

Article

Dual-Sourcing Inventory Routing Problem with Route-Dependent Lead Times in Rolling Horizon Framework

Weibo Zheng ^{1,2,*}  and Hong Zhou ^{1,2} ¹ School of Economics and Management, Beihang University, 37 Xueyuan Road, Beijing 100191, China² Beijing Key Laboratory of Emergency Support Simulation Technologies for City Operations, 37 Xueyuan Road, Beijing 100191, China

* Correspondence: zhengweibo@buaa.edu.cn; Tel.: +86-185-6171-8168

Featured Application: This work can be applied to perishable goods supply chains and emergency logistics.

Abstract: Traditional inventory routing problems ignore the time consumption in transportation. In this paper, an inventory routing problem with air–land transportation and route-dependent lead times is studied. The model is based on the rolling horizon framework which can serve as a “here-and-now” approximation for multi-period inventory routing problems. A planning horizon crossing strategy is proposed to consider the effect of the single-period decision on long-term planning. The box uncertainty set is used to depict demands. A tractable closed-form robust solution for optimal replenishment quantity is derived. An adaptive variable neighborhood search algorithm is developed for this problem. A novel shaking phase is proposed, and the performance of shaking operators is evaluated in numerical experiments. Results also validate the effectiveness of the robust solution and the planning horizon crossing strategy.

Keywords: dual-sourcing; inventory routing problem; rolling horizon; robust optimization; adaptive variable neighborhood search

**Citation:** Zheng, W.; Zhou, H.Dual-Sourcing Inventory Routing Problem with Route-Dependent Lead Times in Rolling Horizon Framework. *Appl. Sci.* **2023**, *13*, 2229. <https://doi.org/10.3390/app13042229>

Academic Editor: Xianpeng Wang

Received: 16 January 2023

Revised: 7 February 2023

Accepted: 8 February 2023

Published: 9 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Integrating the two areas in logistic planning of transportation and inventory can reduce costs system-wisely, such as vendor-managed inventory (VIM), has gained acceptance in many supply chain environments. Comi examines the activities and decision makers involved in urban goods movements, showing the need to integrating optimize purchasing, restocking, and delivery [1]. This paper addresses the combined problem which introducing air–land transportation and the dual-sourcing inventory policy into the inventory routing problem (IRP). The central depot serves several demand locations by land vehicles (e.g., trucks) and aerial crafts (e.g., drones or helicopters). The central depot decides which demand locations are to be replenished by land and aerial modes and in what order. We consider the transportation time in the replenishment as the lead time for each demand location. This feature makes decisions on the replenishment quantity and routing depend more tightly on each other. It is very useful in application scenarios that require high timeliness, such as emergency logistics and perishable goods supply chains.

This problem is an extension of the inventory routing problem, which is first studied by Bell et al. [2]. The central decision-maker determines an inventory policy and a set of vehicle routes to minimize the total cost. All vehicles must leave the depot at the beginning of each inventory planning period and return at the end, while also staying within capacity constraints and meeting the demands of locations. There are many situations where the routing and the replenishment quantity decisions should be determined simultaneously, such as the retail supply chain, for companies such as Wal-Mart and Proctor&Gamble; the petrochemical supply

chain; and emergency logistics [3–5]. Inventory management and vehicle routing decisions are interrelated. The inventory policy determines which demand locations must be served and the replenishment quantity. It affects the inventory cost. On the other hand, the delivery cost depends on the vehicle routes, which, in turn, need information about the demand location selection and the replenishment quantity. This inter-relationship recently motivate some researchers to model these two decisions simultaneously. Moin and Salhi present a review of IRPs from aspects of single-period models, multi-period models, infinite horizon models, and stochastic models [6]. Coelho et al. summarize the 30 years develop of IRP studies from 1983 to 2013 comprehensively [7]. Recently, Shaabani presents a review paper focusing on perishable inventory routing problems [8].

Most existing IRP models assume the replenishment lead time is determined exogenously. As a consequence, their application to physical distribution problems is confined to situations where the timeliness does not significantly influence the decision rather than in an emergency or perishable goods supply chain. Most IRP studies for perishable goods focus on product perishability, such as [9–13]. Li et al. first study the timeliness for perishable IRP by considering the transportation time in replenishment lead time [14]. They propose a myopic policy that in each stage the replenishment and routing decision is only made once, then repeat the stage to minimize the cost. Dror and Ball first study the effect of the short-term on long-term planning [15]. They consider single-period models as sub-problems and use penalty and incentive factors to affect the later period's decisions. Then, Coelho and Laporte discuss several exact algorithms for multi-period integrated optimization models [16]. The multi-period models are useful in that they offer a more realistic strategy. However, these approaches produce a high-quality solution that requires significantly a larger computing effort.

Decomposing a multi-period problem into a series of single-period problems is often achieved. Federgruen and Zipkin first use a single-period model to solve IRP by rolling horizon [17]. They propose a myopic solution of multi-period IRP and describe the scenario for each planning stage: first, the depot's initial inventory level of each demand location is revealed at the beginning of the stage. Then the decision-maker determines the inventory and routing decision. After replenishment is made, the demand occurs. The inventory retention and shortage costs are calculated at each demand location based on demand information. Finally, determine the end-of-the-day inventory levels, and repeat these procedures. The rolling horizon framework solves a short-term problem approximated as a periodic solution over a long-term horizon. Such efforts include Jaillet et al., Sakiani et al., and SteadieSeifi et al. [18–20].

The stochastic demand IRP is a difficult challenge. Archetti et al. consider a finite planning horizon IRP with stochastic demand [21]. Hvattum et al. propose a cyclic distribution strategy [22]. They point out that stochastic IRP can be solved by solving a scenario tree. Huang and Lin study a multi-product IRP with uncertain demand and stockout [23]. They assume that the demand is only revealed when the vehicle arrives at the demand location and included a recourse mechanism consisting of a return trip to the depot when stockouts occur. Alvarez et al. propose a two-stage decision framework to deal with a scenario-based stochastic IRP [24]. They determine the routing decisions in the first stage while the inventory policies are determined in the second stage. These studies are based on stochastic optimization, whereby it assumes that the probability distribution is known for demand. However, obtaining the probability distribution of demand in practice is extremely complex, even impossible sometimes.

Robust optimization is a powerful methodology for uncertain problems with incomplete information on their probability distributions. Robust optimization is to optimize the objective when considering the worst case of uncertain parameters under the defined uncertainty set. Soyster and Allen first propose robust optimization for solving linear programming problems with uncertain parameters, considering their worst-case values in the uncertainty set [25]. Mamani et al. derive a closed-form robust solution for an uncertain inventory problem with stochastic demand [26]. Their work is based on the dual balancing

policies proposed by Levi et al. [27,28]. Then, Sun and Van Mieghem extend their work to dual-sourcing inventory problems [29]. They provide a closed-form robust solution by normalizing the slow ordering cost and fast lead time to 0. Dual-sourcing inventory policy is using two replenishment sources, where the regular one is cheaper but slower than the other one. By interspersing one or several faster and more expensive replenishment periods in a regular one, the dual-sourcing policy can reduce retention costs and the impact of the demand uncertainty [30].

Traffic congestion will lead to a longer travel time, thus affecting the timeliness of replenishment. Musolino et al. study the a vehicle routing problem based on reliable link travel times [31]. The network fundamental diagram is very useful for dynamic traffic assignment, which can reflect the impact of traffic conditions at different times on routing choices. Then, Comi and Polimeni use the floating car data for choosing transportation routes [32]. Croce et al. use floating and probe vehicle data to estimate energy consumption of electric vehicles [33]. In addition, Russo et al. study regional transport plans, which can help the traffic situation from the perspective of upper level planning [34]. To ensure the timeliness of replenishment, it is an essential strategy to combine air and ground transportation, especially in the perishable goods supply chain and emergency supplies distribution [35]. Sacramento et al. propose an air–land transportation vehicle routing problem, which uses drones for last-mile delivery [36]. Their study shows that air–land transportation is more efficient in terms of time than only using land vehicles. Zheng and Zhou study an air–land transportation problem for disaster relief, taking into account road damage [37]. They use a set of congestion coefficients to adjust the adjacency matrix of land transportation to reflect the impact of road congestion on transportation time.

The inventory routing problem is NP-hard because it subsumes the classical VRP [7]. Most of the papers propose heuristics for its solution, such as genetic algorithm [38,39], large neighborhood search approach [40], variable neighborhood search approach [41], etc. However, a number of exact algorithms are also available, such as general-purpose solver [42], Lagrangian relaxation approach [43], branch-and-cut algorithm [16,44], branch-and-pricing algorithm [45], etc.

In this paper, we combine the dual-sourcing inventory policy with the air–land transportation routing problem, where the demand is uncertain and replenishment lead time depends on the routing decision. By doing so, we are filling in a significant gap in the literature, which is also relevant to practice, especially in perishable supply chains and emergency logistics. The purpose is to improve the timeliness of replenishment and reduce the cost of the system as a whole. To our knowledge, ours is the first attempt to integrate the dual-sourcing inventory policy and the air–land transportation routing problem in a single model. For the seek of route-dependent lead times IRP, the complexity of multi-period models is very high. So we develop a single-period model based on the rolling horizon framework, our work can serve as a “here-and-now” approximation. At the end of each planning cycle, the initial inventory level of the next planning cycle is revealed. Then we roll the planning horizon to optimize the next planning cycle. The box uncertainty set is used to depict the demand. A planning horizon crossing strategy is proposed to consider the effect of the single-period decision on long-term planning. We develop a non-linear programming model of the problem. While it is quite complex, the model can be decomposed into a routing master problem and several inventory sub-problems. We can solve this problem using a two-stage algorithm. To illustrate this, we now describe the methodology in greater detail: since the uncertainty demands are only contained in inventory sub-problems, we derive a closed-form robust solution of the sub-problem to determine replenishment quantity. An adaptive variable neighborhood search (AVNS) algorithm is developed to solve the routing master problem. In the first stage, obtain a set of replenishment routes by AVNS. In the second stage, using the closed-form solution of sub-problems, the optimal replenishment quantities of each demand location are revealed. Then return to the first stage, and recheck the capacity of vehicles. A penalty occurs if

any vehicle exceeds its capacity. These two stages are iterated repeatedly, and the whole problem can be solved under the framework of the AVNS algorithm.

To summarize the remainder of the paper, a description of the problem and a single-period model are presented in Section 2. Then, in Section 3, the decomposition methodology and an adaptive variable neighborhood search algorithm are discussed. Section 4 presents numerical results. Conclusions and future work are discussed in Section 5. An Appendix A presents proofs of lemma and propositions.

2. Problem Description and Formulations

The problem addressed in this paper is a dual-sourcing inventory routing problem with route-dependent lead times. The road network congestion can be reflected by the adjacency matrix of land transportation. We propose a single-period model and roll the planning horizon to solve the multi-period problem. Considering the transportation time as the replenishment lead time makes the links between replenishment quantity and delivery routes much tighter and more complex. The replenishment sequence of demand locations will impact the arrival time, which affects the replenishment quantity. Then, in turn, affects the route decision. To minimize the delivery and inventory cost, routes and replenishment quantity should be optimized simultaneously. Furthermore, the inventory cost before the first replenishment arrival in each planning cycle depends on the replenishment quantity of the last planning cycle. So we use a planning horizon crossing strategy to affect long-term planning. The interaction between the routing decision and the inventory policy is shown in Figure 1.

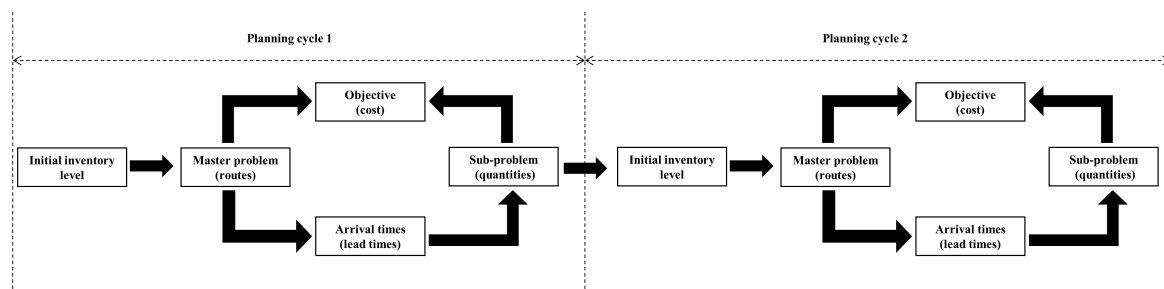


Figure 1. A brief sketch for the problem (two planning cycles).

2.1. Problem Description

In this problem, we define a directed graph $G = (V, A)$, where $V = \{0, \dots, n+1\}$ is a vertex set, and $A = \{(i, j) \in V, i \neq j\}$ is an arc set. The vertex 0 and $n+1$ represent the same central depot. For each route, the starting vertex is 0 and the ending vertex is $n+1$. The vertex set $V' = \{1, \dots, n\}$ represents demand locations. For convenience, we further define $V_0 = V' \cup 0$ and $V_{n+1} = V' \cup n+1$. The arc (i, j) represents the shortest route between vertices i and j .

The planning horizon in each cycle is T . At the beginning of the planning cycle, check the initial inventory level I_{i0} . Then, a set of land vehicles and a set of aerial crafts depart from the central depot. They must return to the central depot before the end of the planning cycle. The unit transportation cost of a land vehicle is c^l , and the unit transportation cost of an aerial craft is c^a , where $c^a > c^l$. Similarly, for each kind of vehicle, there are a capacity Q^l, Q^a and a fixed cost f^l, f^a .

We assume that the replenishment lead time of each demand location under each replenishment mode only considers the time consumption of transportation. We use l^a and l^l to represent the aerial and land replenishment lead time of each demand location, respectively. q^a and q^l represent the aerial and land replenishment quantity. Then we further define the lead time and replenishment quantity as 0 if the replenishment mode does not visit the demand location. For a demand location, the land replenishment mode may arrive earlier than the aerial mode due to the delivery order. Therefore, let $l^f = \min\{l^a, l^l\}$ and $l^s = \max\{l^a, l^l\}$ be defined as the lead time of the faster arrival replenishment method and

the slower one when the demand location is replenished by both two modes. Similarly, q^f and q^s are faster and slower replenishment quantities, respectively. For a demand location, the inventory cost before the first replenishment arrival is not dependent on the replenishment quantity in the planning cycle, only depends on the initial inventory level, as shown in Figure 1. The initial inventory level is determined by the replenishment quantities of the last planning cycle. So, the objective should minimize the routing cost and the inventory cost from the beginning of this planning cycle to the fast replenishment arrival time in the next planning cycle. However, it is impossible to know the lead time information of the next planning cycle when planning the current cycle. Thus, we use the average fast lead time l_{p-1}^{avg} in the previous planning cycle to estimate it. This strategy is the planning horizon crossing.

Due to incomplete information, the distribution of demand is unknown. We assume demands $\tilde{d}_{it}$ are independent, and their values are in the positive symmetric interval $[\bar{d}_{it} - \hat{d}_{it}, \bar{d}_{it} + \hat{d}_{it}]$, where $\bar{d}_{it}$ is the estimated value of the demand and $\hat{d}_{it}$ the maximum deviation of the demand. At the time point t , shelters deduct the demand between $t - 1$ to t from the inventory held. If the available inventory does not meet the demand, it will have a backlog of demand to be met in the future, and a shortage cost will occur; otherwise, the on-holding inventory will be stored in the demand location to meet future demand, and lead to a holding cost. Each demand location has a unit holding cost h and a unit shortage cost s , where $s > h$.

2.2. Notations

Notations used in the model are listed as follows:

Sets

V : Vertex set.

V' : Demand location set.

V_0 : $V' \cup 0$.

V_{n+1} : $V' \cup n + 1$.

K : Land vehicle set.

A : Aerial vehicle set.

Parameters

t_{ij}^l : Time cost of transportation from i to j by land vehicles.

t_{ij}^a : Time cost of transportation from i to j by aerial vehicles.

c^l : The unit cost of land transportation mode.

c^a : The unit cost of aerial transportation mode.

f^l : Fixed cost of land transportation mode.

f^a : Fixed cost of aerial transportation mode.

Q^l : Capacity of a land vehicle.

Q^a : Capacity of an aerial vehicle.

I_{i0} : Initial inventory level of the demand location i at the beginning of the planning cycle.

l_{p-1}^{avg} : Average lead time for the first replenishment in the previous planning cycle.

T : Planning horizon of the planning cycle.

H : Inventory planning horizon, where $H = T + l_{p-1}^{avg}$.

Variables

l_{ik} : Arrival time of the demand location i replenished by vehicle k .

l_{ia} : Arrival time of the demand location i replenished by aerial vehicle a .

I_{it} : Inventory level of the demand location i at time t .

q_i^l : Replenishment quantity of the demand location i replenished by land vehicles.

q_i^a : Replenishment quantity of the demand location i replenished by aerial vehicles.

x_{ijk} : 1 if vehicle k of land transportation mode travels directly from vertex i to j ; 0 otherwise.

x_{ija} : 1 if vehicle a of aerial transportation mode travels directly from vertex i to j ; 0 otherwise.

y_{ik} : 1 if demand location i is assigned to route k ; 0 otherwise.

y_{ia} : 1 if demand location i is assigned to route a ; 0 otherwise.

2.3. Model

The objective function is the total cost in the planning cycle, including the transportation cost, and the inventory cost.

$$\min : f^l \sum_{k \in K} \sum_{j \in V'} x_{0jk} + f^a \sum_{a \in A} \sum_{j \in V'} x_{0ja} + c^l \sum_{k \in K} l_{n+1,k} + c^a \sum_{a \in A} l_{n+1,a} + \sum_{i \in V'} \sum_{t=1}^H \max\{hI_{it}, -sI_{it}\}. \quad (1)$$

s.t.:

$$I_{it} = I_{i0} + q_i^l \operatorname{sgn} \left\{ t - \sum_{k \in K} y_{ik} l_{ik} \right\} + q_i^a \operatorname{sgn} \left\{ t - \sum_{a \in A} y_{ia} l_{ia} \right\} - \sum_{\delta=1}^t \tilde{d}_{i\delta} \quad \forall i \in V', t \in \{1, \dots, H\}. \quad (2)$$

$$q_i^l \leq Q^l \sum_{k \in K} y_{ik} \quad \forall i \in V'. \quad (3)$$

$$q_i^a \leq Q^a \sum_{a \in A} y_{ia} \quad \forall i \in V'. \quad (4)$$

$$l_{ik} + t_{ij}^l - (1 - x_{ijk})T \leq l_{jk} \quad \forall k \in K, i \in V_0, j \in V_{n+1}, i \neq j. \quad (5)$$

$$l_{ia} + t_{ij}^a - (1 - x_{ija})T \leq l_{ja} \quad \forall a \in A, i \in V_0, j \in V_{n+1}, i \neq j. \quad (6)$$

$$\sum_{k \in K} y_{ik} \leq 1 \quad \forall i \in V'. \quad (7)$$

$$\sum_{a \in A} y_{ia} \leq 1 \quad \forall i \in V'. \quad (8)$$

$$\sum_{j \in V_{n+1}} x_{ijk} = y_{ik} \quad \forall i \in V', k \in K. \quad (9)$$

$$\sum_{j \in V_{n+1}} x_{ija} = y_{ia} \quad \forall i \in V', a \in A. \quad (10)$$

$$\sum_{i \in V_0} x_{ijk} = \sum_{i \in V_{n+1}} x_{jik} \quad \forall j \in V', k \in K. \quad (11)$$

$$\sum_{i \in V_0} x_{ija} = \sum_{i \in V_{n+1}} x_{jia} \quad \forall j \in V', a \in A. \quad (12)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in V_0, j \in V_{n+1}, k \in K. \quad (13)$$

$$x_{ija} \in \{0, 1\} \quad \forall i \in V_0, j \in V_{n+1}, a \in A. \quad (14)$$

$$y_{ik} \in \{0, 1\} \quad \forall i \in V', k \in K. \quad (15)$$

$$y_{ia} \in \{0, 1\} \quad \forall i \in V', a \in A. \quad (16)$$

$$l_{ik} \in \{0, \dots, T\} \quad \forall i \in V, k \in K. \quad (17)$$

$$l_{ia} \in \{0, \dots, T\} \quad \forall i \in V, a \in A. \quad (18)$$

$$q_i^l \geq 0 \quad \forall i \in V. \quad (19)$$

$$q_i^a \geq 0 \quad \forall i \in V. \quad (20)$$

Constraint (2) is an inventory balance equation, where $\sum_{\delta=1}^t \tilde{d}_{i\delta}$ is the accumulated demand. $\text{sgn}\{\cdot\}$ is an indicator function, $\text{sgn}\{A\} = 1$ if $A > 0$, and $\text{sgn}\{A\} = 0$; constraints (3) and (4) guarantee the replenishment quantity is zero if no replenishment in the planning cycle; constraints (5) and (6) are used to determine the arrival time which equals to the replenishment lead time of demand locations, and it also eliminates subtours; constraints (7) and (8) guarantee that each demand location can only be replenished no more than once by each transportation mode; constraints (9) and (10) are assignment constraints; equations (11) and (12) are flow conservation constraints; and (13)–(20) are non-negativity and integrity of variables constraints.

3. Methodology

Under the rolling horizon framework, multi-period problems can be solved by following procedures. Firstly, set $H = T$ and solve the single-period model in the first planning cycle. Secondly, after the end of the planning cycle, initial inventory levels I_{i0} of the next planning cycle and the average lead time l_{p-1}^{avg} of the current planning cycle are revealed. Thirdly, update the inventory planning horizon H and roll the planning horizon to repeat the procedures until the last planning cycle. Note that the inventory planning horizon H will reset to T at the last planning cycle.

For the model proposed in Section 2.3, constraint (2) is non-linear and couples variables of replenishment quantities and lead times, which is hard to be solved. We decouple it into a master problem and a set of sub-problems. The sub-problem is an inventory problem. For any demand location, the sub-problem can be formulated as follows:

$$\begin{aligned} \min : & \sum_{t=1}^H \max\{hI_t, -sI_t\} \\ \text{s.t. : } & I_t = I_0 + q^l \text{sgn}\left\{t - \sum_{k \in K} y_k l_k\right\} + q^a \text{sgn}\left\{t - \sum_{a \in A} y_a l_a\right\} - \sum_{\delta=1}^t \tilde{d}_{\delta} \quad \forall t \in \{1, \dots, H\}. \\ & q^l \geq 0 \\ & q^a \geq 0 \end{aligned} \quad (21)$$

For the sub-problem, y and l are exogenous variables, which is determined by the routing decision. We derive a closed-form solution in Section 3.2. Thus, the problem can be solved by following procedures. Firstly, given a set of replenishment routes, the lead time of each demand location is determined. Then the optimal replenishment quantities can be solved by sub-problems. Finally, check the capacity constraints and repeat these procedures.

3.1. Robust Counterpart of Sub-Problem

For any demand locations, we assume the uncertainty demand between $t - 1$ and t with the following non-negative bounded box uncertainty set:

$$U_t = \left\{d_t \mid \bar{d}_t - \hat{d}_t \leq d_t \leq \bar{d}_t + \hat{d}_t, t = 1, 2, \dots, H\right\} \quad (22)$$

Let z_t denote the inventory cost between $t - 1$ and t , the robust counterpart of the inventory model can be formulated as the following three situations:

1. When a demand location is not replenished in the planning cycle, the inventory cost is

$$\begin{aligned}
 \min : & \sum_{t \in H} z_t \\
 \text{s.t. : } & z_t \geq h(I_0 - \sum_{\delta=1}^t (\bar{d}_t - \hat{d}_t)) \quad \forall t \in \{1, \dots, H\} \\
 & z_t \geq -s(I_0 - \sum_{\delta=1}^t (\bar{d}_t + \hat{d}_t)) \quad \forall t \in \{1, \dots, H\}
 \end{aligned} \tag{23}$$

2. When a demand location is replenished by only one mode in the planning cycle, the inventory cost is

$$\begin{aligned}
 \min : & \sum_{t \in H} z_t \\
 \text{s.t. : } & z_t \geq h(I_0 - \sum_{\delta=1}^t (\bar{d}_t - \hat{d}_t)) \quad \forall t \in \{1, \dots, l\} \\
 & z_t \geq -s(I_0 - \sum_{\delta=1}^t (\bar{d}_t + \hat{d}_t)) \quad \forall t \in \{1, \dots, l\} \\
 & z_t \geq h(I_0 + q - \sum_{\delta=1}^t (\bar{d}_t - \hat{d}_t)) \quad \forall t \in \{l+1, \dots, H\} \\
 & z_t \geq -s(I_0 + q - \sum_{\delta=1}^t (\bar{d}_t + \hat{d}_t)) \quad \forall t \in \{l+1, \dots, H\} \\
 & q \geq 0
 \end{aligned} \tag{24}$$

where l is the lead time, and q is the replenishment quantity.

3. When a demand location is replenished by both two modes in the planning cycle, the inventory cost is

$$\begin{aligned}
 \min : & \sum_{t \in H} z_t \\
 \text{s.t. : } & z_t \geq h(I_0 - \sum_{\delta=1}^t (\bar{d}_t - \hat{d}_t)) \quad \forall t \in \{1, \dots, l^f\} \\
 & z_t \geq -s(I_0 - \sum_{\delta=1}^t (\bar{d}_t + \hat{d}_t)) \quad \forall t \in \{1, \dots, l^f\} \\
 & z_t \geq h(I_0 + q^f - \sum_{\delta=1}^t (\bar{d}_t - \hat{d}_t)) \quad \forall t \in \{l^f + 1, \dots, l^s\} \\
 & z_t \geq -s(I_0 + q^f - \sum_{\delta=1}^t (\bar{d}_t + \hat{d}_t)) \quad \forall t \in \{l^f + 1, \dots, l^s\} \\
 & z_t \geq h(I_0 + q^f + q^s - \sum_{\delta=1}^t (\bar{d}_t - \hat{d}_t)) \quad \forall t \in \{l^s + 1, \dots, H\} \\
 & z_t \geq -s(I_0 + q^f + q^s - \sum_{\delta=1}^t (\bar{d}_t + \hat{d}_t)) \quad \forall t \in \{l^s + 1, \dots, H\} \\
 & q^f \geq 0 \\
 & q^s \geq 0
 \end{aligned} \tag{25}$$

where l^f and l^s are the faster and the slower replenishment lead time, respectively. Similarly, q^f and q^s are the faster and the slower replenishment quantity.

3.2. Closed-Form Solution for Replenishment Quantity

Given a set of replenishment routes, the lead time of each demand location is an exogenous variable for a sub-problem. In Section 3.1, a linear programming model for sub-problems is proposed. In this section, we further derive a closed-form solution for replenishment quantity. When $\bar{d} = \bar{d}_t$ and $\hat{d} = \hat{d}_t$ for any $t = \{1, \dots, H\}$, the following proposition derives an approximate optimal solution under the continuous time condition.

Proposition 1. For any demand under a box uncertainty set U , let $\lambda = (s + h)\bar{d}_i + (s - h)\hat{d}_i$ the robust optimal replenishment quantity satisfy:

1. For any demand location i replenished by a single replenishment mode, the optimal replenishment quantity is

$$q_i = \begin{cases} \frac{(sH + hl)}{(s + h)^2} \lambda - I_{i0} & I_{i0} < \frac{\lambda}{s + h} H \\ 0 & I_{i0} \geq \frac{\lambda}{s + h} H \end{cases} \quad (26)$$

2. For any demand location i replenished by two replenishment modes, the faster and slower optimal replenishment quantities are

$$q_i^f = \begin{cases} \frac{(sl^s + hl^f)}{(s + h)^2} \lambda - I_{i0} & I_{i0} < \frac{\lambda}{s + h} l^s \\ 0 & I_{i0} \geq \frac{\lambda}{s + h} l^s \end{cases} \quad (27)$$

and

$$q_i^s = \begin{cases} \frac{(sH + hl^s)}{(s + h)^2} \lambda - I_{i0} - q_i^f & I_{i0} + q_i^f < \frac{\lambda}{s + h} H \\ 0 & I_{i0} + q_i^f \geq \frac{\lambda}{s + h} H \end{cases} \quad (28)$$

Appendix A is the proof of Proposition 1.

3.3. Adaptive Variable Neighborhood Search

The variable neighborhood search (VNS) algorithm uses a set of fixed sequence neighborhoods for search. However, this problem contains two replenishment modes, and changing the routes of one replenishment mode will affect the route selection of another mode. A fixed search sequence may produce a fixed search structure, resulting in insufficient diversity of solutions. In this section, an adaptive variable neighborhood search algorithm is proposed. Two customer pools for each mode to determine which demand location is not served by the replenishment mode in the planning cycle. The “inter-route” and “pool” operators are used in the shaking phase, and the “intra-route” operators are used in the variable neighborhood descent (VND) phase. An adaptive mechanism is developed to select the operator from the shaking neighborhoods set in the first phase. The AVNS heuristic is described as follows:

1. Algorithm framework

The algorithm begins with an initial solution S_{init} which is generated randomly. At each iteration, a neighborhood solution S' is generated from the current solution S_{cur} by using a selected shaking operator. Five heuristics are employed in this phase: removal, insertion, exchange, swap, and relocation operators. Then a neighborhood solution S'' is generated from S' by using the VND strategy. Three heuristics are employed in this phase: 2-Opt, Or-Opt, and 3-Opt operators. The current solution is updated by the simulated annealing (SA) acceptance criterion. At the end of each iteration, scores for shaking operators are updated by the improvement of the solution. A segment contains several

iterations, and the select probability of shaking operators is updated at the end of each segment. These processes repeat until the termination criteria are satisfied.

2. Shaking operators

In the shaking phase, five operators are used. Removal and insertion operators are “pool” operators, which are used for one route to evaluate whether a demand location should be replenished. Let $f(S)$ represent the objective value and the penalty of the solution S , we define the operation cost as $f(S') - f(S_{cur})$. These two operators choose a replenishment mode randomly, then traverse their entire neighborhood of S_{cur} to select the lowest operation cost solution as S' . The removal operator removes a demand location from a route and puts the demand location into the pool accordingly to the replenishment mode. The insertion operator inserts a demand location from the pool accordingly to the replenishment mode into a route. Exchange, swap, and relocation operators are used for “inter-route”, which operates on two routes under the same replenishment mode. These three operators randomly select a solution as S' from their neighborhood. The exchange operator exchanges two segments of different routes randomly. The swap operator randomly chooses two different routes. Then an arc of each route is opened, and swap demand locations are after the breakpoint. The relocation operator randomly chooses a demand location from a route and inserts it into another route. The illustration of these operators is shown in Figure 2.

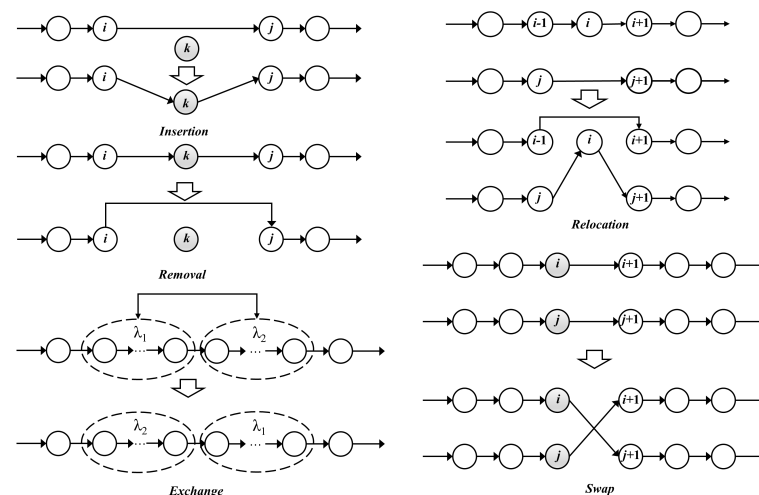


Figure 2. Shaking operators.

3. VND operators

After the shaking phase, the “intra-route” operators with the VND strategy further optimize the affected routes. 2-Opt, Or-Opt, and 3-Opt operators are employed in this phase. These operators traverse their entire neighborhood of S' to select the lowest operation cost solution. Let i, j , and k are three positions in a route, where $i < j < k$. The 2-Opt operator first opens the arc $(i, i + 1)$ and the arc $(k, k + 1)$, then reverse the route between $i + 1$ and k , and finally connect the arc (i, k) and the arc $(i + 1, k + 1)$. The Or-Opt operator opens arc $(i, i + 1)$, $(j, j + 1)$, $(k, k + 1)$ and connect arcs $(i, j + 1)$, $(k, i + 1)$, $(j, k + 1)$. Similarly, the 3-Opt operator opens three arcs and connects them, respectively. However, the connection of this operator contains five situations, where two of them as the same as the 2-Opt and Or-Opt operators. So this operator only checks the other three situations. The illustration of these operators is shown in Figure 3.

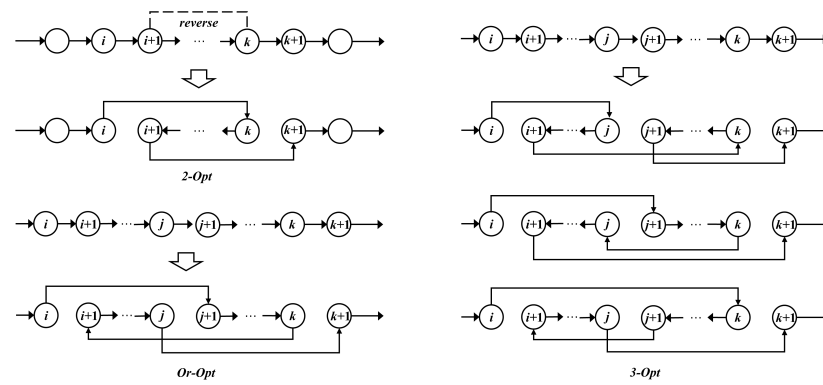


Figure 3. VND operators.

4. SA acceptance criterion

We use an acceptance criterion based on the simulated annealing mechanism. For a current solution S_{cur} , a neighbour solution S'' is accepted if $f(S'') < f(S_{cur})$ and with probability $P_{accept} = e^{-[f(S'') - f(S_{cur})]/\tau}$ otherwise. The current temperature is τ , where $\tau > 0$. The initial temperature is τ_{start} , and there is a cooling rate factor ϕ to decrease the temperature in each iteration, where $0 < \phi < 1$.

5. Adaptive mechanism

The roulette-wheel mechanism determines the choice of which shaking operator is applied in each iteration. The probability for selection is based on the weight of each operator, which depends on its past performance, e.g., let θ_i be the weight of operator i , then operator j will be selected with probability $\theta_j / \sum_{i=1}^n \theta_i$. The weight is updated by scores. In each iteration, the applied shaking operator will gain a score. If the neighbour solution S'' is better than the best solution S_{best} , get σ_1 ; if S'' is better than the current solution S_{cur} , get σ_2 ; if S'' is no better than S_{cur} but accepted by the SA acceptance criterion, get σ_3 ; if S'' is not accepted, get 0. As mentioned before, the search progress is divided into segments, and each segment contains seg_{cap} iterations. After the last iteration of each segment, shaking operators update their weights based on their scores. Let π_i represent the sum of scores in the segment for shaking operator i , and κ_i is the number of times it is used in the segment. Then the weight of operator i in the next segment is updated by

$$\theta_i = \begin{cases} \theta_i & \kappa_i = 0 \\ (1 - \eta)\theta_i + \eta\pi_i / \kappa_i & \kappa_i \neq 0 \end{cases} \quad (29)$$

where $\eta = [0, 1]$ is called the reaction factor, controlling the adjustment speed. We set a protection threshold so that the weight will not be lower than 0.1 to ensure that it is still possible to select the operator with a lower weight in the later-stage iteration

6. Termination criteria

When the temperature reaches the cut-off threshold τ_{cut} at each planning cycle, the iterations will cease.

7. Pseudocode

The pseudocode of the proposed AVNS heuristic shows in Algorithm 1:

Algorithm 1: Adaptive variable neighborhood search

Data: Inventory planning horizon H , initial inventory I_{i0}
Result: Optimal replenishment routes R^* , Optimal replenishment quantity Q^*

- 1 Generate an initial solution S_{init} ;
- 2 S_{cur} : Current solution;
- 3 S_{best} : Best so far solution;
- 4 $iter$: Number of iterations;
- 5 $iter_{seg}$: Iteration number in segment ;
- 6 seg_{cap} : Segment capacity ;
- 7 Determine set of neighborhood structures for shaking $\{N_k | k = 1, \dots, k_{max}\}$ and
for local search $\{N_l | l = 1, \dots, l_{max}\}$;
- 8 $S_{cur}, S_{best} \leftarrow S_{init}$;
- 9 $iter, iter_{seg} \leftarrow 1$;
- 10 $Temperature \leftarrow \tau_{start}$;
- 11 **while** $Temperature \geq \tau_{cut}$ **do** // Apply shaking
- 12 Select a shaking neighborhood N_k by P_{select} ;
- 13 Choose a solution S' from N_k randomly;
- 14 $l \leftarrow 1$;
- 15 **while** $l \leq l_{max}$ **do** // Apply VND
- 16 Find the best solution S'' in $N_l(S')$;
- 17 **if** $f(S'') \leq f(S_{best})$ **then**
- 18 $S_{best} \leftarrow S''$;
- 19 $S_{cur} \leftarrow S''$;
- 20 Update score by σ_1 ;
- 21 **else**
- 22 **if** $f(S'') \leq f(S_{cur})$ **then**
- 23 $S_{cur} \leftarrow S''$;
- 24 Update score by σ_2 ;
- 25 **if** $\rho \leq P_{accept}$ **then**
- 26 $S_{cur} \leftarrow S''$;
- 27 Update score by σ_3 ;
- 28 **else**
- 29 $l \leftarrow l + 1$
- 30 **end**
- 31 **end**
- 32 **end**
- 33 **end**
- 34 **end**
- 35 **end**
- 36 $iter \leftarrow iter + 1$;
- 37 $iter_{seg} \leftarrow iter_{seg} + 1$;
- 38 $Temperature \leftarrow Temperature \times \phi$;
- 39 **if** $iter_{seg} > seg_{cap}$ **then**
- 40 Update P_{select} ;
- 41 Clear score;
- 42 $iter_{seg} \leftarrow 1$;
- 43 **end**
- 44 **end**

4. Numerical Experiments

This section provides the numerical experiments of the methodology proposed in Section 3. The AVNS algorithm is implemented on Python 3.9.0 and on a laptop with an Intel i7 1065G7 CPU processor and 16 GB RAM. All experiment results report in tables are

the average of running 30 times, and outliers removed based on the PauTa criterion. Due to the absence of suitable benchmark instances available for the dual-sourcing inventory routing problem, we use a CVRP benchmark set and adjust some information. Nodes of Augerat's CVRP benchmark instance set A is randomly generated in the range 100×100 , where the first node is the central depot and the others are demand locations.

These numerical experiment parameters' selection is based on the disaster relief background. We set the scale is 1:2 km. Trucks are land replenishment vehicles, and helicopters are aerial replenishment crafts. The land and aerial fleet size are set to be $1 + \lfloor n/15 \rfloor$ and $1 + \lfloor n/30 \rfloor$, respectively. For a truck, the speed is 60 km/h, the fixed cost is 500, the using cost is 50/h, and the capacity is 5000. For a helicopter, the speed is 260 km/h, the fixed cost is 10,000, the using cost is 250/h, and the capacity is 2000. For a demand location, the average demand $\bar{d} = 20$, the maximum deviation $\hat{d} = 10$, the unit holding cost is 0.05, and the unit shortage cost is 2. The planning horizon at each planning cycle is 24 h, that $T = 24$. In Augerat's benchmark instance set A, demands are generated in the interval $[1, 30]$. We use the demand information as the initial inventory level, where the volume times 20 to fit the number scale.

The score update parameters setting refers to Coelho et al. (2012), where $\sigma_1 = 10$, $\sigma_2 = 5$, and $\sigma_3 = 2$ [40]. The starting temperature τ_{start} is set to 30,000 and the cut-off temperature $\tau_{cut} = 0.01$. The cooling rate ϕ is 0.995 which yields roughly 3000 iterations at each planning cycle. The segment capacity $seg_{cap} = 50$ and the reaction factor $\eta = 0.1$.

4.1. Algorithm Analyzes

The normalized weights of shaking operators are the selection probability of the operator in iterations. It can reflect the ability of an operator to find an acceptable/better solution than the current/best solution in the past iterations. As shown in Figure 4, normalized weights of five shaking operators are tested on three different instance scales. Removal and insertion operators outperform the other three operators. The normalized weights tend to be equal at the later-stage iterations. That is because as the temperature cools down and the iteration goes on, operators are more difficult to obtain scores. Their weight will gradually approach the protection threshold. This feature increases the diversity in searching at the later stage of iterations.

However, the high weight can not entirely indicate that an operator can find a new best solution more effectively. The new acceptable solution also can obtain scores. So the $\%UBest/\%IUsage$ ratio can reflect the ability of operators to find a new best solution, where $\%UBest$ is the percentage of an operator finding a new best solution times to the total best solution updated times, and $\%IUsage$ is the percentage of an operator is used times to total iterations. Table 1 shows the $\%UBest/\%IUsage$ ratio of shaking operators, which reflects the ability to find a new best solution in iterations.

The insertion operator shows the most effective ability to find new best solutions, whereas the exchange and swap operators demonstrate the least impressive results. However, removal and insertion operators traverse their entire neighborhood and select the lowest operation cost solution, instead of exchange, swap, and relocation operators selecting randomly. The removal and insertion operators are employed to evaluate if a demand location needs to be replenished, which is much like the "destroy-repair" strategy in the adaptive large neighborhood search algorithm. The other three operators provide diversity, similar to the shaking strategy in traditional VNS. The results of Figure 4 and Table 1 show that, although the weights of exchange, swap, and relocation operators are much lower than removal and insertion operators, the $\%UBest/\%IUsage$ ratio of them are 0.30, 0.29, and 0.42, respectively. That means these three operators also effectively find new best solutions by providing diversity.

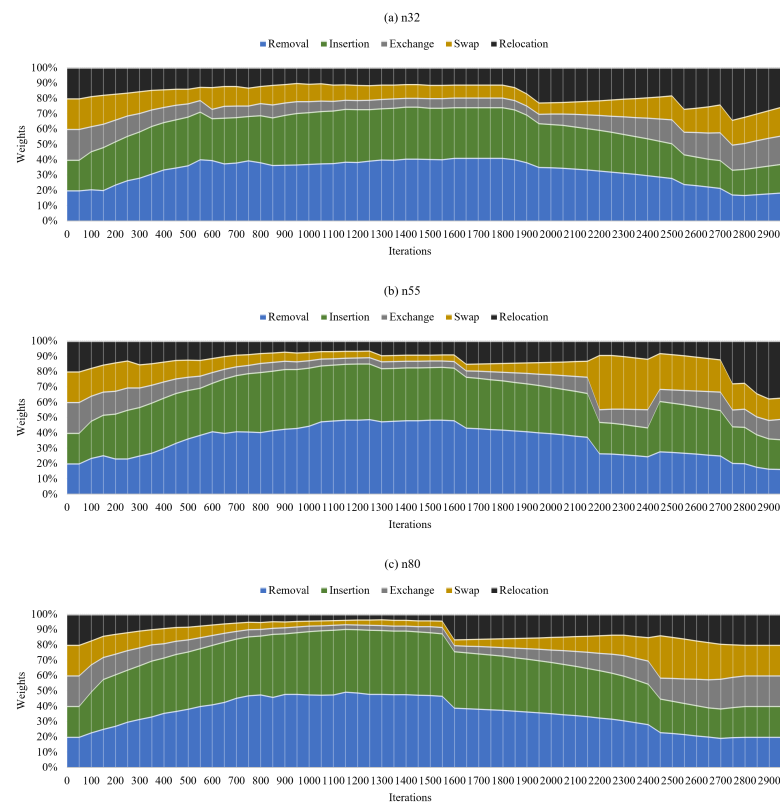


Figure 4. Normalized weights of shaking operators.

Table 1. Performance of shaking operators (%*UBest*/%*IUsage* ratio).

Instance	Removal	Insertion	Exchange	Swap	Relocation
n32	1.28	1.41	0.34	0.18	0.44
n33a	1.30	1.89	0.12	0.18	0.61
n33b	1.22	2.07	0.21	0.19	0.70
n34	1.16	1.44	0.32	0.60	0.89
n36	1.49	1.21	0.61	0.20	0.64
n37a	0.87	1.50	0.24	0.19	1.06
n37b	1.02	1.43	0.37	0.35	0.96
n38	1.10	1.62	0.36	0.31	0.27
n39a	0.83	1.76	0.58	0.13	0.58
n39b	1.05	1.64	0.33	0.22	0.41
n44	0.76	1.96	0.43	0.64	0.21
n45a	1.18	1.63	0.24	0.23	0.37
n45b	1.15	1.59	0.34	0.24	0.51
n46	1.20	1.29	0.44	0.41	0.57
n48	1.22	1.55	0.19	0.07	0.41
n53	1.14	1.48	0.67	0.29	0.18
n54	1.16	1.66	0.06	0.31	0.18
n55	0.97	1.78	0.26	0.23	0.26
n60	1.06	1.68	0.26	0.41	0.23
n61	1.03	1.61	0.26	0.24	0.45
n62	0.93	1.67	0.30	0.25	0.15
n63a	1.00	1.96	0.21	0.35	0.15
n63b	1.14	1.68	0.17	0.27	0.40
n64	0.91	1.76	0.19	0.40	0.20
n65	0.68	1.99	0.24	0.44	0.26
n69	0.91	1.89	0.10	0.20	0.16
n80	0.82	2.08	0.22	0.16	0.17
Average	1.06	1.68	0.30	0.29	0.42

4.2. Model Analyzes

The single-period model under the rolling horizon framework is tested on all instances with three and seven planning cycles, respectively. We first study the robust solution and the deterministic solution on three planning cycles, where the robust solution is proposed by this work and the deterministic solution is solved by the average demands $\bar{d}$. Then the planning horizon crossing strategy is studied on instances with three and seven planning cycles. All gaps are calculated by

$$\text{Gap} = \frac{\text{Obj}_{\text{other}} - \text{Obj}_{\text{baseline}}}{\text{Obj}_{\text{baseline}}} \times 100\%. \quad (30)$$

Table 2 shows the total cost of the robust solution and the deterministic solution when the real demands are $d = \bar{d}$ or $d = \bar{d} + \hat{d}$. The baseline is the robust solution. The gap between the robust solution and the deterministic solution when the real demands are $\bar{d}$ is only 1.12%. This gap is so small that we can hardly use more times experiments to exclude the influence of solution errors on the results. Because the solution is solved by the meta-heuristic algorithm and we cannot evaluate the gap between the obtained best solution and the optimal solution. It can explain why the robust solution is better than the deterministic solution in a few instances. When the real demands are the worst case ($d = \bar{d} + \hat{d}$), the robust solution outperforms the deterministic solution 49.86% on average.

Table 2. Performance of robust solution under mean/maximum demand (3 cycles).

Instances	Average Demands			Maximum Demands		
	Robust	Deterministic	Gap	Robust	Deterministic	Gap
n32	163,211.08	162,561.91	−0.40%	593,052.13	1,061,567.47	79.00%
n33a	163,960.52	160,677.23	−2.00%	654,320.31	1,099,284.42	68.00%
n33b	163,045.56	159,020.76	−2.47%	571,797.45	942,360.21	64.81%
n34	174,358.00	173,273.86	−0.62%	725,834.88	1,159,553.98	59.75%
n36	188,342.72	188,228.09	−0.06%	880,603.57	1,431,663.78	62.58%
n37a	204,748.18	204,123.95	−0.30%	940,380.94	1,442,303.22	53.37%
n37b	206,469.35	200,307.17	−2.98%	898,549.54	1,470,116.91	63.61%
n38	196,769.57	196,682.15	−0.04%	1,009,831.03	1,528,817.70	51.39%
n39a	221,019.42	217,361.24	−1.66%	1,079,649.65	1,686,721.84	56.23%
n39b	227,172.40	228,662.91	0.66%	1,071,752.88	1,547,831.06	44.42%
n44	263,389.99	262,025.32	−0.52%	1,519,597.96	2,213,194.72	45.64%
n45a	257,700.96	253,246.45	−1.73%	1,187,074.80	1,858,238.81	56.54%
n45b	239,301.02	228,462.98	−4.53%	1,113,656.30	1,840,635.76	65.28%
n46	237,234.85	235,542.43	−0.71%	1,199,051.82	1,953,179.97	62.89%
n48	269,601.58	268,197.32	−0.52%	1,401,310.60	2,136,987.47	52.50%
n53	405,822.68	397,117.12	−2.15%	1,806,340.99	2,512,484.42	39.09%
n54	413,459.23	408,530.44	−1.19%	1,902,235.27	2,621,106.86	37.79%
n55	336,885.68	333,575.64	−0.98%	1,875,911.41	2,676,433.96	42.67%
n60	385,908.80	386,102.40	0.05%	1,672,097.42	2,526,153.39	51.08%
n61	361,645.48	363,106.42	0.40%	1,768,019.65	2,627,647.38	48.62%
n62	478,530.30	464,378.70	−2.96%	2,019,104.19	2,718,603.33	34.64%
n63a	432,636.94	434,074.39	0.33%	2,050,851.04	3,064,180.25	49.41%
n63b	378,874.95	369,294.36	−2.53%	1,941,360.67	2,783,145.96	43.36%
n64	430,339.57	424,170.68	−1.43%	2,099,369.22	2,808,516.82	33.78%
n65	449,687.04	448,473.30	−0.27%	2,230,200.12	2,917,425.54	30.81%
n69	665,654.17	657,378.93	−1.24%	2,681,859.09	3,401,871.00	26.85%
n80	947,864.67	944,031.08	−0.40%	3,375,881.26	4,124,163.85	22.17%
Average	-	-	−1.12%	-	-	49.86%

To evaluate the performance of our planning horizon crossing strategy, we test with and without this strategy on instances with three planning cycles and seven planning cycles. The baseline is the solution with planning horizon crossing strategy. Note that the planning horizon in the first and the last planning cycle are not crossing ($H = T$).

As shown in Table 3, the gap between the model with and without planning horizon crossing in three cycles and seven cycles are 6.31% and 15.21%, respectively. The result shows that the planning horizon crossing strategy can balance short-term decisions and long-term decisions, and reduce the out-of-stock cost before replenishment arrives in each planning cycle.

Table 3. Comparing the model with/without planning horizon crossing.

Instances	3 Cycles			7 Cycles		
	Crossing	No Crossing	Gap	Crossing	No Crossing	Gap
n32	163,211.08	164,938.34	1.06%	327,308.35	356,441.07	8.90%
n33a	163,960.52	169,704.94	3.50%	303,855.91	348,547.88	14.71%
n33b	163,045.56	166,200.35	1.93%	298,161.20	347,464.29	16.54%
n34	174,358.00	180,087.03	3.29%	334,671.20	382,719.74	14.36%
n36	188,342.72	202,710.57	7.63%	375,903.20	452,502.22	20.38%
n37a	204,748.18	223,892.86	9.35%	399,978.16	465,360.22	16.35%
n37b	206,469.35	215,343.30	4.30%	398,508.10	473,262.75	18.76%
n38	195,769.57	210,628.85	7.59%	393,328.47	469,974.37	19.49%
n39a	221,019.42	236,318.88	6.92%	449,661.71	524,091.30	16.55%
n39b	227,172.40	240,192.50	5.73%	463,826.21	523,908.45	12.95%
n44	263,389.99	282,210.79	7.15%	996,281.28	1,212,032.23	21.66%
n45a	257,700.96	273,531.88	6.14%	566,026.06	631,478.46	11.56%
n45b	239,301.02	255,655.97	6.83%	503,419.06	574,148.90	14.05%
n46	237,234.85	254,382.18	7.23%	533,961.97	585,141.99	9.58%
n48	269,601.58	294,109.66	9.09%	621,412.08	657,455.90	5.80%
n53	405,822.68	420,274.01	3.56%	1,024,807.46	1,264,783.87	23.42%
n54	413,459.23	459,495.61	11.13%	1,130,646.78	1,557,692.75	37.77%
n55	336,885.68	359,488.04	6.71%	1,258,609.13	1,612,273.16	28.10%
n60	385,908.80	409,793.86	6.19%	800,523.92	885,149.57	10.57%
n61	361,645.48	388,489.22	7.42%	786,827.68	848,578.68	7.85%
n62	478,530.30	506,094.86	5.76%	1,003,965.14	1,084,640.69	8.04%
n63a	432,636.94	459,679.05	6.25%	1,091,721.73	1,141,790.13	4.59%
n63b	378,874.95	408,041.14	7.70%	989,831.89	1,115,628.85	12.71%
n64	430,339.57	455,573.92	5.86%	1,145,910.45	1,268,798.23	10.72%
n65	449,687.04	461,608.99	2.65%	1,197,400.02	1,400,670.03	16.98%
n69	665,654.17	735,229.66	10.45%	1,781,092.36	2,003,821.23	12.51%
n80	947,864.67	1,032,533.93	8.93%	2,710,575.82	3,140,257.61	15.85%
Average	-	-	6.31%	-	-	15.21%

5. Discussion

This study focuses on a novel inventory routing problem in which the lead time is dependent on the routing and has two modes of replenishment. Considering the transportation time in the lead time is of great significance for application scenarios with high timeliness requirements. For example, in emergency rescue, the central depot replenishes materials to shelters, and fresh food retailers replenish goods from the central depot to the front warehouse, etc. Supply of goods to demand locations more precisely can reduce waste and contribute to sustainability.

A robust model and an adaptive variable neighborhood search algorithm are proposed for the problem. There are five attractive features of our work: (1) the joint decision on inventory and transportation is investigated by considering replenishment-route-dependent lead time, which can better highlight the relationship between the arrival time and the related replenishment quantity; (2) a robust optimization model based on the rolling horizon framework is proposed, which is more appropriate to describe the features of the application scenarios under uncertainty, such as emergency logistics and perishable goods supply chains. (3) A closed-form optimal solution of replenishment quantity is derived under the box uncertainty set. (4) A planning horizon crossing strategy is proposed for balancing

short-term decisions and long-term decisions in the rolling horizon framework. (5) A novel shaking stage of the adaptive variable neighborhood search algorithm is developed.

Numerical experiments study the proposed model and algorithm on a modified benchmark instance set. The robust solution is only worse than the deterministic solution 1.12% when the demand is the predicted mean. On the other hand, when the demand is the worst case, the robust solution is 49.86% better than the deterministic solution. It shows that our robust model can greatly reduce the effect of uncertainty at a very low cost. The planning horizon crossing strategy is effective and can save costs. The three cycles and seven cycles experiments show the planning horizon crossing strategy can save 6.31% and 15.21% costs than without it. For shaking operators in the AVNS algorithm, the insertion operator outperforms the others, and the $%UBest/%IUsage$ ratio is 1.69. The $%UBest/%IUsage$ ratio of removal, exchange, swap, and relocation operators are 1.06, 0.30, 0.29, and 0.42, respectively. It shows the insertion operator has the best performance of the ability to find new best solutions in iterations.

As with any research, ours begs for further extensions. Box uncertainty set is too conservative, other more novel uncertainty performs better on overcoming over-conservativeness. However, the closed-form solution for more complex uncertainty sets is a challenge. The rolling horizon framework is solved multi-period problems as a multi-stage decision-making problem by rolling planning horizon. Adjustable robust optimization is a methodology for multi-stage optimization. We will introduce adjustable robust optimization to our problem in future research. Compared with other articles that focus on the timeliness of replenishment, this paper introduces air–land transportation into the inventory routing problem to avoid the shortcomings that only use land transportation, such as road congestion or road damage after disasters. The adjacency matrix coefficient can reflect these situations. However, the model proposed in this paper cannot reflect the road congestion changes over time, and the dynamic model combined with real-time data will be a challenge.

Author Contributions: Conceptualization, W.Z.; methodology, W.Z. and H.Z.; software, W.Z.; validation, W.Z.; formal analysis, W.Z.; investigation, W.Z.; resources, W.Z.; data curation, W.Z.; writing—original draft preparation, W.Z.; writing—review and editing, W.Z. and H.Z.; supervision, H.Z.; funding acquisition, H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 71971011.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. The test set can be found at <http://vrp.galagos.inf.puc-rio.br/index.php/en/> (accessed on 16 January 2023). The source code of the proposed AVNS algorithm can be found at <https://github.com/WeberZheng/AVNS> (accessed on 16 January 2023).

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

Abbreviations

The following abbreviations are used in this manuscript:

AVNS	Adaptive variable neighborhood search
VNS	Variable neighborhood search
VND	Variable neighborhood descent
VIM	Vendor-managed inventory
IRP	Inventory routing problem
SA	Simulated annealing

Appendix A. Proof

Proof of Proposition 1. For a demand location, the inventory cost before the replenishment arrival is not affected by the replenishment quantity, it only depends on the initial inventory level. So the sub-problem is determining replenishment quantities to minimize the inventory cost after the replenishment arrival.

If the demand location is replenished, there are two situations:

1. The demand location is replenished by one replenishment mode.
If the initial inventory is enough, where

$$h(I_0 - H\bar{d} + H\hat{d}) \geq -s(I_0 - H\bar{d} - H\hat{d}). \quad (\text{A1})$$

Let $\lambda = (s + h)\bar{d} + (s - h)\hat{d}$, the optimal replenishment quantity is 0 when

$$I_0 \geq \frac{\lambda}{s + h}H. \quad (\text{A2})$$

If $I_0 < \lambda H / (s + h)$ there is a turning point t_1 between the replenishment arrival time and the inventory planning horizon H . Where the inventory cost is from the holding cost turning to the shortage cost under the worst case. The turning point t_1 and the replenishment quantity q satisfy:

$$h(I_0 + q - t_1\bar{d} + t_1\hat{d}) = -s(I_0 + q - t_1\bar{d} - t_1\hat{d}). \quad (\text{A3})$$

Then,

$$t_1 = \frac{(s + h)(I_0 + q)}{(s + h)\bar{d} + (s - h)\hat{d}}. \quad (\text{A4})$$

For any replenishment lead time l , the inventory cost after the replenishment is

$$IC^{\text{replenished}} = \int_l^{t_1} h(I_0 + q - t\bar{d} + t\hat{d})dt - \int_{t_1}^H s(I_0 + q - t\bar{d} - t\hat{d})dt. \quad (\text{A5})$$

The optimal replenishment quantity is

$$q = \arg \left\{ \frac{dIC^{\text{replenished}}}{dq} = 0 \right\}. \quad (\text{A6})$$

The closed-form solution for optimal replenishment quantity is

$$q = \begin{cases} \frac{(sH + hl)}{(s + h)^2}\lambda - I_{i0} & I_{i0} < \frac{\lambda}{s + h}H \\ 0 & I_{i0} \geq \frac{\lambda}{s + h}H \end{cases} \quad (\text{A7})$$

2. The demand location is replenished by both two replenishment modes.
If the initial inventory is enough before the slower replenishment arrival, where

$$h(I_0 - l^s\bar{d} + l^s\hat{d}) \geq -s(I_0 - l^s\bar{d} - l^s\hat{d}). \quad (\text{A8})$$

The optimal faster replenishment quantity is 0 when

$$I_0 \geq \frac{\lambda}{s + h}l^s. \quad (\text{A9})$$

Similarly, the optimal slower replenishment quantity is 0 when

$$I_0 + q^f \geq \frac{\lambda}{s + h}H. \quad (\text{A10})$$

For the faster arrival replenishment, if $I_0 < \lambda l^s / (s + h)$, there is a turning point t_1 between the twice replenishment.

$$t_1 = \frac{(s + h)(I_0 + q^f)}{(s + h)\bar{d} + (s - h)\hat{d}}. \quad (\text{A11})$$

For any replenishment lead time l^f , the inventory cost between the faster and the slower replenishment arrival time is

$$IC^{\text{replenished}_1} = \int_{l^f}^{t_1} h(I_0 + q^f - t\bar{d} + t\hat{d})dt - \int_{t_1}^{l^s} s(I_0 + q^f - t\bar{d} - t\hat{d})dt. \quad (\text{A12})$$

The closed-form solution for the faster replenishment quantity is

$$q_i^f = \begin{cases} \frac{(sl^s + hl^f)}{(s + h)^2} \lambda - I_{i0} & I_{i0} < \frac{\lambda}{s + h} l^s \\ 0 & I_{i0} \geq \frac{\lambda}{s + h} l^s \end{cases} \quad (\text{A13})$$

For the slower arrival replenishment, if $I_0 + q^f < \lambda H / (s + h)$, there is a turning point t_2 between the slower replenishment arrival time and the inventory planning horizon H .

$$t_2 = \frac{(s + h)(I_0 + q^f + q^s)}{(s + h)\bar{d} + (s - h)\hat{d}}. \quad (\text{A14})$$

For any replenishment lead time l^s , the inventory cost after the slower replenishment arrival is

$$IC^{\text{replenished}_2} = \int_{l^s}^{t_2} h(I_0 + q^f + q^s - t\bar{d} + t\hat{d})dt - \int_{t_2}^H s(I_0 + q^f + q^s - t\bar{d} - t\hat{d})dt. \quad (\text{A15})$$

The closed-form solution for the slower replenishment quantity is

$$q_i^s = \begin{cases} \frac{(sH + hl^s)}{(s + h)^2} \lambda - I_{i0} - q_i^f & I_{i0} + q_i^f < \frac{\lambda}{s + h} H \\ 0 & I_{i0} + q_i^f \geq \frac{\lambda}{s + h} H \end{cases} \quad (\text{A16})$$

□

References

1. Comi, A. A modelling framework to forecast urban goods flows. *Res. Transp. Econ.* **2020**, *80*, 100827. [\[CrossRef\]](#)
2. Bell, W.J.; Dalberto, L.M.; Fisher, M.L.; Greenfield, A.J.; Jaikumar, R.; Kedia, P.; Mack, R.G.; Prutzman, P.J. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces* **1983**, *13*, 4–23. [\[CrossRef\]](#)
3. Waller, M.; Johnson, M.E.; Davis, T. Vendor-managed inventory in the retail supply chain. *J. Bus. Logist.* **1999**, *20*, 183.
4. Dror, M.; Ball, M.; Golden, B. A computational comparison of algorithms for the inventory routing problem. *Ann. Oper. Res.* **1985**, *4*, 1–23. [\[CrossRef\]](#)
5. Çankaya, E.; Ekici, A.; Özener, O.Ö. Humanitarian relief supplies distribution: An application of inventory routing problem. *Ann. Oper. Res.* **2019**, *283*, 119–141. [\[CrossRef\]](#)
6. Moin, N.H.; Salhi, S. Inventory routing problems: A logistical overview. *J. Oper. Res. Soc.* **2007**, *58*, 1185–1194. [\[CrossRef\]](#)
7. Coelho, L.C.; Cordeau, J.F.; Laporte, G. Thirty years of inventory routing. *Transp. Sci.* **2014**, *48*, 1–19. [\[CrossRef\]](#)
8. Shaabani, H. A literature review of the perishable inventory routing problem. *Asian J. Shipp. Logist.* **2022**, *38*, 143–161. [\[CrossRef\]](#)
9. Chao, C.; Zhihui, T.; Baozhen, Y. Optimization of two-stage location–routing–inventory problem with time-windows in food distribution network. *Ann. Oper. Res.* **2019**, *273*, 111–134. [\[CrossRef\]](#)
10. Imran, M.; Habib, M.S.; Hussain, A.; Ahmed, N.; Al-Ahmari, A.M. Inventory routing problem in supply chain of perishable products under cost uncertainty. *Mathematics* **2020**, *8*, 382. [\[CrossRef\]](#)
11. Alvarez, A.; Cordeau, J.F.; Jans, R.; Munari, P.; Morabito, R. Formulations, branch-and-cut and a hybrid heuristic algorithm for an inventory routing problem with perishable products. *Eur. J. Oper. Res.* **2020**, *283*, 511–529. [\[CrossRef\]](#)

12. Daroudi, S.; Kazemipoor, H.; Najafi, E.; Fallah, M. The minimum latency in location routing fuzzy inventory problem for perishable multi-product materials. *Appl. Soft Comput.* **2021**, *110*, 107543. [\[CrossRef\]](#)
13. Harahap, A.Z.M.K.; Rahim, M.K.I.A. A single period deterministic inventory routing model for solving problems in the agriculture industry. *J. Appl. Sci. Eng.* **2022**, *25*, 945–950. [\[CrossRef\]](#)
14. Li, M.; Wang, Z.; Chan, F.T. An inventory routing policy under replenishment lead time. *IEEE Trans. Syst. Man Cybern. Syst.* **2016**, *47*, 3150–3164. [\[CrossRef\]](#)
15. Dror, M.; Ball, M. Inventory/routing: Reduction from an annual to a short-period problem. *Nav. Res. Logist. NRL* **1987**, *34*, 891–905. [\[CrossRef\]](#)
16. Coelho, L.C.; Laporte, G. The exact solution of several classes of inventory-routing problems. *Comput. Oper. Res.* **2013**, *40*, 558–565. [\[CrossRef\]](#)
17. Federgruen, A.; Zipkin, P. A combined vehicle routing and inventory allocation problem. *Oper. Res.* **1984**, *32*, 1019–1037. [\[CrossRef\]](#)
18. Jaillet, P.; Bard, J.F.; Huang, L.; Dror, M. Delivery cost approximations for inventory routing problems in a rolling horizon framework. *Transp. Sci.* **2002**, *36*, 292–300. [\[CrossRef\]](#)
19. Sakiani, R.; Seifi, A.; Khorshiddoust, R.R. Inventory routing and dynamic redistribution of relief goods in post-disaster operations. *Comput. Ind. Eng.* **2020**, *140*, 106219. [\[CrossRef\]](#)
20. SteadieSeifi, M.; Dellaert, N.; Van Woensel, T. Multi-modal transport of perishable products with demand uncertainty and empty repositioning: A scenario-based rolling horizon framework. *EURO J. Transp. Logist.* **2021**, *10*, 100044. [\[CrossRef\]](#)
21. Archetti, C.; Bertazzi, L.; Laporte, G.; Speranza, M.G. A branch-and-cut algorithm for a vendor-managed inventory-routing problem. *Transp. Sci.* **2007**, *41*, 382–391. [\[CrossRef\]](#)
22. Hvattum, L.M.; Løkketangen, A.; Laporte, G. Scenario tree-based heuristics for stochastic inventory-routing problems. *Inform. J. Comput.* **2009**, *21*, 268–285. [\[CrossRef\]](#)
23. Huang, S.H.; Lin, P.C. A modified ant colony optimization algorithm for multi-item inventory routing problems with demand uncertainty. *Transp. Res. Part E Logist. Transp. Rev.* **2010**, *46*, 598–611. [\[CrossRef\]](#)
24. Alvarez, A.; Cordeau, J.F.; Jans, R.; Munari, P.; Morabito, R. Inventory routing under stochastic supply and demand. *Omega* **2021**, *102*, 102304. [\[CrossRef\]](#)
25. Soyster, A.L. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* **1973**, *21*, 1154–1157. [\[CrossRef\]](#)
26. Mamani, H.; Nassiri, S.; Wagner, M.R. Closed-form solutions for robust inventory management. *Manag. Sci.* **2016**, *63*, 1625–1643. [\[CrossRef\]](#)
27. Levi, R.; Pál, M.; Roundy, R.O.; Shmoys, D.B. Approximation algorithms for stochastic inventory control models. *Math. Oper. Res.* **2007**, *32*, 284–302. [\[CrossRef\]](#)
28. Levi, R.; Janakiraman, G.; Nagarajan, M. A 2-approximation algorithm for stochastic inventory control models with lost sales. *Math. Oper. Res.* **2008**, *33*, 351–374. [\[CrossRef\]](#)
29. Sun, J.; Van Mieghem, J.A. Robust dual sourcing inventory management: Optimality of capped dual index policies and smoothing. *Manuf. Serv. Oper. Manag.* **2019**, *21*, 912–931. [\[CrossRef\]](#)
30. Tagaras, G.; Vlachos, D. A periodic review inventory system with emergency replenishments. *Manag. Sci.* **2001**, *47*, 415–429. [\[CrossRef\]](#)
31. Musolino, G.; Polimeni, A.; Vitetta, A. Freight vehicle routing with reliable link travel times: A method based on network fundamental diagram. *Transp. Lett.* **2018**, *10*, 159–171. [\[CrossRef\]](#)
32. Comi, A.; Polimeni, A. Estimating path choice models through floating car data. *Forecasting* **2022**, *4*, 525–537. [\[CrossRef\]](#)
33. Croce, A.I.; Musolino, G.; Rindone, C.; Vitetta, A. Traffic and Energy Consumption Modelling of Electric Vehicles: Parameter Updating from Floating and Probe Vehicle Data. *Energies* **2022**, *15*, 82. [\[CrossRef\]](#)
34. Russo, F.; Rindone, C. Regional transport plans: From direction role denied to common rules identified. *Sustainability* **2021**, *13*, 9052. [\[CrossRef\]](#)
35. Zhang, M.; Yu, J.; Zhang, Y.; Yu, H. Programming model of emergency scheduling with combined air-ground transportation. *Adv. Mech. Eng.* **2017**, *9*, 1687814017739512. [\[CrossRef\]](#)
36. Sacramento, D.; Pisinger, D.; Ropke, S. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. *Transp. Res. Part C Emerg. Technol.* **2019**, *102*, 289–315. [\[CrossRef\]](#)
37. Zheng, W.; Zhou, H. Routing problem with multiple transportation modes considering road damage. In Proceedings of the 2018 International Conference on Computer Modeling, Simulation and Algorithm (CMSA 2018), Beijing, China, 22–23 April 2018; Atlantis Press: Amsterdam, The Netherlands, 2018; pp. 250–253. [\[CrossRef\]](#)
38. Mahjoob, M.; Fazeli, S.S.; Milanlouei, S.; Tavassoli, L.S.; Mirmozaffari, M. A modified adaptive genetic algorithm for multi-product multi-period inventory routing problem. *Sustain. Oper. Comput.* **2022**, *3*, 1–9. [\[CrossRef\]](#)
39. Zheng, W.; Zhou, H. Robust inventory routing problem with replenishment lead time. In Proceedings of the 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Macao, China, 15–18 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 825–829. [\[CrossRef\]](#)
40. Coelho, L.C.; Cordeau, J.F.; Laporte, G. The inventory-routing problem with transshipment. *Comput. Oper. Res.* **2012**, *39*, 2537–2548. [\[CrossRef\]](#)

41. Hernández-Pérez, H.; Rodríguez-Martín, I.; Salazar-González, J.J. A hybrid heuristic approach for the multi-commodity pickup-and-delivery traveling salesman problem. *Eur. J. Oper. Res.* **2016**, *251*, 44–52. [[CrossRef](#)]
42. Mirzapour Al-e hashem, S.M.J.; Rekik, Y. Multi-product multi-period Inventory Routing Problem with a transshipment option: A green approach. *Int. J. Prod. Econ.* **2014**, *157*, 80–88. [[CrossRef](#)]
43. Shen, Q.; Chu, F.; Chen, H. A Lagrangian relaxation approach for a multi-mode inventory routing problem with transshipment in crude oil transportation. *Comput. Chem. Eng.* **2011**, *35*, 2113–2123. [[CrossRef](#)]
44. Agra, A.; Christiansen, M.; Wolsey, L. Improved models for a single vehicle continuous-time inventory routing problem with pickups and deliveries. *Eur. J. Oper. Res.* **2022**, *297*, 164–179. [[CrossRef](#)]
45. Li, C.; Gong, L.; Luo, Z.; Lim, A. A branch-and-price-and-cut algorithm for a pickup and delivery problem in retailing. *Omega* **2019**, *89*, 71–91. [[CrossRef](#)]

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