpha, \beta]$. When $k = m+1$, the corresponding route is $r^b = (v_0, v_1, \ldots, v_m = i, v_{m+1} = j)$ and we derive the domain of the SST t_j at node j as

$$\alpha' = e_j,\ \beta' = \min\{l_j,\ \beta - \sigma_{ji}\}.$$

Node i can be extended to node j if $\alpha' \le \beta'$. Now given the SST t_j at node j, the earliest SST at node i will be $\max\{t_j + \sigma_{ji}, e_i\}$. We consider two cases to calculate the cost evaluation function $\tau^{c'}(t_j)$ of route $r^b = (v_0, v_1, \dots, v_m = i, v_{m+1} = j)$:

(1) If j is a pickup node or the starting depot, then

$$\begin{aligned}
\tau^{c'}(t_j) &= \tau^c\left(\max\left\{t_j + \sigma_{ji},\ e_i\right\}\right) \\
&= \max_{h\in\hat{\mathcal{H}}}\left\{\hat{u}_h\left(\max\{t_j + \sigma_{ji},\ e_i\}\right) + \hat{g}_h\right\} \\
&= \max_{h\in\hat{\mathcal{H}}}\left\{\max\left\{\hat{u}_h t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h,\ \hat{u}_h e_i + \hat{g}_h\right\}\right\} \\
&= \max_{h\in\hat{\mathcal{H}}}\left\{\hat{u}_h t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h,\ \hat{u}_h e_i + \hat{g}_h\right\}.
\end{aligned}$$

(2) If j is a delivery node, then

$$\begin{aligned}
\tau^{c'}(t_j) &= \tau^c\left(\max\left\{t_j + \sigma_{ji},\ e_i\right\}\right) + \pi_j \max\{t_j - \varsigma'_j,\ 0\} \\
&= \max_{h\in\hat{\mathcal{H}}}\left\{\hat{u}_h t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h,\ \hat{u}_h e_i + \hat{g}_h\right\} + \max\{\pi_j t_j - \pi_j\varsigma'_j,\ 0\} \\
&= \max_{h\in\hat{\mathcal{H}}}\Big\{\hat{u}_h t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h + \max\{\pi_j t_j - \pi_j\varsigma'_j,\ 0\},\ \hat{u}_h e_i + \hat{g}_h + \max\{\pi_j t_j - \pi_j\varsigma'_j,\ 0\}\Big\} \\
&= \max_{h\in\hat{\mathcal{H}}}\Big\{\max\{(\pi_j + \hat{u}_h)t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h - \pi_j\varsigma'_j,\ \hat{u}_h t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h\}, \\
&\qquad \max\{\pi_j t_j + \hat{u}_h e_i + \hat{g}_h - \pi_j\varsigma'_j,\ \hat{u}_h e_i + \hat{g}_h\}\Big\} \\
&= \max_{h\in\hat{\mathcal{H}}}\Big\{(\pi_j + \hat{u}_h)t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h - \pi_j\varsigma'_j,\ \hat{u}_h t_j + \hat{u}_h\sigma_{ji} + \hat{g}_h, \\
&\qquad \pi_j t_j + \hat{u}_h e_i + \hat{g}_h - \pi_j\varsigma'_j,\ \hat{u}_h e_i + \hat{g}_h\Big\}.
\end{aligned}$$

Thus, both cases satisfy the form of a piecewise linear convex function. □

A.5. Proof of Proposition 2

The relationship $\Omega(L_2) \subseteq \Omega(L_1)$ holds due to conditions (1)–(3). Let ξ be one of the feasible completions of L_2. For the partial backward path ξ, we assume that (i) the range of the SST t_i at the extension ending node i is $[\alpha^b, \beta^b]$; (ii) the difference between the accumulated dual value and the accumulated travel cost is C^b; and (iii) the sum of the refrigeration and quality decay costs is $\tau^b(t_i)$. Now we show that $\bar{c}(\mathbb{P}(L_2) \otimes \xi) \ge \bar{c}(\mathbb{P}(L_1) \otimes \xi), \forall \xi \in \Omega(L_2)$:

$$\begin{aligned}
\bar{c}(\mathbb{P}(L_1) \otimes \xi) &= \min_{\max\{\alpha_1, \alpha^b\} \le t_i \le \min\{\beta_1, \beta^b\}} \{\eta_1(t_i) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} - C_1 - C^b + \mu_i, \\
\bar{c}(\mathbb{P}(L_2) \otimes \xi) &= \min_{\max\{\alpha_2, \alpha^b\} \le t_i \le \min\{\beta_2, \beta^b\}} \{\eta_2(t_i) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} - C_2 - C^b + \mu_i.
\end{aligned}$$

Subsequently, we have

$$\begin{aligned}\bar{c}(\mathbb{P}(L_2)\otimes\xi)-\bar{c}(\mathbb{P}(L_1)\otimes\xi) &= \min_{\max\{\alpha_2,\alpha^b\}\le t_i\le\min\{\beta_2,\beta^b\}}\{\eta_2(t_i)+\tau^b(t_i)-\pi_i t_i\mathbb{1}(i\in V_d)\}-C_2\\ &- \min_{\max\{\alpha_1,\alpha^b\}\le t_i\le\min\{\beta_1,\beta^b\}}\{\eta_1(t_i)+\tau^b(t_i)-\pi_i t_i\mathbb{1}(i\in V_d)\}+C_1\\ &\ge \min_{\max\{\alpha_2,\alpha^b\}\le t_i\le\min\{\beta_2,\beta^b\}}\{\eta_2(t_i)-\eta_1(t_i)\}-C_2+C_1\\ &\ge 0. \qquad \square\end{aligned}$$

A.6. Proof of Proposition 3

We only discuss the case of $\beta_2>\alpha_2$ since it is easy to compare $\eta_1(t_i)$ and $\eta_2(t_i)$ when $\beta_2=\alpha_2$. $\delta(t_i)=t_i+\omega,\forall\delta(t_i)\in(\alpha,\beta]$ when $\alpha<\beta$; thus, we have

$$\begin{aligned}\phi_1(t_i)&=\delta_1^{-1}(t_i)=t_i-\omega_1 &&\forall t_i\in(\alpha_1,\beta_1],\\ \phi_2(t_i)&=\delta_2^{-1}(t_i)=t_i-\omega_2 &&\forall t_i\in(\alpha_2,\beta_2].\end{aligned}$$

Subsequently,

$$\phi_1(t_i)-\phi_2(t_i)=\delta_1^{-1}(t_i)-\delta_2^{-1}(t_i)=\omega_2-\omega_1 \qquad \forall t_i\in(\alpha_2,\beta_2].$$

Thus, condition (4) in Proportion 2 can be rewritten as

$$\begin{aligned}&\eta_2(t_i)-C_2\ge\eta_1(t_i)-C_1 && \forall t_i\in(\alpha_2,\beta_2]\\ \Longrightarrow\ &\tau_2^f(\phi_2(t_i))-\tau_1^f(\phi_1(t_i))\ge C_2-C_1 && \forall t_i\in(\alpha_2,\beta_2]\\ \Longrightarrow\ &\tau_2^f(\delta_2^{-1}(t_i))-\tau_1^f(\delta_1^{-1}(t_i))\ge C_2-C_1 && \forall t_i\in(\alpha_2,\beta_2]\\ \Longrightarrow\ &\tau_2^f(\delta_2^{-1}(t_i))-\tau_1^f(\delta_2^{-1}(t_i)+\omega_2-\omega_1)\ge C_2-C_1 && \forall t_i\in(\alpha_2,\beta_2]\\ \Longrightarrow\ &\tau_2^f(t_0)-\tau_1^f(t_0+\omega_2-\omega_1)\ge C_2-C_1 && \forall t_0\in(\alpha_2-\omega_2,\beta_2-\omega_2]\\ \Longrightarrow\ &\tau_2^f(t_0)-\bar{\tau}_1^f(t_0)\ge C_2-C_1 && \forall t_0\in(\alpha_2-\omega_2,\beta_2-\omega_2],\end{aligned}$$

where the second to last equation is obtained by using t_0 to replace $\delta_2^{-1}(t_i)$ and $\bar{\tau}_1^f(t_0)$ is obtained by $\bar{\tau}_1^f(t_0)=\tau_1^f(t_0+\omega_2-\omega_1)$. Since $\bar{\tau}_1^f(t_0)$ and $\tau_2^f(t_0)$ are continuous on $[\alpha_2-\omega_2,\beta_2-\omega_2]$, we have

$$\lim_{t_0\to(\alpha_2-\omega_2)^+}(\tau_2^f(t_0)-\bar{\tau}_1^f(t_0))=\tau_2^f(\alpha_2-\omega_2)-\bar{\tau}_1^f(\alpha_2-\omega_2).$$

Thus, we can obtain $\tau_2^f(t_0)-\bar{\tau}_1^f(t_0)\ge C_2-C_1$, $\forall t_0\in[\alpha_2-\omega_2,\beta_2-\omega_2]$ when $\alpha_2<\beta_2$. $\square$

A.7. Proof of Proposition 4

The relationship $\Omega(L_2)\subseteq\Omega(L_1)$ holds due to conditions (1)–(3). Let ξ be one of the feasible completions of L_2. For the partial forward path ξ, we assume that (i) the range of the SST t_i at the extension ending node i is $[\alpha^f,\beta^f]$; (ii) the difference between the accumulated dual value and the accumulated travel cost is C^f; and (3) the sum of the refrigeration and quality decay costs is $\eta(t_i)$. Now we show that $\bar{c}(\mathbb{P}(L_2) \otimes \xi) \geq \bar{c}(\mathbb{P}(L_1) \otimes \xi), \forall \xi \in \Omega(L_2)$:

$$\bar{c}(\mathbb{P}(L_1) \otimes \xi) = \min_{\max\{\alpha_1,\alpha^f\} \leq t_i \leq \min\{\beta_1,\beta^f\}} \{\eta(t_i) + \tau_1^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} - C^f - C_1 + \mu_i,$$
$$\bar{c}(\mathbb{P}(L_2) \otimes \xi) = \min_{\max\{\alpha_2,\alpha^f\} \leq t_i \leq \min\{\beta_2,\beta^f\}} \{\eta(t_i) + \tau_2^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} - C^f - C_2 + \mu_i.$$

Subsequently, we have

$$\begin{aligned}
\bar{c}(\mathbb{P}(L_2) \otimes \xi) - \bar{c}(\mathbb{P}(L_1) \otimes \xi) = & \min_{\max\{\alpha_2,\alpha^f\} \leq t_i \leq \min\{\beta_2,\beta^f\}} \{\eta(t_i) + \tau_2^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} - C_2 \\
& - \min_{\max\{\alpha_2,\alpha^f\} \leq t_i \leq \min\{\beta_2,\beta^f\}} \{\eta(t_i) + \tau_1^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d)\} + C_1 \\
& \geq \min_{\max\{\alpha_2,\alpha^f\} \leq t_i \leq \min\{\beta_2,\beta^f\}} \{\tau_2^b(t_i) - \tau_1^b(t_i)\} + C_1 - C_2 \\
& = \min_{\max\{\alpha_2,\alpha^f\} \leq t_i \leq \min\{\beta_2,\beta^f\}} \{\tau_2^c(t_i) - \rho_1 t_i - (\tau_1^c(t_i) - \rho_1 t_i)\} + C_1 - C_2 \\
& = \min_{\max\{\alpha_2,\alpha^f\} \leq t_i \leq \min\{\beta_2,\beta^f\}} \{\tau_2^c(t_i) - \tau_1^c(t_i)\} + C_1 - C_2 \\
& \geq 0. \qquad \square
\end{aligned}$$

A.8. Proof of Proposition 5

When $t_i \in (\alpha^f, \min\{\beta^f, \beta^b\}]$, we have

$$\begin{aligned}
F(t_i) &= \eta(t_i) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d) \\
&= \tau^f(\delta^{-1}(t_i)) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d) \\
&= \tau^f(t_i - \omega) + \tau^b(t_i) - \pi_i t_i \mathbb{1}(i \in V_d) \\
&= \max_{\mathcal{F}_h \in \mathcal{F}} \{u_h(t_i - \omega) + g_h\} + \max_{\mathcal{C}_h \in \mathcal{C}} \{(\hat{u}_h - \rho_1)t_i + \hat{g}_h\} - \pi_i t_i \mathbb{1}(i \in V_d) \\
&= \max_{\mathcal{F}_h \in \mathcal{F}} \left\{ u_h(t_i - \omega) + g_h + \max_{\mathcal{C}_h \in \mathcal{C}} \{(\hat{u}_h - \rho_1)t_i + \hat{g}_h\} - \pi_i t_i \mathbb{1}(i \in V_d) \right\} \\
&= \max_{\mathcal{F}_h \in \mathcal{F}} \left\{ \max_{\mathcal{C}_h \in \mathcal{C}} \{(u_h + \hat{u}_h - \rho_1 - \pi_i \mathbb{1}(i \in V_d))t_i + g_h + \hat{g}_h - u_h \omega\} \right\}. \qquad \square
\end{aligned}$$

B. Derivation of Function $\eta(t_i)$

This section derives the mathematical expression of function $\eta(t_i)$. Note that the SST at node i can also be expressed as $\delta(t_0)$, i.e., $t_i = \delta(t_0)$. Based on the definition of α_{v_k}, we also have $\alpha = \alpha_i$. For another forward route $r^{f'} = (v_0, v_1, \ldots, v_k = j), k < m$, we denote the SST at node j as $t_j = \delta_j(t_0) = \max\{\omega_j + t_0,\ \alpha_j\}$, whose range is $[\alpha_j, \beta_j]$. Now we derive $\eta(t_i)$:

Case 1: $\alpha = \beta$. We consider two situations based on if there exists an index $k \in [0, m)$ satisfying $(\alpha_{v_k} < \beta_{v_k}) \wedge (\alpha_{v_{k+1}} = \beta_{v_{k+1}})$: (1) Such an index k exists. We denote $\tau^{d'}(t_0)$ as the quality decay cost of route $r^{f'}$. When $t_j = \delta_j(t_0) = \beta_j$, the corresponding t_0 will be $\beta_j - \omega_j$, which is the latest departure time from depot 0 such that the time window constraints of customers in route $r^{f'}$ can be satisfied. Then $\eta(t_i)$ can be computed as

$$\begin{aligned}\eta(t_i) &= \min_{t_0\in[e_0,\beta_j-\omega_j]}\left(\tau^{d'}(t_0)+\rho_1(\alpha-t_0)+\sum_{z\in\Delta_d}\pi_z\alpha_z\right)\\ &= \min_{t_0\in[e_0,\beta_j-\omega_j]}\left(\tau^{d'}(t_0)-\rho_1 t_0\right)+\rho_1\alpha+\sum_{z\in\Delta_d}\pi_z\alpha_z\ ,\end{aligned} \tag{B.1}$$

where Δ_d is the set of delivery nodes visited after j in route r^f; (2) Such an index k does not exist. Then, the SST at all nodes of r^f are fixed and we have

$$\eta(t_i) = \rho_1(\alpha-\alpha_0)+\sum_{z\in\Delta_f}\pi_z\alpha_z\ . \tag{B.2}$$

where Δ_f is the set of delivery nodes visited in route r^f.

Case 2: $\alpha<\beta$. We first derive the inverse function of $\delta(t_0)$. When $\alpha-\omega>0$, the inverse function $\delta^{-1}(\alpha)$ can take an infinite number of values (see Figure 4 in the main paper), which does not meet the definition of a function. Thus, we define function $\phi(t_i)$ to denote the SST at node v_0 given t_i, expressed as

$$\phi(t_i)=\begin{cases}\arg\min\limits_{t_0\in[0,\ \alpha-\omega]}\tau^f(t_0) & \text{if } t_i=\alpha, \quad \text{(B.3a)}\\ \delta^{-1}(t_i) & \text{if } t_i>\alpha. \quad \text{(B.3b)}\end{cases}$$

When $t_i=\alpha$, we set $\phi(t_i)$ as that in Equation (B.3a) because taking other values will lead to a larger value of $\tau^f(t_0)$. Then we can calculate $\eta(t_i)$ as

$$\eta(t_i)=\tau^f(\phi(t_i)). \tag{B.4}$$

C. Data and Detailed Results

Table C1 Details of perishable commodities

Commodity	Bean sprout	Lettuce	Chinese cabbage	Litchi	Bayberry	Mushroom
$P''-P'$ (RMB/kg)	1.25	1.28	3.21	7.1	12.2	6.36
$\varsigma''-\varsigma'$ (days)	2∼3	2∼3	2∼3	2∼3	2∼3	about 3

In Tables C2–C7, n is the number of customers. #*R* is the number of routes contained in the final solution. *RLB* is the lower bound at the root node. *UB* is the final (feasible) solution produce by B&P within the time limit. *Gap* is the percentage difference between *RLB* and *UB*, which is calculated as *(UB - RLB)/UB*×*100*. *Node* reports the number of nodes explored in the B&B tree. *Time* is the computing time of B&P. The next six columns present the travel cost (*TC*), refrigeration cost (*RC*), quality decay cost (*DC*), and their proportions in the total cost. *Label* and *Dominated* are the numbers of generated and dominated labels at the root node of the B&B tree, respectively. *Ratio (%)* provides the percentage of dominated labels, computed as *Ratio (%)* = *Dominated/Label*×*100*. In Tables C5–C7, *TCO* is the sum of TC, RC, and DC.

Table C2 Results for instances in Set Short (minimizing a comprehensive objective function)

n	#R	RLB	UB	Gap	Node	Time	TC	RC	DC	TC%	RC%	DC%	Label	Dominated	Ratio (%)
10	2	411.66	411.66	0.00	1	0.10	223.08	90.95	97.63	54.19	22.09	23.72	183	63	34.43
10	2	420.29	420.29	0.00	1	0.02	219.62	94.01	106.66	52.25	22.37	25.38	431	149	34.57
10	2	409.31	409.31	0.00	1	0.03	223.79	86.52	99.01	54.67	21.14	24.19	662	228	34.44
10	2	345.35	366.31	5.72	3	0.06	219.14	85.03	62.14	59.82	23.21	16.96	1078	362	33.58
10	2	516.22	516.22	0.00	1	0.01	216.72	89.73	209.78	41.98	17.38	40.64	361	101	27.98
10	2	346.93	346.93	0.00	1	0.01	210.36	79.59	56.97	60.63	22.94	16.42	179	61	34.08
10	2	424.13	424.13	0.00	1	0.01	210.53	79.60	134.01	49.64	18.77	31.60	402	101	25.12
10	1	231.39	231.39	0.00	1	0.13	122.89	61.78	46.73	53.11	26.70	20.20	4517	1789	39.61
20	3	725.58	725.58	0.00	1	0.01	470.66	178.44	76.48	64.87	24.59	10.54	893	296	33.15
20	4	723.05	723.05	0.00	1	0.04	481.06	179.46	62.53	66.53	24.82	8.65	2691	977	36.31
20	4	772.38	772.38	0.00	1	0.04	477.23	175.35	119.80	61.79	22.70	15.51	2563	1170	45.65
20	3	622.37	622.37	0.00	1	0.06	379.85	170.04	72.48	61.03	27.32	11.65	3123	1602	51.30
20	4	730.36	730.36	0.00	1	0.03	431.01	177.66	121.69	59.01	24.32	16.66	1967	711	36.15
20	4	662.64	662.64	0.00	1	0.02	463.64	157.71	41.29	69.97	23.80	6.23	1319	486	36.85
20	2	469.35	469.35	0.00	1	0.07	254.88	132.62	81.85	54.30	28.26	17.44	5269	2543	48.26
20	3	511.14	562.75	9.17	11	1.33	353.97	138.60	70.18	62.90	24.63	12.47	11652	6251	53.65
30	7	1142.41	1188.09	3.84	21	0.14	816.63	299.23	72.24	68.73	25.19	6.08	1279	440	34.40
30	6	1061.70	1061.70	0.00	1	0.04	735.58	265.74	60.39	69.28	25.03	5.69	3147	1199	38.10
30	5	1019.63	1019.63	0.00	1	0.09	643.51	250.39	125.73	63.11	24.56	12.33	6136	2619	42.68
30	4	886.03	893.20	0.80	19	2.39	571.39	256.76	65.05	63.97	28.75	7.28	18283	10796	59.05
30	5	991.48	991.48	0.00	1	0.05	623.50	246.29	121.69	62.89	24.84	12.27	3747	1523	40.65
30	6	980.65	980.65	0.00	1	0.03	701.40	237.95	41.29	71.52	24.26	4.21	2435	918	37.70
30	4	725.37	764.81	5.16	3	0.17	483.22	222.10	59.49	63.18	29.04	7.78	9057	4006	44.23
30	4	766.60	818.21	6.31	31	2.57	539.11	215.71	63.39	65.89	26.36	7.75	19934	11328	56.83
40	10	1838.95	1878.65	2.11	5	0.03	1334.06	458.88	85.71	71.01	24.43	4.56	1415	476	33.64
40	7	1393.38	1419.37	1.83	3	0.06	972.59	369.49	77.29	68.52	26.03	5.45	3422	1404	41.03
40	7	1346.38	1410.81	4.57	33	0.98	921.25	335.83	153.73	65.30	23.80	10.90	6663	3322	49.86
40	6	1153.81	1190.64	3.09	63	5.33	784.27	330.78	75.59	65.87	27.78	6.35	19455	12494	64.22
40	8	1561.40	1561.40	0.00	1	0.05	1046.53	376.81	138.06	67.03	24.13	8.84	4067	1564	38.46
40	8	1358.52	1378.10	1.42	3	0.14	970.19	350.11	57.80	70.40	25.41	4.19	3975	1658	41.71
40	6	1187.16	1216.67	2.43	5	0.51	822.16	329.68	64.82	67.57	27.10	5.33	14156	6927	48.93
40	6	1207.42	1274.94	5.30	1273	56.29	864.78	323.85	86.30	67.83	25.40	6.77	30500	18299	60.00
50	12	2152.41	2237.43	3.80	63	0.34	1568.68	537.89	130.85	70.11	24.04	5.85	1594	534	33.50
50	9	1753.96	1753.96	0.00	1	0.05	1214.55	460.21	79.20	69.25	26.24	4.52	4466	1732	38.78
50	8	1579.65	1703.14	7.25	2513	39.22	1177.79	429.02	96.33	69.15	25.19	5.66	4591	2405	52.39
50	7	1363.36	1392.42	2.09	71	10.59	931.32	400.61	60.48	66.88	28.77	4.34	22583	14936	66.14
50	10	1875.73	1875.73	0.00	1	0.04	1306.75	471.37	97.62	69.67	25.13	5.20	3915	1614	41.23
50	9	1785.96	1785.96	0.00	1	0.06	1220.36	440.04	125.57	68.33	24.64	7.03	5374	2164	40.27
50	8	1476.90	1588.56	7.03	1273	33.68	1105.21	417.91	65.44	69.57	26.31	4.12	9911	5141	51.87
50	7	1474.30	1596.21	7.64	133117	5911.01	1059.79	401.96	134.45	66.39	25.18	8.42	25329	15577	61.50
60	14	2685.41	2740.03	1.99	11	0.18	1783.61	640.07	316.35	65.09	23.36	11.55	3051	942	30.88
60	12	2346.77	2346.77	0.00	1	0.07	1592.38	573.23	181.16	67.85	24.43	7.72	5861	2280	38.90
60	9	1815.77	1939.26	6.37	5565	244.66	1298.72	525.47	115.06	66.97	27.10	5.93	14920	8317	55.74
60	7	1495.64	1524.69	1.91	3	2.32	977.18	451.90	95.62	64.09	29.64	6.27	61833	43078	69.67
60	12	2217.80	2217.80	0.00	1	0.10	1531.39	570.79	115.62	69.05	25.74	5.21	7688	3096	40.27
60	12	2406.33	2476.73	2.84	947	22.43	1626.53	550.62	299.58	65.67	22.23	12.10	10205	3623	35.50
60	10	1876.50	1928.30	2.69	117	11.46	1273.50	508.03	146.77	66.04	26.35	7.61	28532	16006	56.10
60	8	1754.34	1849.61	5.15	16011	5034.24	1166.99	479.11	203.52	63.09	25.90	11.00	63264	38193	60.37
80	19	3648.41	3648.41	0.00	1	0.13	2402.70	890.91	354.80	65.86	24.42	9.72	8589	2316	26.96
80	15	2959.71	2959.71	0.00	1	0.67	1959.11	757.52	243.08	66.19	25.59	8.21	27590	10100	36.61
80	12	2466.63	2557.27	3.54	6901	3569.47	1580.40	664.44	312.42	61.80	25.98	12.22	102222	59083	57.80
80	9	2033.06	2073.92	1.97	1723	6851.87	1298.39	611.06	164.47	62.61	29.46	7.93	445913	303712	68.11
80	14	2846.98	2846.98	0.00	1	0.89	1854.51	767.96	224.51	65.14	26.97	7.89	33598	14039	41.79
80	13	2707.42	2778.38	2.55	181	69.72	1781.91	686.78	309.68	64.13	24.72	11.15	44353	18551	41.83
80	12	2536.04	2568.51	1.26	473	452.71	1608.85	677.89	281.77	62.64	26.39	10.97	76208	42068	55.20
80	11	2325.26	2349.17	1.02	213	961.05	1425.48	616.19	307.50	60.68	26.23	13.09	319561	197031	61.66

Table C3 Results for instances in Set Normal (minimizing a comprehensive objective function)

n	#R	RLB	UB	Gap	Node	Time	TC	RC	DC	TC%	RC%	DC%	Label	Dominated	Ratio (%)
10	2	994.47	994.47	0.00	1	0.01	108.73	581.21	304.53	10.93	58.44	30.62	104	15	14.42
10	2	920.60	920.60	0.00	1	0.01	130.58	317.58	472.44	14.18	34.50	51.32	283	90	31.80
10	2	663.27	663.27	0.00	1	0.01	129.66	366.69	166.92	19.55	55.29	25.17	246	108	43.90
10	3	634.96	634.96	0.00	1	0.01	166.83	285.08	183.05	26.27	44.90	28.83	300	118	39.33
10	2	1052.30	1052.30	0.00	1	0.01	107.65	436.10	508.56	10.23	41.44	48.33	156	60	38.46
10	2	1230.43	1230.43	0.00	1	0.01	117.44	601.30	511.69	9.54	48.87	41.59	130	22	16.92
10	2	986.71	986.71	0.00	1	0.01	114.28	388.74	483.69	11.58	39.40	49.02	148	47	31.76
10	2	1024.04	1024.04	0.00	1	0.01	105.10	409.88	509.07	10.26	40.03	49.71	198	71	35.86
10	2	828.34	828.34	0.00	1	0.01	114.17	315.35	398.82	13.78	38.07	48.15	264	113	42.80
20	4	1271.89	1271.89	0.00	1	0.01	278.31	769.90	223.68	21.88	60.53	17.59	792	290	36.62
20	4	1378.74	1405.97	1.94	21	0.88	350.46	646.74	408.77	24.93	46.00	29.07	3442	1230	35.74
20	3	1353.51	1359.91	0.47	9	0.39	293.63	677.47	388.81	21.59	49.82	28.59	5255	1910	36.35
20	4	1101.82	1107.88	0.55	23	4.14	307.67	613.32	186.88	27.77	55.36	16.87	14769	7109	48.13
20	5	1251.50	1251.50	0.00	1	0.01	363.46	699.95	188.08	29.04	55.93	15.03	797	314	39.40
20	3	1324.42	1324.42	0.00	1	0.01	239.20	855.98	229.24	18.06	64.63	17.31	545	197	36.15
20	4	1432.95	1432.95	0.00	1	0.01	369.59	720.03	343.32	25.79	50.25	23.96	1145	485	42.36
20	3	1199.88	1199.88	0.00	1	0.03	236.91	729.06	233.91	19.74	60.76	19.49	2081	842	40.46
20	3	1230.13	1230.13	0.00	1	0.06	292.33	680.13	257.68	23.76	55.29	20.95	4434	2348	52.95
30	6	1792.24	1792.24	0.00	1	0.02	427.76	1140.80	223.68	23.87	63.65	12.48	1825	809	44.33
30	6	1858.94	1858.94	0.00	1	0.06	453.48	989.56	415.91	24.39	53.23	22.37	3944	1852	46.96
30	6	1766.19	1766.19	0.00	1	3.03	444.37	953.65	368.17	25.16	53.99	20.85	60488	36462	60.28
30	6	1471.01	1471.01	0.00	1	10.05	398.62	885.51	186.88	27.10	60.20	12.70	123985	86852	70.05
30	8	1699.17	1699.17	0.00	1	0.02	487.48	1023.61	188.08	28.69	60.24	11.07	2072	995	48.02
30	5	1880.95	1880.95	0.00	1	0.02	388.20	1263.52	229.24	20.64	67.17	12.19	2097	1019	48.59
30	5	1792.18	1792.18	0.00	1	0.04	424.82	1024.03	343.32	23.70	57.14	19.16	3198	1728	54.03
30	4	1700.05	1700.05	0.00	1	0.42	349.21	1116.93	233.91	20.54	65.70	13.76	14352	6077	42.34
30	7	1674.12	1674.12	0.00	1	1.19	483.52	929.23	261.37	28.88	55.51	15.61	28576	14295	50.02
40	8	2542.21	2542.21	0.00	1	0.02	593.88	1676.49	271.83	23.36	65.95	10.69	2053	934	45.49
40	9	2517.29	2544.52	1.07	111	6.62	693.18	1392.17	459.17	27.24	54.71	18.05	14996	6489	43.27
40	7	2343.36	2349.76	0.27	9	8.19	543.01	1341.04	465.72	23.11	57.07	19.82	74042	40761	55.05
40	6	1998.81	2005.15	0.32	37	869.75	497.67	1243.71	263.76	24.82	62.03	13.15	650904	438654	67.39
40	8	2496.22	2507.60	0.45	7	0.20	825.98	1443.01	238.61	32.94	57.55	9.52	5135	2484	48.37
40	10	2841.19	2841.19	0.00	1	0.02	739.05	1758.14	344.00	26.01	61.88	12.11	2016	919	45.59
40	9	2577.41	2577.41	0.00	1	0.07	708.84	1451.58	416.99	27.50	56.32	16.18	4726	2324	49.17
40	8	2374.29	2374.29	0.00	1	0.30	616.60	1485.42	272.27	25.97	62.56	11.47	13247	5772	43.57
40	6	2136.28	2136.28	0.00	1	3.92	525.33	1311.90	299.06	24.59	61.41	14.00	66785	35518	53.18
50	8	3324.36	3324.36	0.00	1	0.02	740.77	2069.71	513.88	22.28	62.26	15.46	1671	786	47.04
50	11	3029.92	3029.92	0.00	1	0.17	861.86	1815.66	352.40	28.44	59.92	11.63	9261	4284	46.26
50	10	2921.37	2921.37	0.00	1	4.44	873.03	1703.40	344.95	29.88	58.31	11.81	60721	34893	57.46
50	8	2511.71	2514.60	0.11	5	268.71	617.06	1549.61	347.93	24.54	61.62	13.84	594800	388909	65.38
50	10	3250.16	3250.16	0.00	1	0.07	748.51	1915.17	586.48	23.03	58.93	18.04	5403	2366	43.79
50	12	3598.08	3598.08	0.00	1	0.04	934.38	1985.35	678.36	25.97	55.18	18.85	3355	1285	38.30
50	11	3111.93	3111.93	0.00	1	0.11	854.98	1582.83	674.12	27.47	50.86	21.66	7507	3865	51.49
50	10	3132.93	3132.93	0.00	1	0.34	758.96	1780.74	593.23	24.23	56.84	18.94	15939	7902	49.58
50	9	2613.87	2614.04	0.01	3	4.86	725.22	1597.55	291.27	27.74	61.11	11.14	53403	29608	55.44
60	10	4699.65	4699.65	0.00	1	0.10	978.67	2460.76	1260.22	20.82	52.36	26.82	7328	4026	54.94
60	13	3932.81	3932.81	0.00	1	1.84	1057.72	2037.97	837.12	26.89	51.82	21.29	46762	23491	50.24
60	12	3854.85	3855.20	0.01	3	216.90	1036.28	1994.95	823.97	26.88	51.75	21.37	392231	242493	61.82
60	12	4040.29	4040.29	0.00	1	0.27	976.01	2268.32	795.96	24.16	56.14	19.70	15507	9429	60.80
60	14	4767.66	4767.66	0.00	1	0.23	1164.32	2226.85	1376.49	24.42	46.71	28.87	10839	4570	42.16
60	13	4120.63	4120.63	0.00	1	0.56	1089.74	1889.39	1141.49	26.45	45.85	27.70	21610	12209	56.50
60	12	4136.37	4136.37	0.00	1	2.08	997.11	2028.95	1110.31	24.11	49.05	26.84	43485	25250	58.07
60	11	3417.78	3424.86	0.21	257	819.66	898.75	1884.48	641.63	26.24	55.02	18.73	176842	102202	57.79
80	17	6197.25	6197.25	0.00	1	2.50	1602.76	3275.97	1318.53	25.86	52.86	21.28	32203	16516	51.29
80	16	6102.43	6102.43	0.00	1	249.93	1453.34	3060.90	1588.18	23.82	50.16	26.03	599057	306630	51.19
80	17	5211.75	5211.75	0.00	1	57.34	1456.70	2848.45	906.60	27.95	54.65	17.40	282797	146382	51.76
80	13	5714.46	5714.46	0.00	1	23.41	1225.57	2972.51	1516.37	21.45	52.02	26.54	164779	83075	50.42
80	15	5616.62	5616.62	0.00	1	298.71	1379.13	2719.07	1518.42	24.55	48.41	27.03	601291	287605	47.83

Table C4 Results for instances in Set Large (minimizing a comprehensive objective function)

n	#R	RLB	UB	Gap	Node	Time	TC	RC	DC	TC%	RC%	DC%	Label	Dominated	Ratio (%)
10	2	4330.39	4330.39	0.00	1	0.01	942.77	2079.62	1307.99	21.77	48.02	30.20	100	16	16.00
10	1	2512.64	2512.64	0.00	1	0.01	754.99	753.44	1004.22	30.05	29.99	39.97	762	447	58.66
10	1	1938.40	1938.40	0.00	1	0.04	741.43	506.95	690.02	38.25	26.15	35.60	2869	1297	45.21
10	2	2752.11	2752.11	0.00	1	0.05	942.47	501.52	1308.12	34.25	18.22	47.53	2049	1067	52.07
10	2	2740.33	2740.33	0.00	1	0.01	707.25	1056.64	976.44	25.81	38.56	35.63	150	63	42.00
10	2	2765.42	2765.42	0.00	1	0.01	732.13	1080.90	952.39	26.47	39.09	34.44	270	104	38.52
10	2	2697.90	2697.90	0.00	1	0.01	707.25	1046.42	944.23	26.21	38.79	35.00	117	42	35.90
10	2	2407.39	2407.39	0.00	1	0.01	713.24	782.74	911.41	29.63	32.51	37.86	245	119	48.57
20	4	4681.68	4681.68	0.00	1	0.01	1468.37	1824.72	1388.59	31.36	38.98	29.66	743	325	43.74
20	4	5174.13	5174.13	0.00	1	0.27	1850.21	2060.73	1263.19	35.76	39.83	24.41	15705	10706	68.17
20	5	3342.96	3342.96	0.00	1	0.49	1807.24	934.70	601.02	54.06	27.96	17.98	24009	16214	67.53
20	3	3632.16	3632.16	0.00	1	0.28	1340.31	1019.77	1272.08	36.90	28.08	35.02	15623	10268	65.72
20	3	3685.24	3685.24	0.00	1	0.03	1485.22	1588.64	611.38	40.30	43.11	16.59	2693	1614	59.93
20	3	4475.65	4475.65	0.00	1	0.03	1746.50	1641.28	1087.87	39.02	36.67	24.31	2654	1350	50.87
20	2	4261.76	4261.76	0.00	1	0.05	1034.53	1728.97	1498.26	24.27	40.57	35.16	4287	2718	63.40
20	3	4300.98	4300.98	0.00	1	0.04	1148.91	1536.12	1615.95	26.71	35.72	37.57	4151	2557	61.60
30	5	6001.87	6001.87	0.00	1	0.08	1942.07	2677.19	1382.61	32.36	44.61	23.04	5818	4009	68.91
30	4	5819.13	5819.13	0.00	1	0.64	2121.80	2428.34	1268.99	36.46	41.73	21.81	31574	19707	62.42
30	4	4261.17	4261.17	0.00	1	10.46	1949.41	1710.74	601.02	45.75	40.15	14.10	133397	95035	71.24
30	4	4342.30	4342.30	0.00	1	256.78	1702.17	1379.33	1260.81	39.20	31.76	29.04	691121	538849	77.97
30	3	4028.89	4028.89	0.00	1	0.69	1684.43	1733.08	611.38	41.81	43.02	15.17	19805	15303	77.27
30	5	5656.45	5656.45	0.00	1	3.04	2444.64	2123.94	1087.87	43.22	37.55	19.23	54987	41292	75.09
30	3	4848.75	4849.47	0.01	3	14.24	1703.49	1670.92	1475.06	35.13	34.46	30.42	98285	80872	82.28
30	3	4847.34	4847.34	0.00	1	3.03	1711.86	1519.53	1615.95	35.32	31.35	33.34	57919	47304	81.67
40	5	7724.19	7724.19	0.00	1	0.30	2373.56	3940.00	1410.63	30.73	51.01	18.26	13790	9733	70.58
40	6	8025.10	8025.10	0.00	1	3.40	2995.89	3686.48	1342.74	37.33	45.94	16.73	110883	82042	73.99
40	5	5401.60	5401.60	0.00	1	30.69	2483.39	2306.12	612.09	45.98	42.69	11.33	289566	215913	74.56
40	5	5257.27	5270.65	0.25	9	819.88	2157.79	1818.73	1294.12	40.94	34.51	24.55	647246	528388	81.64
40	5	6862.58	6862.58	0.00	1	0.80	2458.13	3542.21	862.24	35.82	51.62	12.56	20619	14285	69.28
40	4	7170.86	7170.86	0.00	1	1.96	2468.41	3463.80	1238.65	34.42	48.30	17.27	35886	23682	65.99
40	4	6646.30	6646.30	0.00	1	7.93	2493.99	2457.76	1694.56	37.52	36.98	25.50	92530	69170	74.75
40	5	7128.43	7128.43	0.00	1	2.41	2654.54	2632.76	1841.13	37.24	36.93	25.83	50114	36831	73.49
50	7	10909.69	10909.69	0.00	1	0.17	3284.24	5281.37	2344.08	30.10	48.41	21.49	6012	4028	67.00
50	10	10625.60	10625.60	0.00	1	5.58	4333.65	4810.54	1481.41	40.78	45.27	13.94	137846	92900	67.39
50	6	7261.43	7269.20	0.11	3	34.47	3622.44	3010.02	636.74	49.83	41.41	8.76	317619	209913	66.09
50	3	7046.44	7062.16	0.22	3	2546.85	2591.22	2516.14	1954.80	36.69	35.63	27.68	3770462	3026389	80.27
50	8	8914.98	8914.98	0.00	1	1.91	3662.97	4514.63	737.38	41.09	50.64	8.27	39127	25753	65.82
50	9	8046.72	8046.72	0.00	1	2.19	3420.89	3930.84	694.99	42.51	48.85	8.64	57583	41858	72.69
50	6	7190.05	7190.05	0.00	1	54.59	3097.92	3409.55	682.58	43.09	47.42	9.49	399102	322837	80.89
50	6	7127.50	7127.50	0.00	1	15.93	3213.04	3202.74	711.72	45.08	44.93	9.99	161165	123351	76.54
60	8	12549.44	12549.44	0.00	1	2.90	3744.03	5994.75	2810.65	29.83	47.77	22.40	30011	19287	64.27
60	10	12415.15	12415.15	0.00	1	31.14	4874.39	5512.36	2028.40	39.26	44.40	16.34	283762	198255	69.87
60	7	8954.33	8954.33	0.00	1	459.33	4184.43	3983.85	786.05	46.73	44.49	8.78	1472599	1007634	68.43
60	9	12152.42	12152.42	0.00	1	21.67	4550.06	5523.43	2078.93	37.44	45.45	17.11	93662	61624	65.79
60	7	9638.92	9638.92	0.00	1	70.65	3959.07	4507.70	1172.15	41.07	46.77	12.16	340406	256164	75.25
60	7	10080.20	10080.20	0.00	1	1069.56	4031.68	4110.25	1938.27	40.00	40.78	19.23	3054270	2543752	83.29
60	6	9891.04	9905.12	0.14	5	1496.56	4086.20	3829.53	1989.39	41.25	38.66	20.08	1265810	996644	78.74
80	11	15554.95	15554.95	0.00	1	344.42	4382.74	6256.45	4915.75	28.18	40.22	31.60	399625	267580	66.96
80	9	13251.32	13251.32	0.00	1	5516.94	4398.84	5408.87	3443.62	33.20	40.82	25.99	3674353	2523726	68.68

Table C5 Results for instances in Set Short (minimizing the travel cost)

n	#R	RLB	UB	Gap	Node	Time	TC	RC	DC	TC%	RC%	DC%	TCO	Label	Dominated	Ratio (%)
10	2	223.08	223.08	0.00	1	0.21	223.08	114.40	97.35	51.30	26.31	22.39	434.83	168	34	20.24
10	2	223.08	223.08	0.00	1	0.11	223.08	114.40	97.35	51.30	26.31	22.39	434.83	168	56	33.33
10	2	219.62	219.62	0.00	1	0.02	219.62	133.85	104.26	47.98	29.24	22.78	457.73	385	120	31.17
10	2	223.79	223.79	0.00	1	0.03	223.79	115.23	96.74	51.36	26.44	22.20	435.75	615	230	37.40
10	2	194.92	219.14	11.05	3	0.04	219.14	142.03	57.70	52.32	33.91	13.78	418.86	633	248	39.18
10	2	212.19	212.19	0.00	1	0.01	212.19	148.80	204.98	37.49	26.29	36.22	565.97	348	90	25.86
10	2	207.93	207.93	0.00	1	0.01	207.93	118.83	59.25	53.87	30.78	15.35	386.01	175	52	29.71
10	1	124.81	124.81	0.00	1	0.01	124.81	73.99	240.72	28.40	16.83	54.77	439.51	411	149	36.25
10	1	115.21	119.94	3.94	3	0.12	119.94	61.78	59.92	49.64	25.57	24.80	241.64	4236	1890	44.62
20	3	470.66	470.66	0.00	1	0.01	470.66	183.74	75.72	64.46	25.17	10.37	730.12	875	273	31.20
20	4	481.06	481.06	0.00	1	0.04	481.06	228.17	60.66	62.48	29.64	7.88	769.89	3059	1141	37.30
20	4	474.72	474.72	0.00	1	0.03	474.72	249.01	125.45	55.90	29.32	14.77	849.18	2272	1079	47.49
20	3	379.85	379.85	0.00	1	0.08	379.85	198.91	66.42	58.88	30.83	10.29	645.18	6771	3587	52.98
20	4	428.90	428.90	0.00	1	0.02	428.90	267.20	119.76	52.57	32.75	14.68	815.86	2118	806	38.05
20	4	461.60	461.60	0.00	1	0.01	461.60	233.78	44.09	62.42	31.61	5.96	739.47	1349	429	31.80
20	2	254.88	254.88	0.00	1	0.05	254.88	143.74	81.75	53.06	29.92	17.02	480.38	3958	1969	49.75
20	3	298.95	352.81	15.27	7	0.29	352.81	156.28	73.92	60.51	26.81	12.68	583.02	10202	5564	54.54
30	7	782.68	816.32	4.12	7	0.04	816.32	390.42	71.48	63.86	30.54	5.59	1278.22	1250	397	31.76
30	6	729.29	735.04	0.78	3	0.09	735.04	346.18	60.66	64.37	30.32	5.31	1141.88	3988	1548	38.82
30	5	643.51	643.51	0.00	1	0.06	643.51	301.75	125.45	60.10	28.18	11.72	1070.70	4411	1894	42.94
30	4	563.08	569.44	1.12	11	1.36	569.44	288.76	67.01	61.55	31.21	7.24	925.22	24607	15484	62.93
30	5	621.40	621.40	0.00	1	0.04	621.40	335.60	119.76	57.71	31.17	11.12	1076.76	2760	1085	39.31
30	6	699.36	699.36	0.00	1	0.06	699.36	346.67	44.09	64.15	31.80	4.04	1090.12	2592	876	33.80
30	4	430.33	476.33	9.66	3	0.18	476.33	242.22	72.78	60.19	30.61	9.20	791.33	7484	3196	42.70
30	4	484.09	537.95	10.01	77	4.16	537.95	237.97	67.13	63.81	28.23	7.96	843.05	15452	8903	57.62
40	10	1299.92	1333.76	2.54	5	0.04	1333.76	592.06	84.95	66.33	29.44	4.22	2010.77	1402	429	30.60
40	7	963.75	972.17	0.87	3	0.08	972.17	433.56	74.02	65.70	29.30	5.00	1479.75	3794	1545	40.72
40	7	860.45	921.25	6.60	53	1.33	921.25	414.39	153.45	61.87	27.83	10.30	1489.09	7684	3957	51.50
40	6	749.80	778.26	3.66	43	5.08	778.26	370.67	70.20	63.84	30.40	5.76	1219.12	21778	13928	63.95
40	8	1044.43	1044.43	0.00	1	0.06	1044.43	463.02	135.83	63.56	28.18	8.27	1643.28	4937	2024	41.00
40	8	950.51	970.19	2.03	3	0.15	970.19	469.22	50.40	65.12	31.50	3.38	1489.82	5117	1962	38.34
40	6	787.20	803.12	1.98	3	0.30	803.12	385.41	84.84	63.07	30.27	6.66	1273.36	10063	5103	50.71
40	6	785.19	858.60	8.55	1021	43.62	858.60	350.45	96.83	65.75	26.84	7.41	1305.88	26648	16610	62.33
50	12	1488.96	1568.33	5.06	47	0.35	1568.33	698.28	130.85	65.42	29.13	5.46	2397.46	2096	670	31.97
50	9	1211.84	1211.84	0.00	1	0.07	1211.84	572.60	79.20	65.03	30.72	4.25	1863.64	5041	1960	38.88
50	8	1045.81	1169.24	10.56	5551	80.04	1169.24	471.41	110.81	66.76	26.92	6.33	1751.46	9321	5282	56.67
50	6	890.93	919.35	3.09	73	12.36	919.35	429.17	81.95	64.27	30.00	5.73	1430.47	32491	21690	66.76
50	10	1299.44	1299.44	0.00	1	0.04	1299.44	606.48	105.34	64.61	30.15	5.24	2011.26	3879	1625	41.89
50	9	1220.36	1220.36	0.00	1	0.10	1220.36	556.23	120.87	64.32	29.31	6.37	1897.46	6833	2497	36.54
50	8	998.60	1093.37	8.67	2285	74.81	1093.37	475.23	73.01	66.60	28.95	4.45	1641.61	8608	4743	55.10
50	7	924.90	1053.02	12.17	230903	7200.00	1053.02	437.10	147.56	64.30	26.69	9.01	1637.68	30390	19685	64.77
60	14	1723.36	1773.18	2.81	11	0.24	1773.18	837.71	315.51	60.59	28.63	10.78	2926.40	3019	856	28.35
60	12	1589.67	1589.67	0.00	1	0.13	1589.67	718.64	180.13	63.88	28.88	7.24	2488.43	8102	3255	40.18
60	9	1156.68	1280.11	9.64	12101	726.79	1280.11	575.12	166.67	63.31	28.44	8.24	2021.91	18857	11582	61.42
60	7	943.11	968.63	2.63	3	2.29	968.63	482.33	123.47	61.52	30.64	7.84	1574.43	67750	47261	69.76
60	12	1517.93	1517.93	0.00	1	0.10	1517.93	701.75	131.10	64.57	29.85	5.58	2350.77	6268	2497	39.84
60	12	1557.11	1626.15	4.25	787	27.88	1626.15	718.36	295.95	61.59	27.21	11.21	2640.45	12680	4047	31.92
60	9	1187.12	1240.63	4.31	499	54.95	1240.63	578.06	191.71	61.71	28.75	9.54	2010.40	31834	18494	58.10
60	8	1074.97	1165.25	7.75	15403	7200.00	1165.25	533.77	204.82	61.21	28.04	10.76	1903.84	86116	52315	60.75
80	19	2391.51	2391.51	0.00	1	0.32	2391.51	1192.71	353.08	60.74	30.29	8.97	3937.29	8618	2253	26.14
80	15	1954.14	1954.14	0.00	1	1.23	1954.14	949.77	244.11	62.08	30.17	7.75	3148.03	25762	9697	37.64
80	11	1465.28	1547.25	5.30	5753	6923.25	1547.25	757.75	362.61	58.00	28.41	13.59	2667.61	121459	74424	61.27
80	9	1271.09	1293.23	1.71	495	3248.11	1293.23	650.67	198.78	60.36	30.37	9.28	2142.68	393765	279872	71.08
80	14	1854.51	1854.51	0.00	1	3.21	1854.51	904.51	220.85	62.23	30.35	7.41	2979.87	45022	17229	38.27
80	13	1712.81	1781.85	3.87	253	186.00	1781.85	808.90	309.35	61.44	27.89	10.67	2900.10	54440	21258	39.05
80	11	1562.08	1597.36	2.21	1511	3064.29	1597.36	759.41	302.59	60.07	28.56	11.38	2659.36	118692	64461	54.31
80	11	1400.56	1422.44	1.54	133	1479.10	1422.44	737.56	315.99	57.45	29.79	12.76	2476.00	400424	243117	60.71

Table C6 Results for instances in Set Normal (minimizing the travel cost)

n	#R	RLB	UB	Gap	Node	Time	TC	RC	DC	TC%	RC%	DC%	TCO	Label	Dominated	Ratio (%)
10	2	108.26	108.26	0.00	1	0.01	108.26	765.60	303.19	9.20	65.04	25.76	1177.05	120	6	5.00
10	2	114.28	114.28	0.00	1	0.01	114.28	615.84	527.41	9.09	48.97	41.94	1257.53	333	70	21.02
10	2	119.43	119.43	0.00	1	0.01	119.43	581.44	194.32	13.34	64.95	21.71	895.19	243	51	20.99
10	2	115.27	115.27	0.00	1	0.01	115.27	578.64	256.59	12.13	60.88	27.00	950.50	319	71	22.26
10	2	107.65	107.65	0.00	1	0.01	107.65	594.40	508.56	8.89	49.10	42.01	1210.61	154	13	8.44
10	2	117.44	117.44	0.00	1	0.01	117.44	743.60	511.69	8.56	54.17	37.28	1372.73	140	3	2.14
10	2	114.28	114.28	0.00	1	0.01	114.28	569.44	483.69	9.79	48.78	41.43	1167.41	149	14	9.40
10	2	105.10	105.10	0.00	1	0.01	105.10	494.22	509.07	9.48	44.59	45.93	1108.39	179	35	19.55
10	2	112.52	112.52	0.00	1	0.01	112.52	572.49	398.99	10.38	52.81	36.81	1084.00	261	58	22.22
20	3	240.07	240.07	0.00	1	0.01	240.07	969.92	222.84	16.75	67.69	15.55	1432.83	601	56	9.32
20	3	279.60	294.82	5.16	3	0.11	294.82	816.53	448.09	18.91	52.36	28.73	1559.44	5351	627	11.72
20	3	277.34	277.34	0.00	1	0.12	277.34	905.77	401.20	17.51	57.17	25.32	1584.31	7739	1538	19.87
20	3	231.98	231.98	0.00	1	0.42	231.98	898.35	315.15	16.05	62.15	21.80	1445.48	30020	8726	29.07
20	3	283.92	283.92	0.00	1	0.01	283.92	931.20	183.68	20.30	66.57	13.13	1398.80	849	82	9.66
20	3	239.20	239.20	0.00	1	0.01	239.20	956.72	229.24	16.78	67.13	16.09	1425.16	554	44	7.94
20	4	332.69	332.69	0.00	1	0.01	332.69	1054.07	339.09	19.28	61.08	19.65	1725.85	914	104	11.38
20	3	229.16	229.16	0.00	1	0.03	229.16	917.60	206.30	16.94	67.82	15.25	1353.06	4658	1157	24.84
20	3	264.62	278.92	5.13	3	0.10	278.92	828.15	279.06	20.12	59.75	20.13	1386.13	8668	2783	32.11
30	5	381.06	381.06	0.00	1	0.03	381.06	1687.20	220.92	16.65	73.70	9.65	2289.18	1911	215	11.25
30	4	367.62	367.62	0.00	1	0.12	367.62	1183.33	448.09	18.39	59.19	22.42	1999.04	9655	2062	21.36
30	4	372.33	372.33	0.00	1	2.28	372.33	1214.40	402.48	18.72	61.05	20.23	1989.21	95955	48991	51.06
30	4	301.93	301.93	0.00	1	13.96	301.93	1333.90	311.63	15.50	68.49	16.00	1947.46	432967	230304	53.19
30	5	375.94	375.94	0.00	1	0.07	375.94	1430.40	183.68	18.89	71.88	9.23	1990.02	3267	524	16.04
30	5	387.16	387.16	0.00	1	0.03	387.16	1473.60	229.24	18.52	70.51	10.97	2090.00	1497	177	11.82
30	5	387.93	387.93	0.00	1	0.08	387.93	1394.07	339.09	18.29	65.72	15.99	2121.09	4660	1161	24.91
30	4	347.63	347.63	0.00	1	0.34	347.63	1262.78	217.79	19.01	69.07	11.91	1828.20	16911	4072	24.08
30	4	370.72	370.72	0.00	1	1.02	370.72	1176.95	284.02	20.24	64.25	15.51	1831.69	46039	13890	30.17
40	7	555.42	555.42	0.00	1	0.04	555.42	2192.72	270.99	18.40	72.63	8.98	3019.13	1782	159	8.92
40	6	507.83	523.05	2.91	3	0.51	523.05	1815.03	498.49	18.44	63.99	17.57	2836.57	24232	5626	23.22
40	6	500.11	500.11	0.00	1	2.39	500.11	1831.64	478.11	17.80	65.19	17.02	2809.86	97994	40891	41.73
40	4	416.71	418.44	0.41	7	25.30	418.44	1453.36	395.98	18.45	64.09	17.46	2267.78	653482	346207	52.98
40	7	642.86	642.86	0.00	1	0.09	642.86	2096.18	227.98	21.67	70.65	7.68	2967.02	4988	1163	23.32
40	8	630.78	630.78	0.00	1	0.03	630.78	2407.92	344.00	18.65	71.18	10.17	3382.70	2304	456	19.79
40	7	619.94	619.94	0.00	1	0.09	619.94	2031.41	370.15	20.52	67.23	12.25	3021.50	7707	1587	20.59
40	7	563.76	563.76	0.00	1	0.40	563.76	2112.94	244.66	19.30	72.33	8.37	2921.36	21252	5077	23.89
40	5	472.82	484.29	2.37	3	2.26	484.29	1499.66	320.44	21.02	65.08	13.91	2304.39	100633	31519	31.32
50	8	700.68	700.68	0.00	1	0.06	700.68	2514.95	504.97	18.83	67.60	13.57	3720.60	2341	289	12.35
50	8	710.70	710.70	0.00	1	0.28	710.70	2389.29	379.76	20.42	68.66	10.91	3479.75	25753	6763	26.26
50	8	649.90	669.84	2.98	171	139.39	669.84	2499.37	367.42	18.94	70.67	10.39	3536.63	137547	58888	42.81
50	6	506.55	509.29	0.54	7	108.40	509.29	1927.45	570.75	16.93	64.09	18.98	3007.49	2174096	1190342	54.75
50	9	712.98	712.98	0.00	1	0.10	712.98	2534.15	570.03	18.68	66.39	14.93	3817.16	8456	1825	21.58
50	9	714.23	714.23	0.00	1	0.04	714.23	2858.80	678.36	16.80	67.24	15.96	4251.39	2604	527	20.24
50	10	759.80	759.80	0.00	1	0.11	759.80	2525.28	630.93	19.40	64.49	16.11	3916.01	10758	2802	26.05
50	8	693.90	693.90	0.00	1	0.34	693.90	2485.72	573.39	18.49	66.23	15.28	3753.01	22677	7344	32.39
50	6	572.52	574.01	0.26	3	2.39	574.01	1852.46	399.69	20.31	65.55	14.14	2826.16	143899	46630	32.40
60	10	938.58	938.58	0.00	1	0.21	938.58	3113.51	1251.31	17.70	58.71	23.59	5303.40	9481	2182	23.01
60	10	926.91	926.91	0.00	1	2.62	926.91	2732.71	878.26	20.43	60.22	19.35	4537.88	129387	47421	36.65
60	9	881.20	902.64	2.38	233	4430.29	902.64	2777.22	892.86	19.74	60.73	19.53	4572.72	832025	410952	49.39
60	10	940.15	940.15	0.00	1	0.42	940.15	2925.38	779.50	20.24	62.98	16.78	4645.03	26216	8512	32.47
60	11	939.45	939.45	0.00	1	0.40	939.45	3414.80	1376.49	16.39	59.59	24.02	5730.74	10281	2584	25.13
60	11	957.25	957.25	0.00	1	0.75	957.25	3013.68	1119.98	18.80	59.20	22.00	5090.91	30903	9728	31.48
60	10	930.82	930.82	0.00	1	2.47	930.82	2935.81	1090.47	18.78	59.22	22.00	4957.10	76398	31234	40.88
60	7	688.66	702.92	2.03	23	36.09	702.92	2171.92	866.21	18.79	58.06	23.15	3741.05	435972	157124	36.04
80	16	1474.75	1474.75	0.00	1	7.02	1474.75	4483.29	1306.94	20.30	61.71	17.99	7264.98	48193	12993	26.96
80	13	1313.40	1313.40	0.00	1	485.75	1313.40	4059.25	1652.49	18.70	57.78	23.52	7025.14	845299	355742	42.08
80	14	1299.23	1299.23	0.00	1	22.05	1299.23	3944.73	890.83	21.18	64.30	14.52	6134.79	202724	64826	31.98
80	12	1131.37	1131.37	0.00	1	36.05	1131.37	3711.35	1516.37	17.79	58.36	23.85	6359.09	154946	51464	33.21
80	13	1229.24	1229.24	0.00	1	206.99	1229.24	3584.02	1487.80	19.51	56.88	23.61	6301.06	618815	218947	35.38

Table C7 Results for instances in Set Large (minimizing the travel cost)

n	#R	RLB	UB	Gap	Node	Time	TC	RC	DC	TC%	RC%	DC%	TCO	Label	Dominated	Ratio (%)
10	2	942.77	942.77	0.00	1	0.01	942.77	2252.80	1307.99	20.93	50.02	29.04	4503.57	101	14	13.86
10	1	733.74	733.74	0.00	1	0.01	733.74	965.60	1004.22	27.14	35.72	37.14	2703.56	1227	674	54.93
10	1	733.74	733.74	0.00	1	0.07	733.74	676.71	711.47	34.58	31.89	33.53	2121.92	4926	2259	45.86
10	1	895.98	895.98	0.00	1	0.02	895.98	1160.00	1214.92	27.39	35.46	37.14	3270.90	2988	1518	50.80
10	2	692.82	692.82	0.00	1	0.01	692.82	1638.00	972.80	20.97	49.58	29.45	3303.62	276	100	36.23
10	2	732.13	732.13	0.00	1	0.01	732.13	1492.40	952.39	23.05	46.98	29.98	3176.92	272	96	35.29
10	2	695.14	695.14	0.00	1	0.01	695.14	1590.40	940.38	21.55	49.30	29.15	3225.91	225	68	30.22
10	1	558.94	558.94	0.00	1	0.01	558.94	1050.80	894.45	22.32	41.96	35.72	2504.19	366	163	44.54
20	3	1373.16	1373.16	0.00	1	0.01	1373.16	2962.57	1388.59	23.99	51.75	24.26	5724.32	725	228	31.45
20	3	1800.28	1800.28	0.00	1	0.45	1800.28	3271.67	1298.33	28.26	51.36	20.38	6370.28	21401	14921	69.72
20	2	1579.85	1579.85	0.00	1	0.98	1579.85	2279.29	780.59	34.05	49.13	16.82	4639.72	49081	32102	65.41
20	2	1203.25	1203.25	0.00	1	0.35	1203.25	2435.20	1244.48	24.64	49.87	25.49	4882.92	23568	15834	67.18
20	3	1162.61	1162.61	0.00	1	0.03	1162.61	3198.97	597.94	23.44	64.50	12.06	4959.53	1995	1008	50.53
20	3	1404.02	1404.02	0.00	1	0.02	1404.02	2906.50	1131.91	25.80	53.40	20.80	5442.43	2298	1265	55.05
20	2	996.27	996.27	0.00	1	0.06	996.27	2177.29	1545.28	21.11	46.14	32.75	4718.83	5666	3494	61.67
20	2	996.27	996.27	0.00	1	0.04	996.27	2165.37	1592.06	20.96	45.55	33.49	4753.71	3758	2055	54.68
30	4	1869.24	1869.24	0.00	1	0.54	1869.24	4033.69	1382.61	25.66	55.37	18.98	7285.54	10522	5354	50.88
30	4	2110.25	2110.25	0.00	1	0.81	2110.25	4137.44	1207.55	28.31	55.50	16.20	7455.25	26013	15308	58.85
30	2	1804.21	1804.21	0.00	1	24.38	1804.21	2279.29	780.59	37.09	46.86	16.05	4864.08	380115	260238	68.46
30	3	1565.11	1565.11	0.00	1	37.62	1565.11	3501.41	1234.88	24.84	55.57	19.60	6301.40	467150	366047	78.36
30	3	1320.14	1320.14	0.00	1	1.36	1320.14	3330.70	597.94	25.15	63.46	11.39	5248.78	27133	17939	66.12
30	5	2081.28	2081.28	0.00	1	11.29	2081.28	4853.70	1153.80	25.73	60.01	14.26	8088.78	79614	48697	61.17
30	2	1168.62	1168.62	0.00	1	24.44	1168.62	2309.04	1800.23	22.14	43.75	34.11	5277.89	342481	282326	82.44
30	3	1341.45	1341.45	0.00	1	5.39	1341.45	3256.16	1592.06	21.67	52.61	25.72	6189.68	90249	69351	76.84
40	5	2181.84	2181.84	0.00	1	0.81	2181.84	5075.67	1401.03	25.20	58.62	16.18	8658.55	17893	8803	49.20
40	5	2804.44	2804.44	0.00	1	2.66	2804.44	5311.53	1694.19	28.59	54.14	17.27	9810.16	86424	58791	68.03
40	3	2269.63	2269.63	0.00	1	49.11	2269.63	3047.67	791.66	37.15	49.89	12.96	6108.97	670135	480920	71.76
40	4	1988.98	1994.39	0.27	15	178.23	1994.39	4548.30	1256.91	25.57	58.31	16.12	7799.60	668534	538519	80.55
40	5	2162.21	2162.21	0.00	1	0.63	2162.21	5171.29	848.80	26.43	63.20	10.37	8182.30	17571	9293	52.89
40	5	2372.94	2372.94	0.00	1	3.36	2372.94	5017.70	1253.81	27.45	58.05	14.50	8644.45	40967	21141	51.60
40	4	2108.31	2108.31	0.00	1	6.02	2108.31	4044.89	1920.25	26.11	50.10	23.78	8073.45	93417	65467	70.08
40	5	2234.93	2234.93	0.00	1	4.36	2234.93	4603.07	1864.85	25.68	52.89	21.43	8702.86	59956	35155	58.63
50	6	3214.41	3214.41	0.00	1	0.57	3214.41	6834.47	2321.82	25.98	55.25	18.77	12370.71	17869	9133	51.11
50	7	3812.25	3812.25	0.00	1	1.36	3812.25	7311.20	1799.26	29.50	56.58	13.92	12922.71	54171	38076	70.29
50	5	3198.80	3198.80	0.00	1	22.33	3198.80	5241.16	628.38	35.27	57.80	6.93	9068.34	472876	324711	68.67
50	3	2451.25	2456.46	0.21	7	5103.87	2456.46	3503.35	2081.69	30.55	43.57	25.89	8041.51	4398458	3623042	82.37
50	7	3144.48	3144.48	0.00	1	5.84	3144.48	6898.50	767.28	29.09	63.81	7.10	10810.26	38051	16723	43.95
50	6	2872.56	2872.56	0.00	1	7.65	2872.56	6183.70	678.59	29.51	63.52	6.97	9734.85	80416	46248	57.51
50	5	2844.75	2844.75	0.00	1	165.52	2844.75	5021.43	649.31	33.41	58.97	7.63	8515.50	590133	438568	74.32
50	5	2585.97	2585.97	0.00	1	37.45	2585.97	5098.08	671.27	30.95	61.02	8.03	8355.33	167015	99907	59.82
60	8	3531.96	3531.96	0.00	1	9.11	3531.96	8410.07	2785.68	23.98	57.10	18.91	14727.71	60707	24533	40.41
60	8	4259.07	4259.07	0.00	1	14.69	4259.07	8209.37	2321.89	28.80	55.50	15.70	14790.33	176087	121917	69.24
60	6	3670.36	3670.36	0.00	1	222.77	3670.36	6028.87	1487.60	32.81	53.89	13.30	11186.83	1424430	1045497	73.40
60	9	3837.23	3837.23	0.00	1	143.59	3837.23	9009.23	2104.66	25.67	60.26	14.08	14951.11	164368	63836	38.84
60	7	3215.53	3215.53	0.00	1	357.23	3215.53	7310.03	1081.30	27.70	62.98	9.32	11606.85	408336	208173	50.98
60	7	3450.04	3450.04	0.00	1	6941.37	3450.04	7000.56	1899.15	27.94	56.69	15.38	12349.74	2781462	2070234	74.43
60	7	3191.26	3191.26	0.00	1	1370.74	3191.26	7080.80	1921.33	26.17	58.07	15.76	12193.40	868153	529935	61.04
80	9	4139.93	4139.93	0.00	1	2180.06	4139.93	8747.20	4915.75	23.25	49.13	27.61	17802.88	830783	361049	43.46