

Multimodal Transportation of Perishable Products with Empty Repositioning of Multiple Sizes – A Formulation and a Heuristic

M. SteadieSeifi, N.P. Dellaert, T. van Woensel

Beta Working Paper series 522

BETA publicatie	WP 522 (working paper)
ISBN	
ISSN	
NUR	
Eindhoven	January 2017

Multimodal Transportation of Perishable Products with Empty Repositioning of Multiple Sizes – A Formulation and a Heuristic

Maryam SteadieSeifi*, Nico Dellaert, Tom van Woensel

School of Industrial Engineering and Innovation Sciences,
Technical University of Eindhoven, P.O.Box 513, 5600 MB, Eindhoven, Netherlands

*Email: m.steadie.seifi@tue.nl

Edition January 18, 2017

Abstract

The multimodal long-haul transportation of perishable products with management of Returnable Transport Items (RTIs) has been addressed in SteadieSeifi et al. [2014]. In that research, our RTIs had equal sizes. In this research, we extend the proposed model to include three sizes of RTIs. This size difference adds extra features to the problem regarding their loading, which increases the complexity of the problem. Therefore, solving real-world instances of such a planning problem to optimality is impossible. In this work, we propose a multi-stage constructive algorithm, where we segregate the solution structure into several layers based on the size of RTIs, and via four stages, we route and reposition small, medium and big RTIs. We provide detailed computational results and analysis. Our proposed Mixed-Integer Program (MIP) and algorithm are the first steps in modeling and solving such a complicated planning problem.

Keywords: Multimodal transportation, Network flow problem, Heterogeneous reusable transport items, Perishability, Mixed-Integer Program

1 Introduction

In SteadieSeifi et al. [2014], we proposed a Mixed-Integer Program (MIP) for the multimodal transportation of perishable products with integrated management of Returnable Transport Items (RTIs). In that paper, we assumed RTIs to have an equal size. In a perishable industry however, RTIs can have different sizes ranging from a small box to a large 45-feet container. In the horticultural industry of the Netherlands for instance, flowers and bouquets are loaded in small RTIs (e.g. boxes and buckets) and small RTIs are then loaded onto medium RTIs (e.g. cages, Dense fusters, staple wagons, trolleys). In SteadieSeifi et al. [2014], we emphasized the importance of timely repositioning of empty RTIs, and no RTI of any size is an exception. Their number is limited, and their availability plays a crucial role in decreasing handling times and preserving the product quality.

Asset management with multiple sizes in essence is unique, and different from some of the state-of-the-art heterogeneous fleet management problems (such as in Baldacci et al. [2010]). *Heterogeneous* fleet or in general assets, have more than one type, each with its own capacity, costs, and some other distinct characteristics. A multimodal fleet can be viewed as a special case of a heterogeneous fleet. However, these assets are not loaded into (onto) each other. We here call this phenomenon the loading hierarchy.

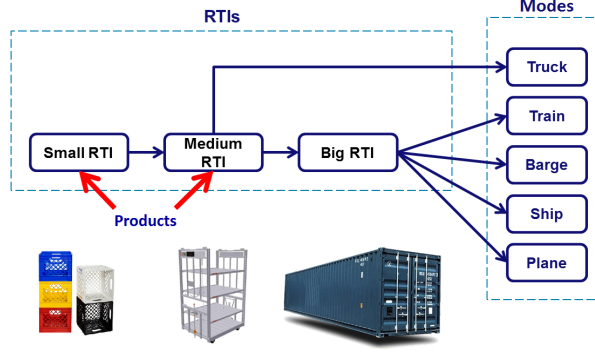


Figure 1: Resource allocation structure in a long-haul transportation system



Figure 2: Empty nesting and folding

Loading hierarchy brings extra features to the model. In perishable long-haul transportation for example: 1) Small RTIs are loaded onto medium ones if transported by trucks, 2) Medium RTIs are then loaded into big RTIs if transported by trains or barges (Figure 1). 3) Capacity of medium and big RTIs is defined by the number of smaller ones they can carry, and capacity of transport modes is defined based on the number of biggest RTIs they can accommodate. 4) Bigger RTIs also have a storage location, and finally, 5) unlike small RTIs, medium and big ones are labeled, which enables tracking the orders that are consolidated and carried on each of them.

One of the biggest challenges resulted by this loading relation is *synchronization* of operations for all RTI sizes. To pick up a transportation order, all required RTIs should be available in time based on their loading hierarchy, and their operations cannot be segregated. For example, medium RTIs cannot start loading, if small RTIs have not arrived to the pick up location and are ready.

Besides these new features, we introduce a nesting ratio to the multi-size RTI problem. *Nesting ratio* is the average ratio of space occupied by an empty RTI compared to a full one. Figures 2a and 2b show how much space nesting or folding RTIs can save. For instance, if a loaded trolley takes on average one unit of space, its empty one takes on average 0.3 unit of space. We then use this ratio in related capacity constraints. Folding or nesting empty RTIs help more consolidation, less usage of transport modes, and therefore, a cheaper repositioning operation.

In this paper, we extend the same-size RTI problem to include three sizes (small, medium, and large). We model this multi-size RTI planning problem and propose a heuristic to solve it. Our main contribution is to include the interactions between these classes of RTIs (e.g. their loading hierarchy) into our previous same-size RTI model. In this regard, in Sections 2 and 3, we describe the problem and review the related literature. In Section 4 then, we explain the

modeling approach and the mathematical formulation. Then, in Sections 5 and 6, we present our solution algorithm, and compare its performance with a state-of-the-art solver. Finally, in Section 7, we provide some concluding remarks.

2 Problem Description

Tactical planning deals with utilizing a given transportation infrastructure by choosing the best services and associated transportation modes, efficient allocation of their capacities to customer orders, and management of all resources involved over a particular time span or horizon.

The main decisions are the flow of products (customer demand), the repositioning of RTIs (for all sizes), the selection of transport modes and schedules to do these jobs, and the number of vehicles needed for each transport mode.

RTIs being the loading units used for transportation, are the key elements of this problem, and the main decisions are defined on their flow throughout the network. Their number throughout the network is limited and their flow is subjected to strict resource balance constraints. For instance, the number of RTIs available at the beginning of the planning horizon should be equal to their number at the end of horizon. The initial number of RTIs however is given, and its value is not a decision.

Demand is here represented by orders. An order is characterized by its pair of origin and destination locations, its volume, its pickup and delivery schedules, and its freshness requirements. Since RTIs are the loading units used to transport the products, the order volume shows the number of *small* RTIs (e.g. 32 buckets) needed to transport products from the origin to the destination. The products can be picked up and loaded onto the RTIs at an earliest given time, and should be delivered to the destination and unloaded at an latest given time. These schedules are not definite though, and as long as the order is transported within the length of this schedule, its flow is feasible. For instance, an order might be held for a few hours at its origin destination before it is loaded and transported.

Besides management of RTIs, the other factor adding extra complexity to this problem is preserving product quality and delivering the desired freshness to its market. Perishable products lose their value over time, and due to geographic distances between production sites and final markets, preserving product quality is a determining factor in all transportation decisions. However, measuring and controlling the health of products are not straightforward. In the horticultural supply chain of the Netherlands, product freshness is approximated by a Time Temperature Sum (TTS) measure, and a maximum limit is enforced on TTS of orders.

Our problem is an extension to the classic Fixed-charge Capacitated Multicommodity Network Flow Problem (FCMNFP), where additional sets of constraints are added to these problems. Typically, demand in a FCMNFP problem is defined as commodities, and the only flow decision is defined on their routing throughout the network. However, we have the following types of decisions:

1. small *laden* flow decisions for loaded small RTIs that are transporting the products,
2. small *assign* flow decisions for empty small RTIs that are repositioned from a location with surplus of RTIs, to the origin locations of orders to be assigned to their transport,
3. small *repos* flow decisions for empty small RTIs that are repositioned from the destination locations of orders back to the storage locations with RTI shortage.
4. *medi* usage decisions for each available, which is equal to 1, if a particular medium RTI is used to pickup and deliver the small RTIs flowed around the network, and is equal to 0, otherwise,

5. *big* usage decisions for each available, which is equal to 1, if a particular big RTI is used to pickup and deliver the medium RTIs flowed around the network, and is equal to 0, otherwise.

Decisions on small RTIs are different from the decisions on medium and big RTIs. Small RTIs are not unique (labeled) and have a nonnegative nature, while medium and big RTIs are labeled, and usage and routing of each of them is optimized in this planning problem. Therefore, their decision variable has a binary nature.

In order to transport full and empty RTIs, we have fleets of different transportation modes, available throughout the network. Each transport mode has its own schedules, which is assumed to be given, and its vehicles operate based on these schedules. For instance, trains depart every 6 hours from a hub location. We assume each mode to have a specific temperature, which for example shows the temperature inside a truck trailer, a train car, or a barge storage room. Vehicles of a transport mode are capacitated (here, based on RTIs), and there are limited number of vehicles available for each transport modes.

We introduced the main decisions as decisions on the flows and on transport modes. Besides transportation activities carried out by the available fleet, these decisions include the relevant handling or holding activities as well. Therefore, locations are also assumed to have their own temperature, which is included in the quality measure of loading, unloading, and holding of laden RTIs at these locations.

Finally, the objective function of our problem is to minimize total system costs. Costs of renting a truck or reserving space (capacity) on a freight train, costs of handling (loading, unloading, or transshipping) products or RTIs, costs of storing them, administrative costs of these operations, etc., are examples of operational costs in such a transportation system.

In the following section, we give an overview on related literature.

3 Related Literature

In our planning problem, we have perishable products. The literature of planning transportation of such products is identified by extra preservation constraints or penalty costs. There is a vast literature on distribution planning of perishable products where they are transported from (or to) a central hub, usually a warehouse or a distribution center, to (or from) the final shops or customers. However, the literature of which explicitly studies long-haul transportation is very limited. Long-haul transportation is the one among hubs around the network, which are globally dispersed and are connected by direct non-truck or a sequence of multimodal options. Reis and Leal [2015] propose a MIP model for a soybean shipping chain planning problem where choice of transportation mode is included in the model besides decisions for annual crop purchase. Since their real-world application deals with significant uncertainty related to crop production, they define several combinations of scenarios for this uncertainty and apply their MIP model to each scenario in order to give insights for their decision makers. and Bortolini et al. [2016] propose a tri-objective LP for tactical planning of a food distribution network considering operating cost, carbon footprint and delivery time goals. They apply their tool to a real-world distribution problem and show the trade-off between the operational costs and the carbon footprint.

Our research adds to the literature of long-haul transportation of perishable products by incorporating management of resources, here RTIs, which is missing in Reis and Leal [2015] and Bortolini et al. [2016]. Resources (or assets) in general can be RTIs, vehicles, crews, power units, engines, etc., and positioning, balancing, allocating, repositioning, and rotation of assets are the subject of asset management.

The general literature on heterogeneous asset management is vast. As we stated earlier, the entire multimodal transportation planning literature can be considered as a subdivision

of heterogeneous asset management literature. Baldacci et al. [2010], Andersen et al. [2009], and Topaloglu and Powell [2006] are some explicit examples. Baldacci et al. [2010] provide an overview of different variants of heterogeneous vehicle routing problems (VRPs), and discuss presented solution approaches in the literature. Andersen et al. [2009] study Service Network Design (SND) problems with management of multiple fleet, and provide models and formulations for them. Topaloglu and Powell [2006] models a heterogeneous fleet management problem and the substitution among them as a Approximate Dynamic Program (ADP).

Combined planning of different assets with their complex interactions is modeled via *layering* [Powell, 2003], therefore, our FCMNFP with multiple sizes of RTIs can be formulated as a multi-layer MIP. The closest study to this environment is Zhu et al. [2014]. They study a rail transportation problem where customer demand comes to the system as the number of cars they need to transport. Then, these cars are classified, rearranged into blocks and trains, which are then routed throughout the rail network. They model it as a three-layered Service Network Design (SND) problem and solve it by means of a hybrid metaheuristic combining slope scaling, enhanced by long-term memory-based perturbation strategies and ellipsoidal search method.

Our problem structure and modeling approach is different from Zhu et al. [2014]. For instance, in their study, a new layer was added to the network for each asset management job, while we do not increase the dimensions of our network in SteadieSeifi et al. [2014]. There are other differences but more in the details, such as not including loading hierarchy and nesting ratio in their problem. Our contribution is therefore to include all of these into our model and since it is a complex problem, design an algorithm to solve it.

In the following sections, we describe our modeling approach and the mathematical formulation.

4 A Three-RTI Mathematical Model

The physical transportation network is characterized by nodes $i \in \mathcal{N}$ representing the hub locations, and the arcs (i, j) representing different routes connecting these locations. Between each location pair, at least one transportation mode $m \in \{1, \dots, \mathcal{M}\}$ can operate.

Similar to SteadieSeifi et al. [2014], in order to include all activities such as handling and holding operations into the model, we transform the physical network into a *mode-space-time representation*. First, we divide the time horizon T (e.g. 48 hours) into a set of time periods $t = 1, \dots, T$ (e.g. an hour), and map the physical network in both time and space. Each node $v \in \mathcal{V}$ in our mode-space-time network represents a location $i \in \mathcal{N}$ at a time $t \in \{1, \dots, T\}$ period on a mode $m \in \{1, \dots, \mathcal{M} + 1\}$. Layers $m = 1, \dots, \mathcal{M}$ can accommodate all transport activities, whether with fixed timetables or flexible, but the extra layer of $m = \mathcal{M} + 1$, here called *holding mode* H , is added to enable modeling handling and holding activities. A feasible arc $a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V})$, represents four types of operations:

- (i) a *travel arc* for traveling between hub locations (i, j) , leaving at particular time t by a particular mode type $m_1 = m_2 = m$ (Figure 3a). Depending on its departure time t , a travel arc has a length of $r_{(i,j),t}^m$.
- (ii) a *loading/unloading arc* for loading RTIs to a particular mode m (or unloading from it) at a location $i = j$. A loading arc has a modal state of $(m_1, m_2) = (H, m)$ and an unloading arc has a modal state of $(m_1, m_2) = (m, H)$, and like travel arcs, it can have different length of $r_{i,t}^{(m_1,m_2)}$ depending on the location and time t .
- (iii) a *waiting arc* representing the stand-by state of a mode $m_1 = m_2 = m$ (e.g. for switching rail tracks at borders, or custom clearance) at a location $i = j$ at time t . The length of

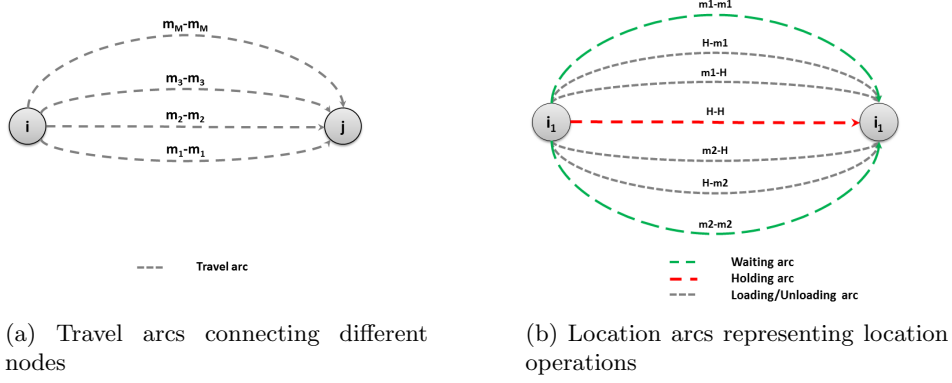


Figure 3: An illustration of different arcs in the model

waiting arc is one.

(iv) a *holding arc* for holding RTIs at a location $i = j$ at time t for one time period.

Since the last three activities occur at the locations, the arcs representing them are called *location arcs* (Figure 3b). Let $A_1 = \{a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}) \mid i \neq j, m_1 = m_2\}$ be the set of all feasible and given *travel arcs* in the mode-space-time network, and let $A_2 = \{a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}) \mid i = j, m_1 = H \parallel m_2 = H\}$ be the set of all feasible *location arcs* in the network. Similarly, let $A_m = \{a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}) \mid m_1 = H \parallel m_2 = H\}$ be the set of all feasible loading, unloading, traveling, and waiting arcs in the network related to mode m .

Each transport mode has its given schedules and time-dependent travel time $r_{(i,j),t}^m$, vehicle capacity cap^m , total F^m number of available vehicles, a temperature $l_{(i,j),t}^m$, fixed costs C^m representing the administrative costs associated with usage of its vehicles, and variable costs C_{RTI}^m for carrying RTIs (small laden, small empty, medi, and big). Compared to the problem with same-size RTIs, capacity of each mode is now defined on medium and big RTI scale. In general, if transportation modes have their own installed big containers (e.g. truck trailers, train cars, etc.), their capacity is defined on the number of medium RTIs they can accommodate. Otherwise, their capacity is based on big RTIs. Here, we assume to have reefer trucks, trains, and barges as examples of modes. Since, reefer trucks are not able to carry containers, their capacity is defined by the number of medium RTIs they can carry. Such an assumption makes our MIP a special case, and adds to the complexity of the problem, but it is not unrealistic. Our reason is to show how these capacity differences can be represented in a MIP.

Each location i has a temperature $l_{i,t}$. Moreover, each location i is assumed to have $S_{i,small}, S_{i,medi}, S_{i,big} \geq 0$ number of small, medium, and big RTIs available at the beginning of the planning horizon, and the number of RTIs at the end of horizon should be equal to the initial value. Of course, the location of small, medium, and big RTI storages are not necessarily the same, and inbound hubs are not obliged to keep similar number of them. Let $C_{i,small}^{(m_1,m_2)}, C_{i,medi}^{(m_1,m_2)}$, and $C_{i,big}^{(m_1,m_2)}$ be the general term for loading, unloading, and holding costs per small, medium, and big RTI per time period, which are enforced on the location arcs.

An order p is characterized by its origin $O(p)$, its destination $D(p)$, volume (based on small RTIs) w_p , an earliest pick up time of $PT(p)$, a latest delivery due date of $DT(p)$, and the maximum allowed TTS of $\mathcal{L}_p$. The number of needed medium and big RTIs is not indicated in the order. Their usage and routing are decisions that the model will determine based on all consolidation options and order requirements.

We introduced the decisions of our planning problem in Section 2, namely *laden*, *assign*, and *repos* decisions for small RTIs flows, and *medi* and *big* decisions determining the usage of

medium and big RTIs. For each order p , we define two variables $\tilde{x}$ to show how many small RTIs enter an arc $a_{(i,j),t,(m_1,m_2)}$, and $\hat{x}$ to show how many small RTIs exit the arc. Medium and big RTIs are labeled, meaning that we optimize the usage and routing of each one of them in our model. Assuming $g = 1, \dots, \mathcal{G}$ and $k = 1, \dots, \mathcal{K}$ number of medium and big RTIs available in the system, new sets of binary variables are introduced showing their *usage*, and similar to small RTI decisions, two sets of variables $\check{e}_g, \check{e}_k$ show a medium and a big RTI entering an arc, and two sets of variables $\hat{e}_g$ and $\hat{e}_k$ show a medium and a big RTI exiting the arc. When a medium and a big RTI are used to move small RTIs between locations i , and j , departing at time t , their usage variables are equal to 1.

Similarly, we have two sets of variables $\tilde{y}$ and $\hat{y}$ to show the number of vehicles respectively entering and exiting an arc. The auxiliary binary variables $\check{b}$ and $\hat{b}$ are also added to help calculating TTS.

Three categories of variables U are added to the model to keep track of the inbound and outbound small RTI flows at each location, and to connect the flows of loaded and empty small RTIs throughout the network. There are variables U_g and U_k to track the inbound and outbound medium and big RTI usage throughout the network.

Note that each medium RTI g has a capacity cap_g based on the number of small ones it can carry, and each big RTI k has a capacity cap_k equal to the number of medium RTIs it can accommodate.

In the remainder of this section, we formulate our FCMNFP model for the multimodal network flow problem with product quality preservation and management of RTIs with three sizes as:

$$\min \sum_{a_{(i,j),t,(m_1,m_2)} \in A_1} r_{(i,j),t}^m \left[\sum_{p=1}^{\mathcal{P}} C_{laden}^m \left(\tilde{x}_{p,(i,j),t,(m_1,m_2)}^{laden} \right) \right] \quad (1a)$$

$$+ \sum_{a_{(i,j),t,(m_1,m_2)} \in A_1} r_{(i,j),t}^m \left[\sum_{p=1}^{\mathcal{P}} C_{empty}^m \left(\tilde{x}_{p,(i,j),t,(m_1,m_2)}^{assign} + \tilde{x}_{p,(i,j),t,(m_1,m_2)}^{repos} \right) \right] \quad (1b)$$

$$+ \sum_{a_{(i,j),t,(m_1,m_2)} \in A_2} r_{i,t}^{(m_1,m_2)} \left[\sum_{p=1}^{\mathcal{P}} C_{i,small}^{(m_1,m_2)} \left(\tilde{x}_{p,(i,i),t,(m_1,m_2)}^{laden} \right) \right] \quad (1c)$$

$$+ \sum_{a_{(i,j),t,(m_1,m_2)} \in A_2} r_{i,t}^{(m_1,m_2)} \left[\sum_{p=1}^{\mathcal{P}} C_{i,small}^{(m_1,m_2)} \left(\tilde{x}_{p,(i,i),t,(m_1,m_2)}^{assign} + \tilde{x}_{p,(i,i),t,(m_1,m_2)}^{repos} \right) \right] \quad (1d)$$

$$+ \sum_{m=1}^{\mathcal{M}} \sum_{a_{(i,j),t,(m_1,m_2)} \in A_m} C^m \times \tilde{y}_{(i,j),t,(m_1,m_2)} \quad (1e)$$

$$+ \sum_{a_{(i,j),t,(m_1,m_2)} \in A_1} r_{(i,j),t}^m \left(C_{medi}^m \sum_{g=1}^{\mathcal{G}} \check{e}_{g,(i,j),t,(m,m)}^{medi} + C_{i,medi}^{(m_1,m_2)} \sum_{g=1}^{\mathcal{G}} \check{e}_{g,(i,i),t,(m_1,m_2)}^{medi} \right) \quad (1f)$$

$$+ \sum_{a_{(i,j),t,(m_1,m_2)} \in A_1} r_{(i,j),t}^m \left(C_{big}^m \sum_{k=1}^{\mathcal{K}} \check{e}_{k,(i,j),t,(m,m)}^{big} + C_{i,big}^{(m_1,m_2)} \sum_{k=1}^{\mathcal{K}} \check{e}_{k,(i,i),t,(m_1,m_2)}^{big} \right) \quad (1g)$$

The objective function is in the form of minimizing total system costs, where its terms respectively represent (1a) flow costs of the loaded small RTIs, (1b) flow costs of the assigned

and repositioned empty small RTIs, (1c) and (1d) locational costs of loaded, assigned, and repositioned small RTIs, and (1e) costs of using the modes. The terms (1f) and (1g) represent all costs related to medium and big RTIs.

Constraints (2)-(11) match pairs of entering and exiting variables, to ensure network arc connectivity.

$$\begin{aligned}
\check{x}_{p,(i,j),t,(m_1,m_2)}^{\text{cat}} &= \hat{x}_{p,(i,j),(t+r_{(i,j),t}^m),(m_1,m_2)}^{\text{cat}} & \forall cat \in \{\text{laden}, \text{empty}, \text{repos}\}, \\
& & a_{(i,j),t,(m_1,m_2)} \in A_1 \\
& & p = 1, \dots, \mathcal{P} \tag{2} \\
\check{x}_{p,(i,i),t,(m_1,m_2)}^{\text{cat}} &= \hat{x}_{p,(i,j),(t+r_{(i,j),t}^{(m_1,m_2)}),(m_1,m_2)}^{\text{cat}} & \forall cat \in \{\text{laden}, \text{empty}, \text{repos}\}, \\
& & a_{(i,j),t,(m_1,m_2)} \in A_2 \\
& & p = 1, \dots, \mathcal{P} \tag{3} \\
\check{y}_{(i,j),t,(m_1,m_2)} &= \hat{y}_{(i,j),(t+r_{(i,j),t}^m),(m_1,m_2)} & \forall a_{(i,j),t,(m_1,m_2)} \in A_1 \\
& & \tag{4} \\
\check{y}_{(i,i),t,(m_1,m_2)} &= \hat{y}_{(i,j),(t+r_{(i,j),t}^{(m_1,m_2)}),(m_1,m_2)} & \forall a_{(i,j),t,(m_1,m_2)} \in A_2 \\
& & \tag{5} \\
\check{b}_{p,(i,j),t,(m_1,m_2)} &= \hat{b}_{p,(i,j),(t+r_{(i,j),t}^m),(m_1,m_2)} & \forall a_{(i,j),t,(m_1,m_2)} \in A_1 \\
& & p = 1, \dots, \mathcal{P} \tag{6} \\
\check{b}_{p,(i,i),t,(m_1,m_2)} &= \hat{b}_{p,(i,j),(t+r_{(i,j),t}^{(m_1,m_2)}),(m_1,m_2)} & \forall a_{(i,j),t,(m_1,m_2)} \in A_2 \\
& & p = 1, \dots, \mathcal{P} \tag{7} \\
\check{e}_{g,(i,j),t,(m,m)}^{\text{medi}} &= \hat{e}_{g,(i,j),t+r_{(i,j),t}^m,(m_1,m_2)}^{\text{medi}} & \forall a_{(i,j),t,(m_1,m_2)} \in A_1 \\
& & g = 1, \dots, \mathcal{G} \tag{8} \\
\check{e}_{g,(i,i),t,(m_1,m_2)}^{\text{medi}} &= \hat{e}_{g,(i,j),t+r_{(i,j),t}^{(m_1,m_2)},(m_1,m_2)}^{\text{medi}} & \forall a_{(i,j),t,(m_1,m_2)} \in A_2 \\
& & g = 1, \dots, \mathcal{G} \tag{9} \\
\check{e}_{k,(i,j),t,(m,m)}^{\text{big}} &= \hat{e}_{k,(i,j),t+r_{(i,j),t}^m,(m_1,m_2)}^{\text{big}} & \forall a_{(i,j),t,(m_1,m_2)} \in A_1 \\
& & k = 1, \dots, \mathcal{K} \tag{10} \\
\check{e}_{k,(i,i),t,(m_1,m_2)}^{\text{big}} &= \hat{e}_{k,(i,j),t+r_{(i,j),t}^{(m_1,m_2)},(m_1,m_2)}^{\text{big}} & \forall a_{(i,j),t,(m_1,m_2)} \in A_2 \\
& & k = 1, \dots, \mathcal{K} \tag{11}
\end{aligned}$$

Constraints (12) are flow conservation constraints. These constraints define variables U as the total net flow (loaded small RTIs, assigned empty and repositioned empty small RTIs) for each order at each location and time period.

$$\begin{aligned}
U_{pit}^{\text{cat}} &= \sum_{j \in \mathcal{V} - \{i\}} \sum_{t' > t} \sum_{m=1}^{\mathcal{M}+1} \check{x}_{p,(i,j),t,(m_1,m_2)}^{\text{cat}} \\
&\quad - \sum_{j \in \mathcal{V} - \{i\}} \sum_{t' < t} \sum_{m=1}^{\mathcal{M}+1} \hat{x}_{p,(j,i),t,(m_1,m_2)}^{\text{cat}} & \forall cat \in \{\text{laden}, \text{empty}, \text{repos}\}, \\
& & i \in \mathcal{V}, \\
& & t = 1, \dots, T \\
& & p = 1, \dots, \mathcal{P} \tag{12}
\end{aligned}$$

Constraint set (13) enforces the flow of loaded small RTIs (orders) between origin and destination locations. It enforces the outbound flow of an origin node to be w_p and the inbound flow of a destination node to be $-w_p$. Note that even though Constraints (13) are equality constraints, there is no obligation for an order to immediately be loaded and transported at its earliest pick up time. In a feasible solution, an order might be held for several time periods

before loading. Similarly, the delivery due date of an order is not definite.

$$U_{pit}^{\text{laden}} \begin{cases} = w_p & i = O(p), t = PT(p) \\ = -w_p & i = D(p), t = DT(p) \\ = 0 & o.w. \end{cases} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ p = 1, \dots, \mathcal{P} \end{matrix} \quad (13)$$

$$U_{pit}^{\text{assign}} \begin{cases} \leq S_{i,small} & S_{i,small} > 0, t = 0 \\ = -w_p & i = O(p), t = PT(p) \\ 0 & o.w. \end{cases} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ p = 1, \dots, \mathcal{P} \end{matrix} \quad (14)$$

$$U_{pit}^{\text{repos}} \begin{cases} = w_p & i = D(p), t = DT(p) \\ \geq -S_{i,small} & S_{i,small} > 0, t = T \\ 0 & o.w. \end{cases} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ p = 1, \dots, \mathcal{P} \end{matrix} \quad (15)$$

Constraints (14) and (15), in a similar fashion enforce the flow of empty small RTIs between origin and destination locations. The origin locations of the assigned empty RTIs are the RTI storage locations with $S_{i,small} > 0$. Their destination locations are the locations where they are needed to be loaded and transport the products (O(p),PT(p)). On the other hand, the repositioned empty RTIs need to get back to the storage locations. Therefore, the locations with $S_{i,small} > 0$ are their destinations and their origin locations are the locations where the loaded RTIs are unloaded (D(p),DT(p)). Constraint (16) enforces the number of empty small RTIs assigned from a storage locations to be equal to the number of RTIs returned there.

$$\sum_{j \in \mathcal{V} - \{i\}} \sum_{m=1}^{\mathcal{M}+1} \sum_{p=1}^{\mathcal{P}} \tilde{x}_{p,(i,j),0,(m_1,m_2)}^{\text{assign}} = \sum_{j \in \mathcal{V} - \{i\}} \sum_{m=1}^{\mathcal{M}+1} \sum_{p=1}^{\mathcal{P}} \hat{x}_{p,(j,i),T,(m_1,m_2)}^{\text{repos}} \quad \forall i \in \mathcal{V} : S_i > 0 \quad (16)$$

Constraints (17) and (18) are equivalent conservation constraints for using medium and big RTIs, and constraint sets (19) and (20) enforces that at each medium and big RTI storage location, a medium or a big RTI can be at most used once, and for the rest of network, if a medium or a big RTI enters a node, it should also leave it.

$$U_{git} = \sum_{j \in \mathcal{V} - \{i\}} \sum_{t' > t} \sum_{m=1}^{\mathcal{M}+1} \tilde{e}_{g,(i,j),t,(m_1,m_2)}^{\text{medi}} - \sum_{j \in \mathcal{V} - \{i\}} \sum_{t' < t} \sum_{m=1}^{\mathcal{M}+1} \hat{e}_{g,(j,i),t,(m_1,m_2)}^{\text{medi}} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ g = 1, \dots, \mathcal{G} \end{matrix} \quad (17)$$

$$U_{kit} = \sum_{j \in \mathcal{V} - \{i\}} \sum_{t' > t} \sum_{m=1}^{\mathcal{M}+1} \tilde{e}_{k,(i,j),t,(m_1,m_2)}^{\text{big}} - \sum_{j \in \mathcal{V} - \{i\}} \sum_{t' < t} \sum_{m=1}^{\mathcal{M}+1} \hat{e}_{k,(j,i),t,(m_1,m_2)}^{\text{big}} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ k = 1, \dots, \mathcal{K} \end{matrix} \quad (18)$$

$$U_{git} = \begin{cases} \leq 1 & S_{i,medi} > 0, t = 0 \\ \leq 1 & S_{i,medi} > 0, t = T \\ = 0 & o.w. \end{cases} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ g = 1, \dots, \mathcal{G} \end{matrix} \quad (19)$$

$$U_{kit} = \begin{cases} \leq 1 & S_{i,big} > 0, t = 0 \\ \leq 1 & S_{i,big} > 0, t = T \\ = 0 & o.w. \end{cases} \quad \begin{matrix} \forall i \in \mathcal{V}, \\ t = 1, \dots, T \\ k = 1, \dots, \mathcal{K} \end{matrix} \quad (20)$$

Constraints (21) and (22) are logical constraints which are used to calculate the TTS of orders. Note that M is the classic “big M ”. Based on these constraints then, if there is no flow

of products on a specific arc ($x = 0$), that arc will not be included in the constraint on TTS ($b = 0$).

$$\tilde{x}_{p,(i,j),t,(m_1,m_2)}^{\text{laden}} \geq \check{b}_{p,(i,j),t,(m_1,m_2)} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \quad p = 1, \dots, \mathcal{P} \quad (21)$$

$$\tilde{x}_{p,(i,j),t,(m_1,m_2)}^{\text{laden}} \leq M\check{b}_{p,(i,j),t,(m_1,m_2)} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \quad p = 1, \dots, \mathcal{P} \quad (22)$$

Based on Constraints (21) and (22), Constraint (23) states that for each order, the total *time* $\times$ *temperature* of moving and handling an order must be less than or equal to the total required TTS of that order.

$$\begin{aligned} & \sum_{a_{(i,j),t,(m_1,m_2)} \in A_1} l_{(i,j),t}^m \times r_{(i,j),t}^m \times \check{b}_{p,(i,j),t,(m,m)} \\ & + \sum_{a_{(i,j),t,(m_1,m_2)} \in A_2} l_{i,t}^{(m_1,m_2)} \times r_{i,t}^{(m_1,m_2)} \times \check{b}_{p,(i,i),t,(m_1,m_2)} \leq \mathcal{L}_p \quad \forall p = 1, \dots, \mathcal{P} \end{aligned} \quad (23)$$

In this work, there are capacity restrictions enforced on medium RTIs, big RTIs, and on transportation modes, represented by Constraints (24)-(33). In order to incorporate nesting ratio into capacity computations, first, Constraints (24), (25), and (26) define the total number of small RTIs, medium RTIs, and big RTIs between i and j at time t . These constraints are later used to find the proportion of medium RTIs, big RTIs that are empty and are nested in a bigger RTI or vehicle.

$$\begin{aligned} \Pi_{(i,j),t,(m,m)}^{\text{small}} &= \sum_{p=1}^{\mathcal{P}} \tilde{x}_{p,(i,j),t,(m,m)}^{\text{laden}} + \beta^{\text{small}} \sum_{p=1}^{\mathcal{P}} \tilde{x}_{p,(i,j),t,(m,m)}^{\text{assign}} \\ & + \beta^{\text{small}} \sum_{p=1}^{\mathcal{P}} \tilde{x}_{p,(i,j),t,(m,m)}^{\text{repos}} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m \end{aligned} \quad (24)$$

$$\Pi_{(i,j),t,(m,m)}^{\text{medi}} = \sum_{g=1}^{\mathcal{G}} \check{e}_{g,(i,j),t,(m,m)}^{\text{medi}} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m \quad (25)$$

$$\Pi_{(i,j),t,(m,m)}^{\text{big}} = \sum_{k=1}^{\mathcal{K}} \check{e}_{k,(i,j),t,(m,m)}^{\text{big}} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m \quad (26)$$

Constraints (27) and (28) are weak and strong capacity constraints on the medium RTIs respectively. The total number of small RTIs (laden or empty) moved between i and j at time t should be less than or equal to the total capacity of medium RTIs transporting them.

$$\Pi_{(i,j),t,(m,m)}^{\text{small}} \leq \text{cap}^{\text{medi}} \times \Pi_{(i,j),t,(m,m)}^{\text{medi}} \quad a_{(i,j),t,(m_1,m_2)} \in A_m \quad (27)$$

$$\begin{aligned} \tilde{x}_{p,(i,j),t,(m,m)}^{\text{cat}} &\leq \text{cap}^{\text{medi}} \times \Pi_{(i,j),t,(m,m)}^{\text{medi}} \quad \forall \text{cat} \in \{\text{laden}, \text{empty}, \text{repos}\}, \\ &a_{(i,j),t,(m_1,m_2)} \in A_m \end{aligned} \quad (28)$$

Likewise, constraints (29) and (30) are weak and strong capacity limits on big RTIs. Using ceiling function, constraint (29) enforces the total loaded and empty medium RTIs to be less

than the capacity of the big RTIs carrying them.

$$\left[\Pi_{(i,j),t,(m,m)}^{small} / cap^{medi} \right] + \beta^{medi} \left(\Pi_{(i,j),t,(m,m)}^{medi} - \left[\Pi_{(i,j),t,(m,m)}^{small} / cap^{medi} \right] \right) \leq cap^{big} \times \Pi_{(i,j),t,(m,m)}^{big} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m \quad (29)$$

$$\check{e}_{g,(i,j),t,(m,m)}^{medi} \leq cap^{big} \times \Pi_{(i,j),t,(m,m)}^{big} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m \quad (30)$$

Finally, constraints (31)-(33) enforce the capacity of transportation modes. The total number of big RTIs transported by each mode should not exceed the capacity of vehicles transporting them, if that mode is chosen. It was explained in the previous section that big RTIs are not allowed on reefer trucks (constraint (33)). It was assumed that big RTIs are transported by trains and barges.

$$\Pi_{(i,j),t,(m,m)}^{medi} \leq cap_m \times \check{y}_{(i,j),t,(m,m)} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m, m = \text{reefer truck} \quad (31)$$

$$\Pi_{(i,j),t,(m,m)}^{big} \leq cap_m \times \check{y}_{(i,j),t,(m,m)} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m, m \neq \text{reefer truck} \quad (32)$$

$$\sum_{k=1}^{\mathcal{K}} \check{e}_{k,(i,j),t,(m,m)}^{big} = 0 \quad \forall a_{(i,j),t,(m_1,m_2)} \in A_m, m = \text{reefer truck} \quad (33)$$

Let $A_{t,m} = \{a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}) \mid m_1 = m, m_2 = m, \tilde{t} \leq t, \hat{t} \geq t\}$ be the set of all arcs of mode $m \in \{1, \dots, \mathcal{M}\}$ crossing time period t . Constraint (34) then states that in each time period, the number of used vehicles of a mode type must be less than or equal to a maximum value F^m .

$$\sum_{a_{(i,j),t,(m_1,m_2)} \in A_{t,m}} \check{y}_{(i,j),t,(m_1,m_2)} \leq |F^m| \quad \begin{matrix} \forall t = 1, \dots, T, \\ m = 1, \dots, \mathcal{M} \end{matrix} \quad (34)$$

Finally, Constraints (35)-(39) define the nature of variables in this formulation.

$$\check{x}_{p,(i,j),t,(m_1,m_2)}^{cat}, \hat{x}_{p,(i,j),t,(m_1,m_2)}^{cat} \geq 0 \quad \begin{matrix} \forall cat \in \{\text{laden}, \text{empty}, \text{repos}\}, \\ a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \\ p = 1, \dots, \mathcal{P} \end{matrix} \quad (35)$$

$$\check{b}_{p,(i,j),t,(m_1,m_2)} \in \{0, 1\} \quad \begin{matrix} \forall a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \\ p = 1, \dots, \mathcal{P} \end{matrix} \quad (36)$$

$$\check{e}_{g,(i,j),t,(m_1,m_2)}^{medi}, \hat{e}_{g,(i,j),t,(m_1,m_2)}^{medi} \in \{0, 1\} \quad \begin{matrix} \forall a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \\ g = 1, \dots, \mathcal{G} \end{matrix} \quad (37)$$

$$\check{e}_{k,(i,j),t,(m_1,m_2)}^{big}, \hat{e}_{k,(i,j),t,(m_1,m_2)}^{big} \in \{0, 1\} \quad \begin{matrix} \forall a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \\ k = 1, \dots, \mathcal{K} \end{matrix} \quad (38)$$

$$\check{y}_{(i,j),t}^{(m_1,m_2)} \in \mathbb{N} \quad \forall a_{(i,j),t,(m_1,m_2)} \in A(\mathcal{V} \times \mathcal{V}), \quad (39)$$

In the next section, we describe the designed solution algorithm to solve this problem in details.

5 Solution Algorithm

Balakrishnan et al. [1997] have shown that the fixed-charge capacitated network design problem is NP-hard. Our problem incorporates additional product preservation and resource management decisions and constraints which add further complexity to such a problem. There are two main challenges in designing a solution algorithm for a problem with multiple sizes of RTIs:

1. The structure of a solution in this problem has several interconnected layers, each representing movements of a particular RTI category. If several small RTIs are moved for an order, we can have a path for that order showing the flow of these RTIs. In comparison, if a medium or big RTI is moved, we will have a path showing its route around the network. These paths have similar segments due to the fact that smaller ones are moved by the bigger ones, and bigger ones are moved by the transport modes. It is not straightforward to improve a solution of this problem by traditionally removing and reinserting a particular part of it. To put it simply, if we remove an arc from the path of a medium RTI, several smaller RTI flows, a big RTI path, as well as related fleets are removed, leaving the other paths in different layers damaged. Therefore, an algorithm should repair not only the medium RTI paths, but should also repair all the other damaged parts. This intertwined solution structure makes designing a metaheuristic algorithm and its possible neighborhoods extremely hard. Here, we propose an algorithm that segregates the solution structure and its layers.
2. Secondly, synchronization of operations at all layers of the solution, makes scheduling calculations inflexible and difficult to handle. In order to avoid complicated scheduling, we work with *scheduling intervals*. Therefore, a feasible solution is one where all small RTI flows, medium RTIs, and big RTIs using a particular transport mode, have a *common time interval*, even though their individual intervals are not necessarily equal. We explain this scheduling tactic later in this section.

Our proposed algorithm is a multi-stage constructive algorithm, where it starts with empty repositioning and flows of small RTIs, then generates routes for medium and big RTIs, and finally arranges the fleet set. Figure 4 gives an overall view on the designed solution algorithm.

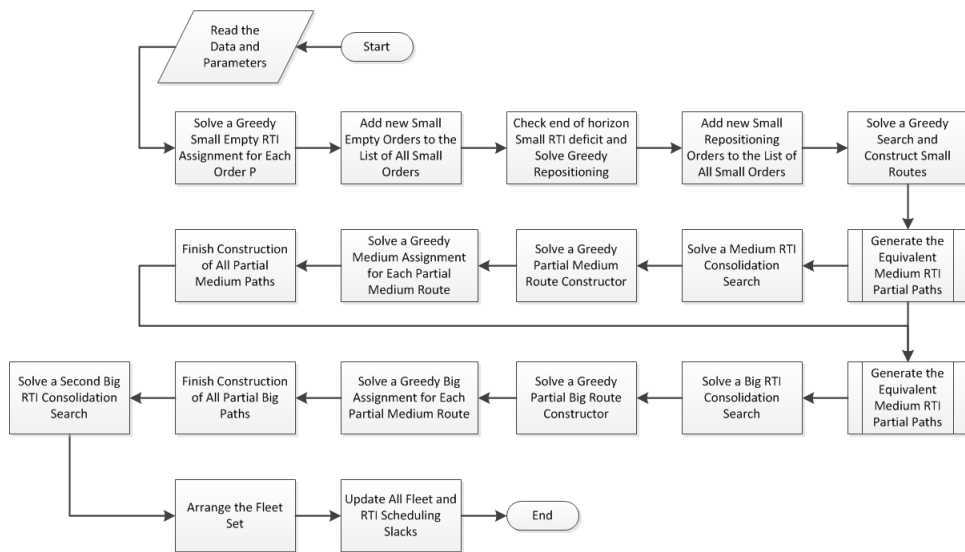


Figure 4: The algorithm of the Multi-stage Greedy Heuristic

A solution z consists of three sets of *S-routes*, *M-routes*, and *B-routes*, auxiliary sets of *M-partial paths* and *B-partial paths*, and a set of *fleet*. A S-route R_p represents the scheduled flow of (laden or empty) RTIs for an order p . In other words, an R_p is the set of all $\tilde{x}_p^{\text{laden}}$ (or $\tilde{x}_p^{\text{assign}}$, or $\tilde{x}_p^{\text{repos}}$) with positive values. Likewise, a M-route R_g and a B-route R_k represent the scheduled movements of medium and big RTIs respectively. These routes show the sequence of locations that small, medium, and big RTIs pass through, as well as the departing schedules of the modes transporting them among these locations. Note that in order to generalize the solution structure, and have a simpler and more systematic solution space search, we add *ghost big RTIs* in the algorithm to carry the medium RTIs transported by reefer trucks. These ghost RTIs have no capacity, no costs, and no storages, and therefore, they do not need assignment and repositioning either. As we explained in Section 2, reefer trucks are not able to carry big RTIs, but if this assumption changes, our ghost RTIs can easily be replaced by real ones.

In order to take synchronization of these connected routes into account, for each location that is visited on these routes, we define two scheduling intervals on departing from the location, and arriving to that location. These two intervals are defined by the following four values: *earliest departure time (ED)*, *earliest arrival time (EA)*, *latest departure time (LD)*, and *latest arrival time (LA)*. ED and LD show the earliest and the latest possible departing time from a location, and EA and LA show the earliest and the latest feasible arrival to a location on a route. A feasible solution is then a solution that for all arcs in it, the related S-route, M-route, B-route, and fleet traveling that arc, have scheduling overlaps to ensure their timing feasibility and synchronization.

The $f_{(i,j),t}^m$ in the set of fleet represents the mode m used for the arc (i, j) departing at time t . The set of fleet shows the number of vehicles used ($\tilde{y}$ with positive values) for each space-time-mode arc. As a result, the solution space is the set of all possible routes and their related fleet arrangements.

First Stage

At the first stage of our algorithm, we solve an assignment problem with a set of *supply locations* with empty small RTI surplus, and a set of *demand location* with empty RTI shortage or need, twice. In the first assignment problem, supply locations are the RTI storage locations with available number of RTIs $S_{i,small} > 0$, and the demand locations are the origin points $O(p)$ with demand w_p . In the second though, supply locations are the destination point $D(p)$ with supply w_p and the demand points are the storage locations with RTI deficit $S_{i,small} < 0$. This assignment problem is an extension to the classic transportation problem which can be solved via a state-of-the-art solver in a polynomial time.

The result of solving these assignment problems is *empty orders*. These empty orders define how many empty small RTIs and between which locations and at what time should be transported. We add them to the set of all orders, and then we build the S-routes.

In constructing the S-routes, we start with orders that have tighter time windows. Such orders have less flexible time intervals, and they are less likely to find cheaper but slower transportation. The greedy S-route constructor is a *cheapest path algorithm* which looks for the cheapest possible path for an order.

While constructing S-routes with its arcs, we generate partial paths for medium RTIs and big RTIs as well. The partial path sets keep track of needed medium and big RTIs that are needed for the S-routes. We use these partial paths in the next stages of the algorithm to construct M-routes and B-routes.

Second and Third Stages

At the second stage, we first go through all generated M-partial paths and try to find consolidation options, where the loads of two medium RTIs can be combined into one, and the excess medium RTI can be removed. It is important to note that this procedure is done while ensuring scheduling and synchronization feasibility. In this regard, we search for combinations that have cheaper costs than the original paths. If we find such a consolidation option, we then update schedules of all the related S-Routes.

After that, we call a greedy M-partial path constructor, in order to assemble the collection of M-partial paths together and generate longer partial paths. This greedy path constructor is a *cheapest insertion* algorithm, where starting from earliest M-partial paths, tries to find the cheapest feasible assembly to other M-partial paths. Each new and longer path now represents the sequence of locations visited by one particular medium RTI, transporting the small RTIs in it. We later use these longer paths to construct M-routes.

Afterwards, we employ a designated assignment heuristic to check whether the head and the tail locations of these medium partial paths are their storage locations or not. If not, the heuristic searches for the cheapest assignment and reposition of a medium RTI for that path, and then completes the M-routes, so that their starting and ending locations are their storage locations.

Similar to medium RTIs, in the third stage, we execute a consolidation search, a partial path assembly, and an assignment heuristic for completing the B-routes. In this stage also, if it is needed, we update the schedules of all related M-routes and S-routes.

The third stage has an additional consolidation search, where starting from the B-routes with lowest number of medium RTIs on them, we look for options to consolidate these medium RTIs once again, to see whether we can remove the B-route and its related big RTI.

By the end of second and third stages, all partial paths have been inserted into M-routes and B-routes, so the auxiliary path sets will become empty.

Fourth Stages

The final stage is a fast procedure to arrange the fleet and schedule their exact depart times, while monitoring all time interval feasibilities. Again, if necessary, we update schedules of all related routes.

In the next section, We present the computational results of solving this problem and the performance of our multi-stage greedy heuristic.

6 Numerical Experiments

In this section, the proposed formulation presented in Section 4 is verified on two groups of instances with 7 and 11 locations, three transportation modes of reefer trucks, train, and barge, and three sizes of RTIs. All instances are solved on a 2.4 GHz CPU with 16.00 GB RAM, and Gurobi solver 6.5.2 is used as the MIP solver. The instances are run for a maximum time of 10 hours.

6.1 Instance Sets

The instance sets we use, were inspired by transportation of horticultural products on the Trans-European Transport Network (TEN-T) [Tosi, 2014, Verhoeven, 2014, Vlassak, 2014, Rosenboom, 2014].

Table 1: Transportation mode inputs

	$freq^m$	cap_m^m (1)	cap_m^m (2)	F^m	l_i	$speed^m$	C_{fix}^m	C_{hr}^m	C_{km}^m
Reefer Truck services	1	22	-	200	5	65	136.34	49.02	0.28
Train services	6	-	80	10	5	32.5	179.37	13.17	0.24
Barge services	2	-	32	10	5	18.52	118.04	3.95	0.06

1: based on medium RTIs

2: based on big RTIs

Table 2: Returnable Transport Item (RTI) inputs

	cap_g (1)	cap_k (2)	β	C_{hr}	C_H
Small	-	-	0.33	0.04	0.002
Medi	10	-	0.44	0.20	0.008
Big	-	22	1.00	0.29	0.010

1: based on medium RTIs

2: based on big RTIs

We have two sets of instance with 7 and 11 locations. The first group only includes only the hubs in the Netherlands, the second group includes the hubs in the Netherlands, Belgium and Luxemburg. Two of the location (Aalsmeer and Naaldwijk) are inbound hubs, and the rest are outbound hubs. Inbound hubs are locations where the products come from all around the world to get sold, sorted, and consolidated for the shipments. The outbound hubs on the other hand are locations that the shipments are divided and packaged for the last-mile distribution.

Regarding the transportation modes, there are three classes of reefer truck, train, and barge transportation. Table 1 gives the parameter setting for the modes. Table 2 too gives the parameter setting on the RTIs ¹, ².

Assuming $dist_{(i,j)}$ to be the distance between locations i and j , the cost of using a mode C^m is calculated via $C^m = C_{fix}^m + C_{KM}^m \times dist_{(i,j)}$. The variable cost of moving an empty RTI of a particular size is calculated as $C_{size}^m = C_{HR}^m / (cap^m \times util^m)$. If the RTI is loaded, its C_{size}^m is multiplied by a coefficient VAT, to show the higher cost of transporting laden RTIs compared to empty ones.

Without loss of generality, and by observing the practice, we assume that the empty RTIs in this network are stored at the inbound hubs. We argue that our model is general enough to accept any mapping of RTI storages. Small RTIs are assumed to be stored at Aalsmeer, while medium and big RTIs are stored at Naaldwijk. Later, we compare the results with the case where all types of RTIs are stored at Aalsmeer, to show the repositioning cost differences.

The instances are named as "nAmBrCoD" where value of A shows the number of locations, value of B is the number of mode types, value of C is the number of RTI sizes, and value of D stands for the number of orders. The instances with $m1$ only have reefer truck options, the instances with $m2$ have truck and train options, and the instances with $m3$ have all modes available.

6.2 Computational Strengthening

The number of decision variables in our problem goes very quickly beyond what can be solved to optimality with a state-of-the-art MIP solver. Keeping all the matrices can cost a huge amount of memory. To deal with huge matrices of the parameters and variables that are extremely sparse, and to decrease the memory consumption of Gurobi, we used the so-called "colt"³ library

¹<http://www.containerhomeplans.org/2015/07/how-much-does-it-cost-to-transport-a-shipping-container/>

²<http://www.approvedindex.co.uk/storage-containers/iso-containers>

³<http://acs.lbl.gov/software/colt/>

Table 3: Comparison of exact and greedy results

Exact (in 10 Hours)											Multi-stage Greedy Constructive Algorithm							
											No. of RTIs				No. of Vehicles			
n	m	r	o	Upper Bound	Gap (%)	Medi	Big	R.	T.	B.	Total System Cost	Comp. Time (sec.)	Medi	Big	R.	T.	B.	
7	1	3	5	5889	15.08	12	-	13	-	-	9310	1	12	-	31	-	-	
7	2	3	5	4589	13.56	12	3	6	7	-	6462	4	12	3	18	4	-	
7	3	3	5	N.S.	-	-	-	-	-	-	5006	23	12	7	4	2	12	
7	1	3	10	N.S.	-	-	-	-	-	-	23042	1	35	-	84	-	-	
7	2	3	10	N.S.	-	-	-	-	-	-	15585	8	35	7	44	14	-	
7	3	3	10	N.S.	-	-	-	-	-	-	9146	52	35	12	10	3	20	
7	1	3	20	N.S.	-	-	-	-	-	-	42043	1	64	-	150	-	-	
7	2	3	20	N.S.	-	-	-	-	-	-	27359	12	63	9	83	17	-	
7	3	3	20	N.S.	-	-	-	-	-	-	14432	78	63	13	20	3	21	
11	1	3	5	3821	44.15	7	-	11	-	-	6237	10	8	-	22	-	-	
11	2	3	5	3348	44.4	7	2	7	5	-	5467	131	8	2	17	3	-	
11	3	3	5	N.S.	-	-	-	-	-	-	4621	268	8	7	6	1	14	
11	1	3	10	7682	45.45	14	-	21	-	-	12579	12	16	-	45	-	-	
11	2	3	10	N.S.	-	-	-	-	-	-	11668	164	16	5	37	6	-	
11	3	3	10	N.S.	-	-	-	-	-	-	6917	458	16	11	6	4	18	
11	1	3	20	N.S.	-	-	-	-	-	-	26370	27	34	-	89	-	-	
11	2	3	20	N.S.	-	-	-	-	-	-	21158	231	33	7	64	10	-	
11	3	3	20	N.S.	-	-	-	-	-	-	12468	938	34	17	14	5	26	

Note: N.S. stands for “No Solution”, meaning that Gurobi is not able to find any solutions for that instance in 10 Hour time
Note: R. stands for reefer trucks, T. stands for trains, and B. stands for barges.

to replace the standard matrix format with a sparse one. Colt libraries are a set of JAVA libraries designed for efficient use of scientific and technical data structure and algorithms, such as in multi-dimensional arrays, Monte Carlo simulations, etc. In multi-dimensional matrices for instance, it helps structuring the data such that its memory consumption is minimized.

6.3 Initial Results

In this section, we compare the results of Gurobi solver with the ones from our proposed multi-stage algorithm. Tables 3 gives an overview of the results, including the obtained upper bounds within 10 hours, the optimality gap, and the multi-stage algorithm cost comparison for the instances that Gurobi was able to find a solution.

As shown in the table, Gurobi solver is not able at all to find the optimal solution of even the smallest instances within 10 hours. The computation effort depends not only the number of locations, but on the number of modes and orders. For instances with more than 10 orders, Gurobi runs out of memory before starting the branch-and-bound.

In an attempt, we added some time window and some capacity constraints to the model to enhance preprocessing. However, it did not help at all. Increasing preprocessing setting of Gurobi itself did not help either.

Comparing results of our proposed algorithm with the obtained upper bounds, it is clear that despite our algorithm being able to find a feasible solution in less than an hour, it is not able to find near optimal solutions. The clear difference is in the fixed costs and the fleet arrangement (Table 4), which shows that the small-towards-big-size approach of the multi-stage algorithm is causing it to employ too many vehicles, even though the number of medium and big RTIs are not much different. Table 4 gives an overview on the total cost distribution among different assets, its small RTI costs representing terms (1a)-(1d) in the objective function, medium RTI costs representing term (1f), big RTI costs showing term(1g), and finally fixed costs of using the modes in the term (1e) of the objective function. Table 4 shows that despite cheaper small RTI

Table 4: Comparison of exact and greedy detailed costs

Exact (10 Hours)								
n	m	RTIs	o	Small RTI Costs	Medi RTI Costs	Big RTI Costs	Fixed Costs	
7	1	3	5	1026	2069	0	2794	
7	2	3	5	654	788	91	3148	
7	3	3	5	-	-	-	-	
11	1	3	5	675	1092	0	2055	
11	2	3	5	541	572	41	2194	
11	3	3	5	-	-	-	-	
Multi-stage Greedy Algorithm								
n	m	RTIs	o	Small RTI Costs	Medi RTI Costs	Big RTI Costs	Fixed Costs	
7	1	3	5	1026	1962	0	6323	
7	2	3	5	776	1300	116	4271	
7	3	3	5	687	833	368	3117	
11	1	3	5	420	1213	0	4604	
11	2	3	5	354	909	100	4105	
11	3	3	5	248	477	342	3554	

Table 5: Comparison of results for repositioning location cases

Different Locations											Similar Locations										
				No. of RTIs			No. of Vehicles								No. of RTIs			No. of Vehicles			
n	m	RTIs		Upper Bound	Gap (%)	Medi	Big	R.	T.	B.	Upper Bound	Gap (%)	Medi	Big	R.	T.	B.				
7	1	3	5	5889	15.08	12	-	13	-	-	5072	7.63	12	-	10	-	-				
7	2	3	5	4589	13.56	12	3	6	7	-	4222	21.12	12	2	5	6	-				
7	3	3	5	N.S.	-	-	-	-	-	-	5082	42.52	12	6	4	5	8				
7	1	3	10	N.S.	-	-	-	-	-	-	N.S.	-	-	-	-	-	-				
7	2	3	10	N.S.	-	-	-	-	-	-	7155	27.94	33	6	6	13	-				
7	3	3	10	N.S.	-	-	-	-	-	-	6934	36.99	35	8	5	4	13				
11	1	3	5	3821	44.15	7	-	13	-	-	2801	39.37	7	-	9	-	-				
11	2	3	5	3348	44.4	7	2	7	5	-	2671	43.79	7	2	5	4	-				
11	3	3	5	N.S.	-	-	-	-	-	-	N.S.	-	-	-	-	-	-				
11	1	3	10	7682	45.45	14	-	21	-	-	N.S.	-	-	-	-	-	-				
11	2	3	10	N.S.	-	-	-	-	-	-	5502	59.9	17	6	3	14	-				
11	3	3	10	N.S.	-	-	-	-	-	-	5566	63.05	16	8	3	7	10				

Note: R. stands for reefer trucks, T. stands for trains, and B. stands for barges.

costs, medium and big RTI routes are not cheap, and the fleet is expensive.

6.4 Similar RTI Storage Locations

So far, we assumed that the empty small RTIs are stored at Aalsmeer, while medium and big RTIs are stored at Naaldwijk. In this section, we compare the results to the case where all types of RTIs are stored at Aalsmeer. The reason is that Aalsmeer is the biggest inbound hub with processing logistics of the majority of imported and exported products. Table 5 and 6 show the results of comparing solutions for different and similar RTI locations. Looking at Table 5, Gurobi solver is able to find an upper bound for more instances within 10 hours. However, this difference is not significant. Moreover, it is clear that new solutions have less repositioning movements with less vehicles.

Table 6 gives an overview on medium and big repositioning costs for the few comparable instances. Comparing the results of Gurobi solver, storing all RTIs at Aalsmeer shows a significant reduction of repositioning costs for medium and big RTIs, and for instances with 11 locations, the cost decrease is even more than 90%.

Table 6 does not show the comparison of small RTI repositioning. The reason is that repositioning

Table 6: Comparison of "repositioning costs" of medium and big RTI for all cases

n	m	RTIs	Different Location		Similar Location	
			Medi RTIs	Big RTIs	Medi RTIs	Big RTIs
7	1	3 5	514	0	-22	-
7	2	3 5	351	28	-69	-49
7	3	3 5	-	-	-	-
7	1	3 10	-	-	-	-
7	2	3 10	-	-	-	-
7	3	3 10	-	-	-	-
11	1	3 5	314	0	-91	-
11	2	3 5	300	41	-90	-100
11	3	3 5	-	-	-	-
11	1	3 10	-	-	-	-
11	2	3 10	-	-	-	-
11	3	3 10	-	-	-	-

tioning small RTIs did not have any significant difference in all comparable instances. However, since there are only a few comparable instances at hand, we could not draw further conclusions on small RTI repositioning. Furthermore, we did not test the separation of medium RTIs storage locations from the big ones, but our model can still provide solutions for any different medium RTI storage mapping. In addition, we argue that storing medium RTIs in locations with the most product flow traffic would results in a repositioning cost decrease, similar to Table 6.

In the next section, we provide some concluding remarks.

7 Concluding Remarks

In this working paper, we extend the multimodal transportation problem of perishable products with repositioning of same-size RTIs, to include three RTI (small, medium, and big) sizes, their loading hierarchy, and the empty nesting (folding) phenomenon. In practice, perishable products are loaded in small RTIs, then these RTIs are loaded onto the medium and big sizes, in order to be transported via a multimodal network. The empty RTIs to be repositioned are either folded, or nested onto each other, in order to save space and utilize less mode capacity.

In this regard, we introduced new variables and constraints to the classic FCMNFP problems. This problem is NP-hard, and as results show, a state-of-the-art MIP solver is hardly able to find optimal solutions for even the smallest instances with 5 orders within 10 hours. Therefore, solving real-world sizes of such a planning problem to optimality is impossible. Then, we introduced a multi-stage constructive algorithm, where the solution structure is segregated into several layers based on the size of RTI, and via four stages, it routes and repositions small, medium and big RTIs to orders, and arranges the fleet.

Results show that our proposed algorithm alone is not powerful and sufficient to provide near optimal solutions, but it can be used as the first step to design more advanced heuristic or metaheuristic algorithms. The biggest weakness of our algorithm is in repositioning medium and big RTIs. Therefore, the first possible future work can be to improve the medium and big RTI assignment and repositioning modules. Moreover, the small-towards-big-size approach of this algorithm does not efficiently directs it towards the optimal solution, thus, another future extension could be to add further stages to the algorithm that take a big-towards-small approach and search for improvements in all layers of the solution.

In designing metaheuristic algorithms, the challenges are threefold: in structuring the solutions, in defining the search neighborhoods, and in designing the relevant destroy and repair operators. A solution of our problem has different intertwined layers of RTI flows and usage, and an interesting future work is to find a simple structure that can be used in the body of a metaheuristic algorithm. Assuming the solution structure to be similar to our segregated one,

the second interesting future work is to define simple and tractable search neighborhoods on all layers, and their relevant destroy and repair operators. Standard arc-based operators, especially on the medium RTI layer might make the metaheuristic algorithm prone to errors, inflexible and time consuming search, and getting stuck in local optimums. Path-based operators particularly on small RTI and big RTI layers are more manageable. Therefore, another future extension is to design these operators, and by using the solution of the multi-stage algorithm as an initial solution, design further iterative stages to improve it. This is an ongoing work.

References

- J. Andersen, T.G. Crainic, and M. Christiansen. Service network design with management and coordination of multiple fleets. *European Journal of Operational Research*, 193(2):377–389, 2009.
- A. Balakrishnan, T.L. Magnanti, and P. Mirchandani. Network design. In M. Dell’Amico, F. Maffoli, and S. Martello, editors, *Annotated Bibliographies in Combinatorial Optimization*, chapter 16, pages 311–334. 1997.
- R. Baldacci, M. Battarra, and D. Vigo. Routing a heterogeneous fleet of vehicles. In B. Golden, S. Raghavan, and E. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, chapter 1, pages 3–27. 2010.
- M. Bortolini, M. Faccio, E. Ferrari, M. Gamberi, and F. Pilati. Fresh food sustainable distribution: Cost, delivery time and carbon footprint three-objective optimization. *Journal of Food Engineering*, 174:56–67, 2016.
- W.B. Powell. Dynamic models of transportation operations. In A.G. de Kok and S.C. Graves, editors, *Supply Chain Management: Design, Coordination and Operation*, volume 11 of *Handbooks in Operations Research and Management Science*, pages 677–756. 2003.
- S.A. Reis and J.E. Leal. A deterministic mathematical model to support temporal and spatial decisions of the soybean supply chain. *Journal of Transport Geography*, 43:48–58, 2015.
- A. Rosenboom. Development of a decision support system for an orchestrator in the horticultural sector, 2014. Master Thesis, TUE School of Industrial Engineering.
- M. SteadieSeifi, N. Dellaert, and T. van Woensel. A multimodal network flow problem with product quality preservation, transshipment, and asset management. Technical Report 442, Technische Universiteit Eindhoven, 2014. A BETA working paper.
- H. Topaloglu and W.B. Powell. Dynamic-programming approximations for stochastic time-staged integer multicommodity-flow problems. *INFORMS Journal on Computing*, 18(1):31–42, 2006.
- M. Tosi. Allocation of VALS activities in a metro model network in the European floricultural market, 2014. Master Thesis, Wageningen Department of Social Sciences.
- S. Verhoeven. Hub location decisions in the metromodel, 2014. Master Thesis, TUE School of Industrial Engineering.
- T. Vlassak. Future-proof return logistics in the floriculture sector, 2014. Master Thesis, TUE School of Industrial Engineering.
- E. Zhu, T.G. Crainic, and M. Gendreau. Scheduled service network design for freight rail transportation. *Operations research*, 62(2):383–400, 2014.