# Combined Maritime Fleet Deployment and Inventory Management with Port Visit Flexibility in Roll-on Roll-off Shipping

**Abstract**

We consider a combined fleet deployment and inventory management problem in Roll-on Roll-off shipping. Along given trade routes there are ports with inventories that should be kept within their limits. Current planning practice is to visit all ports every time a trade route is serviced. We instead aim at determining the sailing routes of each voyage along the trade route, where some ports can be skipped on certain voyages. A novel mixed integer programming model is proposed and tested on realistic instances. The results indicate that substantial gains can be achieved from this more flexible way of planning.

*Keywords:* Fleet deployment, Maritime inventory routing, Roll-on Roll-off shipping

## 1. Introduction

Liner shipping is one of the major transportation modes in maritime transportation and has similarities with a bus service where the ships service a given set of voyages along trades according to a published schedule. Container ships are the most dominant ship type of liner shipping, and a comprehensive review on container ship routing and scheduling in liner shipping is provided by Meng et al. (2013) and more recently by Christiansen et al. (2019). Roll-on roll-off (Ro-Ro) shipping is another segment in liner shipping. In Ro-Ro shipping, a variety of rolling cargoes, such as cars and trucks, is transported. Even though not being as dominant as container shipping, Ro-Ro shipping is still an important segment with a world fleet of around 5000 ships with a total capacity of more than 24 million deadweight tons (ISL, 2017).

In this paper we study the combined *fleet deployment and inventory management problem* (FDIMP) emerging in Ro-Ro shipping. This important planning problem has usually been solved in two steps, where one first solves the fleet deployment problem and then handles the inventory management at a lower planning level given the plan for the fleet deployment.

The maritime *fleet deployment problem* can be described as optimally assigning available ships in the shipping company's fleet to given voyages along predefined trade routes over a planning horizon that typically stretches from a few months and up to a year. A trade route is a shipping route connecting ports in different geographical regions. Example illustrations of two trade routes are shown in Figure 1, where in Trade 1, cargo is loaded at given ports in Europe (Piraeus) and unloaded at given ports in North-America (New York), while in

Trade 2, cargo goes in the opposite direction and is loaded in North-America (Miami and Baltimore) and unloaded in Europe (Gothenburg).

Even though the example in Figure 1, for illustrational purposes, only shows one loading port and one unloading port on Trade 1, the realistic trade routes always have at least two ports in each region. Each trade route must be serviced regularly, e.g. bi-weekly, so a number of voyages spread over the planning horizon are defined. Each voyage usually has a given time window when it must start to ensure that the voyages along the same trade route are spread over time, but also to provide some planning flexibility (increasing with increased width of the time windows). The fleet deployment plan provides information about each ship's sailing route, which consists of a sequence of voyages to be sailed, possibly with some ballast sailing between successive voyages (dashed lines in Figure 1).



Trade 1(Europe-America): 1(Piraeus),2(New York)
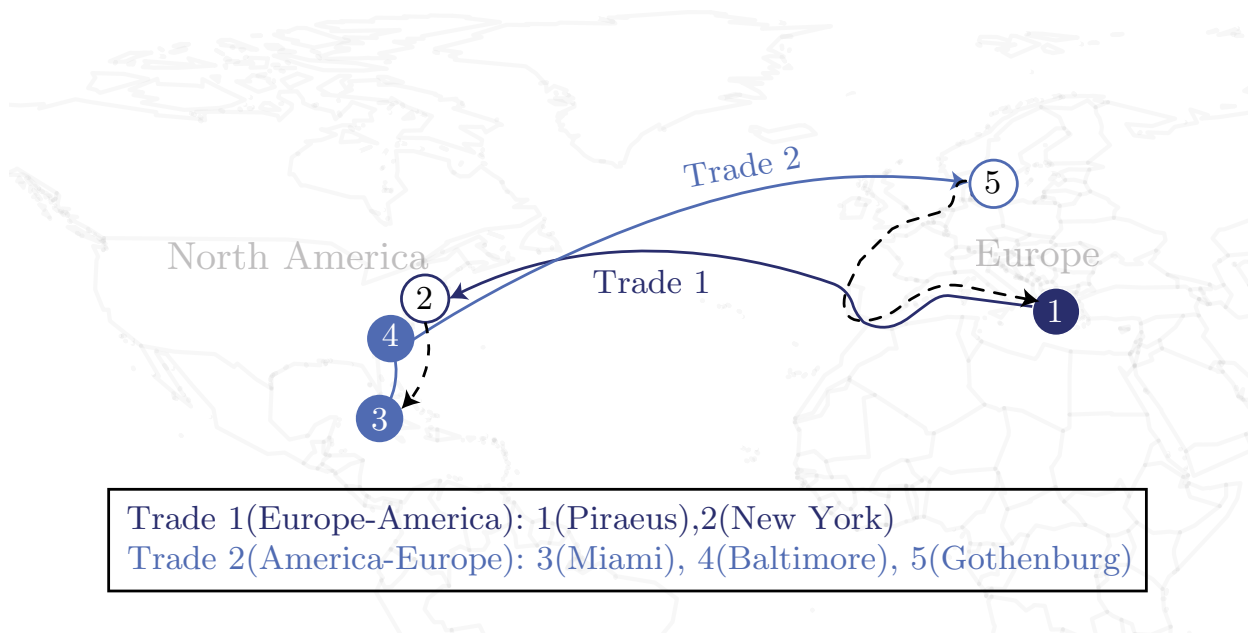Trade 2(America-Europe): 3(Miami), 4(Baltimore), 5(Gothenburg)

Figure 1: Trade 1: Loading port in Europe and unloading port in America. Trade 2: Loading ports in America and unloading port in Europe.

Most of the literature on fleet deployment in liner shipping is for container shipping, see for example Gelareh and Meng (2010), Liu et al. (2011), Wang and Meng (2012), and Ng (2015). However, the fleet deployment problem in container shipping separates itself from Ro-Ro shipping in two main aspects: 1) The container shipping lines are cyclic with each ship assigned to a rotation (i.e. a short cycle of trade routes serviced in sequence) for the entire planning horizon, while in Ro-Ro shipping each ship is assigned to a sequence of voyages from different trade routes. 2) In container shipping one usually considers deployment of ship types rather than individual ships, which might be a reasonable assumption in container shipping due to 1). However, this is a too crude approximation in Ro-Ro shipping since one cannot take into account the different initial positions of the ships at the beginning of the planning horizon.

There are also some studies on fleet deployment in Ro-Ro shipping. Fagerholt et al. (2009) present a somewhat simplified model that is solved using a multi-start local search heuristic, while Andersson et al. (2015) extend this model by also including speed decisions and propose a rolling horizon heuristic for solving it. Fischer et al. (2016) extend this further and handle the different cargo types and ship capacities more detailed and explicitly. They also propose alternative strategies for creating robust solutions and for handling disruptions, and test these through simulation.

The combined Ro-Ro fleet deployment and inventory management problem (FDIMP) has also recently been studied to some extent. In this context, inventory management is about deciding the quantities to ship between the ports along each voyage along the different trade routes, while keeping the inventory at each port within its given limit. Chandra et al. (2016) introduced the FDIMP in Ro-Ro shipping and proposed a time-discrete model for the combined problem. Later, Dong et al. (2017) proposed a time-continuous model for the same problem, and it was shown that their formulation is more efficient than the one from Chandra et al. (2016). It was also shown that significant gains can be achieved from integrating the fleet deployment and inventory management decisions.

One important assumption in the models proposed by Chandra et al. (2016) and Dong et al. (2017), is that even though inventory management is integrated in the fleet deployment planning, one still requires that all ports along the trade routes are visited on each voyage. Despite that this represents the current practice, it may impose an unnecessary restriction in Ro-Ro shipping. To utilize the integrated planning of fleet deployment and inventory management to save sailing and port costs, one might instead want to skip the port visits on some voyages along the trades, as long as the inventories are kept within its limits. Using the example from Figure 1: if the inventory in Baltimore can be kept within its limit for one month, then one might not necessarily need to visit that port along each voyage, even though the trade itself needs to be serviced bi-weekly.

Therefore, instead of assuming that all ports along a trade are visited every time the trade is serviced, which has been common to simplify planning, we extend the previous model for the FDIMP (Dong et al., 2017) by utilizing the inherent planning flexibility regarding port visits. In contrast to the previous model, we determine which ports to visit and which ports to skip along each voyage based on their inventory levels. This can be seen as taking some of the flexibility in maritime inventory routing in industrial shipping (e.g. Song and Furman (2013); Zhang et al. (2018); Rodrigues et al. (2019)) regarding deciding which ports to visit and how many products to transport, and combine this with operating on given trade routes as in liner shipping.

The purpose of this paper is therefore to introduce a novel mixed integer programming (MIP) model for this new and more flexible version of the FDIMP in Ro-Ro shipping. We use the model to solve a number of realistic but partly randomly generated test instances of different sizes and characteristics with regards to inventory limits and planning flexibility. Furthermore, we show that even though the inclusion of this additional planning flexibility makes the model significantly more complex and harder to solve, it also results in substantial financial savings.

Section 2 provides a definition of this new version of the FDIMP and describes the mod-

eling approach that we have chosen, while Section 3 presents the mathematical formulation. Section 4 describes the computational study, before we conclude in the final section.

## 2. Problem Definition, Notation and Modeling Approach

The FDIMP considers a planning problem for a Ro-Ro shipping company that operates a heterogeneous fleet of Ro-Ro ships, defined by the set $\mathcal{V}$. The ships may have different capacities, costs, speeds, fuel consumption and origins at the beginning of the planning horizon. Furthermore, the ships can have different times for when they become available, as they might need to finish the voyages they are currently sailing at the time when the planning is done.

The shipping company services a set of trade routes (or just trades in short), $\mathcal{I}$. Figure 2 shows the node representation of the two trades corresponding to those in Figure 1. In Figure 2, Trade 1 consists of one loading port, represented by node 1 (Piraeus), and one unloading port, 2 (New York). Trade 2 consists of two loading ports, nodes 3 and 4 (Miami and Baltimore), and one unloading port, 5 (Gothenburg). Without loss of generality, we assume that all ports along a trade are indexed in an increasing order in which they appear in the sequence geographically. It can be noted that a given port can appear in more than one trade, and possibly as a loading port in one trade and unloading port in another.
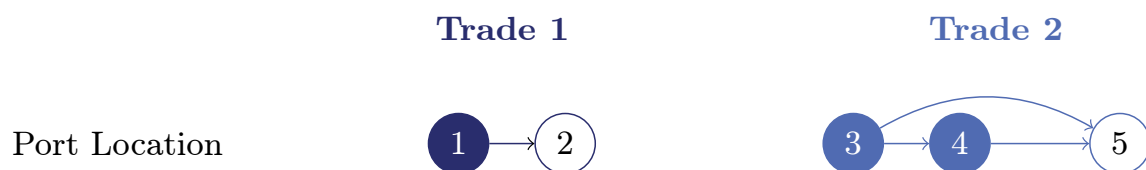


Figure 2: Node representation of the two trades from Figure 1.

Due to contractual requirements and demand, each trade must be serviced periodically to transport cargoes between ports in the loading and unloading regions. Therefore, for each trade $i$, a set of voyages $\mathcal{M}_i$ is defined. A ship deployed to voyage $m$ on trade $i$ visits ports along the trade, transporting cargo from the loading port(s) to the unloading port(s). When a ship completes a voyage on one trade, it can continue to service another voyage (possibly on the same trade route) or terminate its route. The ship often needs to reposition itself without load to be ready for the next voyage. Using the small example from Figures 1 and 2, a ship can first traverse a voyage on Trade 1, which ends in New York, and then continue to perform a voyage on Trade 2, which starts in Miami. Here, the ship has to sail in ballast (i.e. empty) between New York and Miami, as illustrated in Figure 1 with a dashed line.

Although all trades should be periodically serviced, one has some flexibility regarding when each voyage must commence. A time window defined for each voyage, given by an earliest start time, $\underline{T}_{im}$, and latest start time, $\bar{T}_{im}$, restricts within what time interval the operation of voyage $m$ for trade $i$ must begin.

4

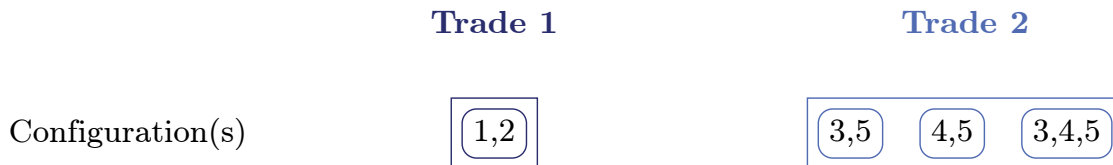|  | **Trade 1** | **Trade 2** |
|---|---|---|
| Configuration(s) | 1,2 | 3,5   4,5   3,4,5 |

Figure 3: Two trades with the possible configurations.

We define the set of all ports belonging to trade $i$ as $\mathcal{P}_i$. Some ports may be skipped during a voyage depending on production/demand and inventory limits. We therefore define a set of configurations, $\mathcal{C}_i$, for each trade $i$. A configuration $c$ of a trade $i$, shown as rounded rectangles (or nodes) in Figure 3 for our previous example, is defined as an ordered set of ports, $\mathcal{P}_{ic}^C$, that are chosen to be visited in that configuration. A cluster is a set of all possible configurations, shown as rectangles in Figure 3. In particular, the cluster of Trade 1 has only one configuration {1,2}, since one always has to visit at least one port in both the loading and unloading regions. The cluster of Trade 2 has three possible configurations {3,4}, {4,5}, {3,4,5}. This also means that we assume it is not allowed to visit the ports in a sequence that deviates from the ports' natural ordering along the trade as it would increase the sailing distance, e.g. configuration {4,3,5} is not allowed.

The sequencing of the ports along configuration $c$ of trade $i$ is given by $\mathcal{R}_{ic}^C$. For example, if $\mathcal{P}_{ic}^C = \{3, 4, 5\}$ (as for Trade 2 in Figure 3), then $\mathcal{R}_{ic}^C = \{1, 2, 3\}$, such that $r = 1$ corresponds to port number $p_1 = 3$, $r = 2$ to port number $p_2 = 4$, and so on. $\hat{R}_{ic}$ is the number of elements (i.e. ports) in configuration $c$ for trade $i$, such that $\hat{R}_{21} = 2$, $\hat{R}_{22} = 2$, and $\hat{R}_{23} = 3$, if we refer to our example in Figure 3. We also let $\mathcal{R}_i$ define the set of sequence numbers for the full trade $i$, such that $\mathcal{R}_1 = \{1, 2\}$ and $\mathcal{R}_2 = \{1, 2, 3\}$.

For each ship $v$, a ship route consists of a geographical path specifying the trades and voyages performed, a schedule specifying when each voyage begins, a loading plan specifying the amount of product loaded at each loading port and an unloading plan specifying the amount of product discharged at each unloading port. Figure 4 shows example routes and schedules for ships 1 and 2, with no port choice flexibility, as in (Dong et al., 2017), and with port choice flexibility, respectively.

In Figure 4, a node represents a voyage $m$ along a trade $i$ with configuration $c$. The filled nodes in the figure correspond to the chosen configuration for each voyage, and the unfilled nodes represent the configurations that were not chosen. Exactly one node, $(i, m, c)$, must be chosen inside each cluster. A set of nodes and arcs that can be serviced by ship $v$ is given by $(i, m, c) \in \mathcal{N}_v$ and $((i, m, c), (j, n, d)) \in \mathcal{A}_v$, respectively. In the upper part of Figure 4, both ships visit all ports on a trade for all voyages, while in the bottom part, ship 1 first services Voyage 1 on Trade 1 with configuration {1,2}, before it traverses Voyage 1 on Trade 2 with configuration {4,5}. This means that Port 3 is skipped in this service of Voyage 1 on the trade. The ship then services Voyage 3 on Trade 1 with configuration {1,2}, before terminating its route at the end of the planning horizon, $T$.
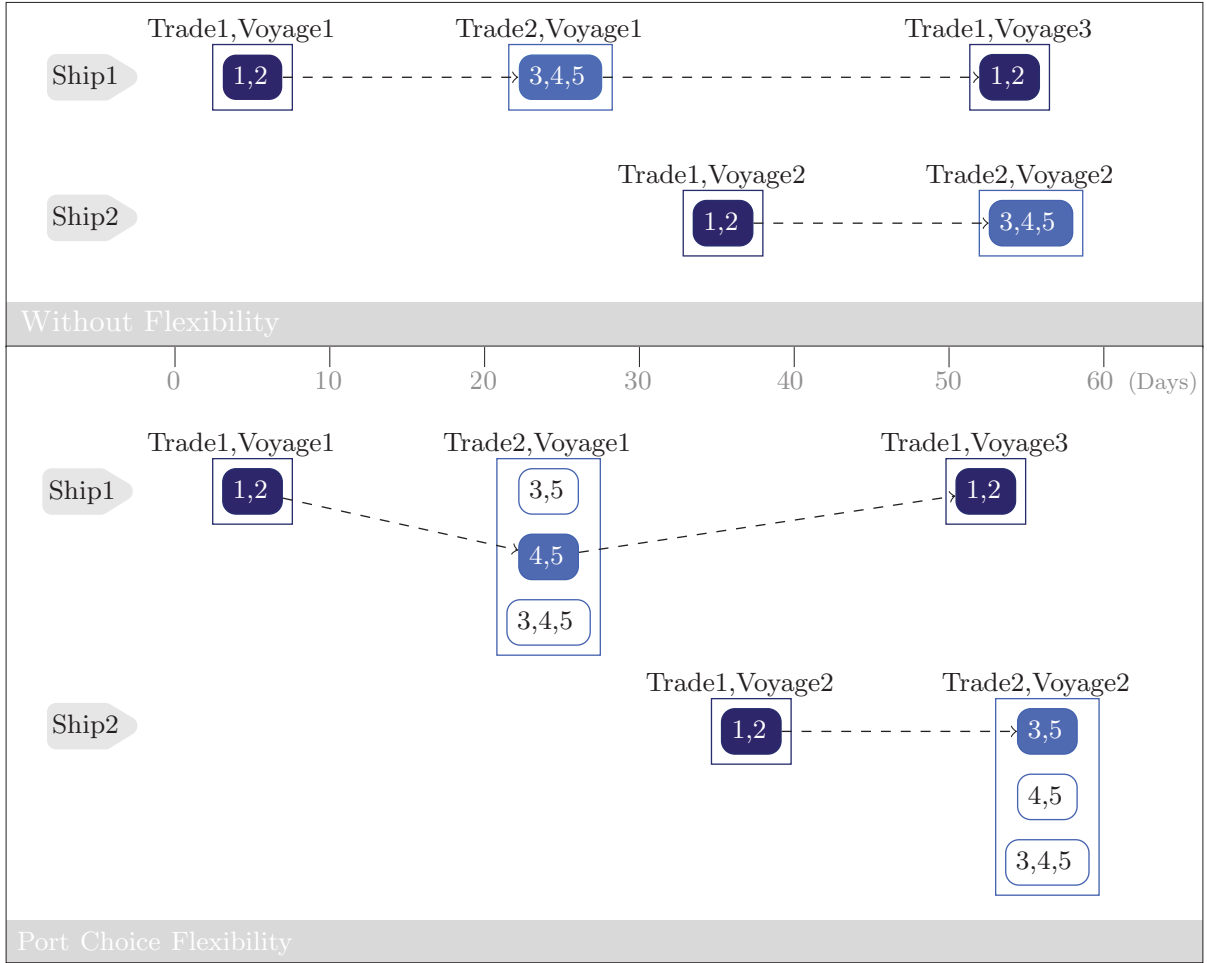
5

Figure 4: Example of routes for two ships, one with no port choice flexibility (as in Dong et al. (2017)), and the other with port choice flexibility.

There is a given sailing time from the initial position of ship $v$ to the start of any voyage along trade $i$, defined by $T^O_{icv}$. It should be noted that this time depends on the configuration, since different configurations might have different starting ports. Each ship is assumed to sail at a fixed speed during the planning horizon, so that the sailing time between any two ports are determined based on the distance. It is assumed that a ship can wait before the start of a voyage, while it cannot do so at ports along the voyage. Thus, each voyage has an estimated duration depending on which ship and configuration that are chosen. This duration includes the sailing time between all visited ports and the time spent in these ports. We assume that the time spent in each port is independent of the amount of cargo loaded/unloaded. The motivation behind this is threefold: 1) Much of the time spent in port consists of fixed terms related to piloting and berthing; 2) The planning horizon for the FDIMP is several months and the sailing times for the trades can be several weeks, while the

time spent in each port is typically around one day. Therefore, modelling the time spent in each port very detailed does not add much from a practical point of view, as long as there is still uncertainty in sailing times due to different weather conditions that more than outweigh this; 3) We only need to estimate the time for performing each voyage. In most practical situations it is a reasonable assumption that the ship is always (close to) fully loaded when starting a voyage. This means that the total amount loaded and unloaded along a voyage is close to constant, which again mean that the time spent on loading/unloading in all ports along a voyage will also be close to constant.

For a voyage with configuration $c$ of trade $i$, the sailing time of ship $v$ from the start position of the voyage to port $p$ is $T^R_{icpv}$. The time spent in port $p$ in trade $i$ is assumed to be only dependent on the port and is denoted by $T^P_{ip}$. The service time of a voyage with configuration $c$ of trade $i$ plus the ballast sailing time from the final port of configuration $c$ of trade $i$ to the starting port of configuration $d$ of trade $j$ with ship $v$ is given by $T_{icjdv}$.

Although some of the ports along a trade could be skipped during a given voyage, it is assumed that each port has its associated arrival time during each voyage. A skipped port will be assigned a pseudo arrival time which enables the possibility of keeping track of the inventory level at the corresponding pseudo arrival time. More specifically, if a port is visited during a voyage, the time arriving at that port is determined by the start time of that voyage plus the sailing time from the start position of the chosen configuration to that port and the operation times at ports already visited along the voyage. Otherwise, if a port is skipped, then we also assign a pseudo arrival time for that port (the pseudo departure time equals pseudo arrival time as the operation time at a skipped port is 0). If the skipped port is before the first port of the chosen configuration (e.g. Port 3 in Voyage 1 of Trade 2 with configuration {4,5} in Figure 4), the pseudo arrival time for that port is the same as the start time of the voyage (i.e. the start time of Voyage 1). If the skipped port is located after the last port of the chosen configuration, the pseudo arrival time for that port is the same as the end time of the voyage. If the skipped port is between the first port and the last port of the chosen configuration (e.g. Port 4 in Trade 2 with configuration {3,5} in Figure 4), the pseudo arrival time for that port is the same as the end of service time at a visited port which is visited just before "visiting" port $p$ (i.e. the end of service time at port 3).

The parameter $C^O_{icv}$ is the sailing cost for ship $v$ between its origin and the first port of configuration $c$ of any voyage along trade $i$ for ship $v$. Performing a voyage of trade $i$ incurs costs, $C^R_{icv}$, depending on the ship type $v$ and the configuration $c$ chosen. The cost of performing a voyage along configuration $c$ of trade $i$ plus the cost of ballast sailing to the first port of configuration $d$ of trade $j$ from the end of configuration $c$ of trade $i$ with ship $v$ is $C^B_{icjdv}$.

A single type of product is considered. The set of ports of trade $i$, $\mathcal{P}_i$, is partitioned into loading and unloading ports. At a loading port (or production port), the products are produced before the ship can load products, while at an unloading port (or consumption port), the ship discharges products and these products can then be sold. A port $p$ can be either a loading port or an unloading port. We define the parameter $I_{ip}$, which is equal to 1 and $-1$ if port $p$ in trade $i$ is a loading and unloading port, respectively, and 0 otherwise. The production rate, $R_{ip}$, which is assumed to be constant during the planning horizon, is

7

positive at loading (production) ports and negative at unloading (consumption) ports. At the beginning of the planning horizon, the initial inventory at port $p$ along trade $i$ is $S_{ip}^O$. Each depot located at a port $p$ has a limited inventory capacity given by an upper, $\bar{S}_{ip}$, and a lower bound, $\underline{S}_{ip}$. The cargo capacity of a ship $v$ is $Q_v^C$.

Transshipment only rarely occurs in Ro-Ro shipping and is therefore disregarded. Furthermore, it is assumed that only one ship can berth in a port at the same time to load or unload products.

The objective is to choose a service configuration for each voyage, assign the ships in the fleet to the voyages (i.e. determine the routes of the ships), and to decide the loading and unloading quantities at all ports along each voyage, so that the total transportation costs are minimized. Furthermore, we must make sure that all planned voyages are serviced within their time windows, and the inventory limits at each port are kept within their maximum and minimum limits at all times during the planning horizon.

It should be noted that we do not include inventory holding costs in the FDIMP. We assume instead, as in most literature on maritime inventory routing problems, that no matter where the cargo is (either at loading ports or at unloading ports), the holding cost is the same. This issue was discussed for the first time in Christiansen (1999).

## 3. Model Formulation

Section 3.1 presents the mathematical model for the FDIMP with port choice flexibility. Since this model includes some non-linear constraints, we show how these have been linearized in Section 3.2. Finally, several reduction tests used to remove elements from the arc sets $\mathcal{A}_v$ are presented in Section 3.3.

### 3.1. Model with Port Choice Flexibility

Here, we use the following binary variables: let the binary variable $x_{imcv}^O$ equal 1 if ship $v$ travels to the beginning of node $(i, m, c)$ right from its origin $o(v)$, and 0 otherwise. Let the binary variable $z_v^O$ equal 1 if ship $v$ is not deployed, and 0 otherwise. The binary variable $x_{imcjndv}$ equals 1 if ship $v$ travels right from node $(i, m, c)$ to node $(j, n, d)$ (i.e. ship $v$ operates at node $(i, m, c)$ and then subsequently traverses in ballast to node $(j, n, d)$ to operate at that node straightly), and 0 otherwise. The binary variable $z_{imcv}$ equals 1 if ship $v$ operates at node $(i, m, c)$ as its final voyage, and 0 otherwise. The binary variable $w_{imcv}$ equals 1 if ship $v$ visits node $(i, m, c)$, and 0 otherwise.

The time at which service of voyage $m$ of trade $i$ begins is given by the continuous variable $t_{im}$. The inventory levels at the beginning and end of service at port $p$ for voyage $m$ along trade $i$ are given by the continuous variables $s_{imp}$ and $s_{imp}^E$, respectively. If a port $p$ along trade $i$ is skipped for a voyage $m$, then $s_{imp}$ equals $s_{imp}^E$ as the processing time at a skipped port is 0. If the skipped port $p$ is located before the first port of the chosen configuration, then $s_{imp}$ is defined as the inventory of that port at the time of the visit to the configuration's first port. If the skipped port is after the last port of the chosen configuration, then $s_{imp}$ is assumed to be the inventory of that port at the time of the visit to the configuration's last port. If the skipped port $p$ is located between the first and the last port of the chosen

8

configuration, then $s_{imp}$ is measured by the inventory of that port at the time when the operation completes at the visited port which is called right before "arriving at" port $p$.

The continuous variable $l_{imcpv}$ represents the amount on board ship $v$ when departing from port $p$ on voyage $m$ of trade $i$ with configuration $c$. Finally, the continuous variable $q_{imcpv}$ denotes the quantity loaded/unloaded to/from ship $v$ at port $p$ of voyage $m$ of trade $i$ with configuration $c$.

*Objective Function.*

$$
\min \quad \sum_{v \in \mathcal{V}} \sum_{(i,m,c) \in \mathcal{N}_v} C_{icv}^R z_{imcv} + \sum_{v \in \mathcal{V}} \sum_{(i,m,c) \in \mathcal{N}_v} C_{icv}^O x_{imcv}^O + \\
\sum_{v \in \mathcal{V}} \sum_{((i,m,c),(j,n,d)) \in \mathcal{A}_v} C_{icjdv}^B x_{imcjndv} \tag{1}
$$

The objective function (1) seeks to minimize the total operating page 7costs, where the three terms represent the costs for the last voyage serviced by the ships, for the initial ballast sailing, and for the ballast sailing between consecutive voyages and for servicing all voyages except for the last one, respectively.

*Routing Constraints.*

$$
\sum_{(i,m,c) \in \mathcal{N}_v} x_{imcv}^O + z_v^O = 1, \quad v \in \mathcal{V} \tag{2}
$$

$$
x_{imcv}^O + \sum_{(j,n,d) \in \mathcal{N}_v} x_{jndimcv} - w_{imcv} = 0, \quad v \in \mathcal{V}, (i,m,c) \in \mathcal{N}_v \tag{3}
$$

$$
w_{imcv} - z_{imcv} - \sum_{(j,n,d) \in \mathcal{N}_v} x_{imcjndv} = 0, \quad v \in \mathcal{V}, (i,m,c) \in \mathcal{N}_v \tag{4}
$$

$$
z_v^O + \sum_{(i,m,c) \in \mathcal{N}_v} z_{imcv} = 1, \quad v \in \mathcal{V} \tag{5}
$$

Constraints (2) ensure that ship $v$ either departs to a node $(i, m, c)$ from its origin or is not deployed. Constraints (3) and (4) make sure that a ship entering a node also leaves from that node by either terminating its route or entering another node. Constraints (5) ensure that ship $v$ terminates after serving a node unless it is not utilized. In Figure 5, the relationship between the flow and visit variables $x_{imcv}^O$, $x_{imcjndv}$, $w_{imcv}$, $z_{imcv}$, and $z_v^O$ is illustrated for the same example as in Figure 4. All variables with value 0 are not shown in the figure.

*Voyage Fulfillment Constraints.*

$$
\sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} w_{imcv} = 1, \quad i \in \mathcal{I}, m \in \mathcal{M}_i \tag{6}
$$

Constraints (6) state that precisely one node inside a cluster is serviced by one ship in the fleet, i.e. that each voyage is serviced once with any configuration.
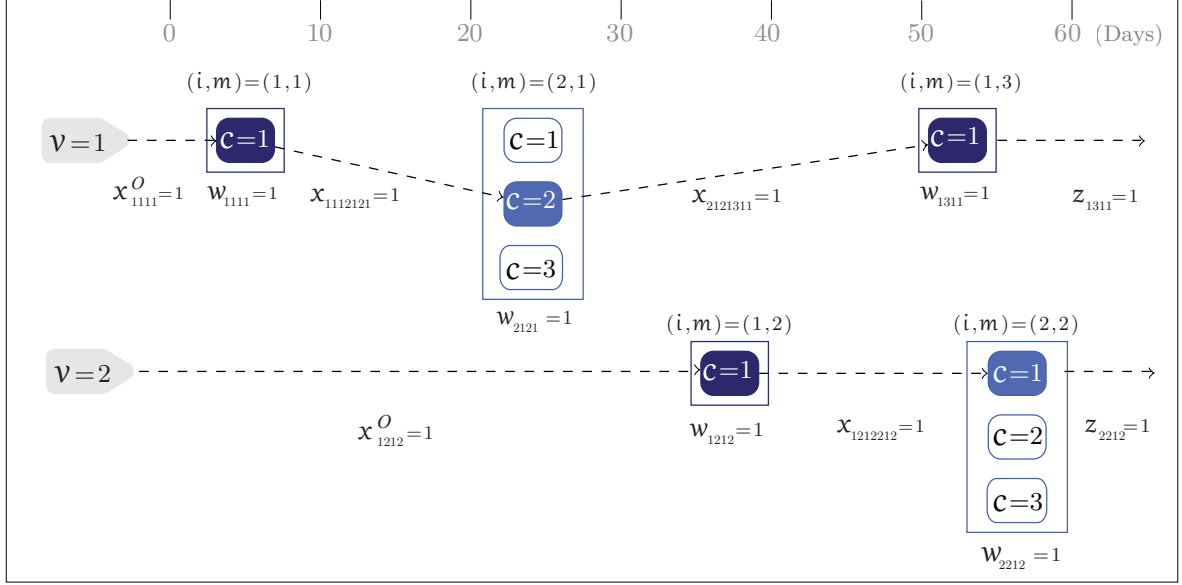
Figure 5: Illustration of the flow and visit variables for the example in Figure 4.

*Scheduling Constraints.*

$$t_{im} - \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} T^O_{icv} x^O_{imcv} \geq 0, \quad i \in \mathcal{I}, m \in \mathcal{M}_i \tag{7}$$

$$(t_{jn} - t_{im} - T_{icjdv}) x_{imcjndv} \geq 0, \quad v \in \mathcal{V}, ((i, m, c), (j, n, d)) \in \mathcal{A}_v \tag{8}$$

$$\underline{T}_{im} \leq t_{im} \leq \bar{T}_{im}, \quad i \in \mathcal{I}, m \in \mathcal{M}_i \tag{9}$$

Constraints (7) state that if ship $v$ traverses from its initial node to node $(i, m, c)$, then voyage $m$ of trade $i$ cannot start before the ship has arrived. Constraints (8) make sure that if ship $v$ travels directly between nodes $(i, m, c)$ and $(j, n, d)$, then the operation at voyage $n$ of trade $j$ must commence no earlier than the beginning time of operation at voyage $m$ of trade $i$ plus the service time of $(i, m, c)$ and the travel time between the nodes. Constraints (9) set the time windows associated with each node.

*Load Handling Constraints.*

$$l_{imcp_r v} = \begin{cases} I_{ip_r} q_{imcp_r v} & v \in \mathcal{V}, (i, m, c) \in \mathcal{N}_v, r = 1, p_r \in \mathcal{P}^C_{ic} \\ I_{ip_r} q_{imcp_r v} + l_{imcp_{r-1} v} & v \in \mathcal{V}, (i, m, c) \in \mathcal{N}_v, r \in \mathcal{R}^C_{ic} \setminus \{1\}, p_r \in \mathcal{P}^C_{ic} \end{cases} \tag{10}$$

$$l_{imcp_r v} = 0, \quad v \in \mathcal{V}, (i, m, c) \in \mathcal{N}_v, r = \hat{R}_{ic}, p_r \in \mathcal{P}^C_{ic} \tag{11}$$

$$0 \leq l_{imcpv} \leq Q^C_v w_{imcv}, \quad v \in \mathcal{V}, (i, m, c) \in \mathcal{N}_v, p \in \mathcal{P}^C_{ic} \tag{12}$$

10

$$0 \le q_{imcpv} \le \min\{Q_v^C, \bar{S}_{ip} - \underline{S}_{ip}\}w_{imcv}, \quad v \in \mathcal{V}, (i,m,c) \in \mathcal{N}_v, p \in \mathcal{P}_{ic}^C \tag{13}$$

$$\sum_{p \in \mathcal{P}_{ic}|I_{ip}=1} q_{imcpv} - \sum_{p \in \mathcal{P}_{ic}|I_{ip}=-1} q_{imcpv} = 0, \quad v \in \mathcal{V}, (i,m,c) \in \mathcal{N}_v \tag{14}$$

Constraints (10) calculate the load on board the ships in the course of their voyages. It is assumed that all ships are empty when each voyage starts. If port $p_r$ is the first port visited on a trade $i$ with configuration $c$, the load aboard is equivalent to the amount loaded at the first port of node $(i,m,c)$. When a ship sails from port $p_{r-1}$ to the subsequent port $p_r$, the load on board the ship leaving port $p_r$ must equal the load at the preceding port $p_{r-1}$ adjusted for the amount loaded or discharged at $p_r$. Likewise, each ship in the fleet is assumed to finish their voyages empty as in constraints (11). Constraints (12) are the ship capacity constraints, while constraints (13) enforce an upper limit on the amount loaded or discharged. Constraints (14) make sure that the amount loaded at loading ports equals the amount discharged at unloading ports during a voyage.
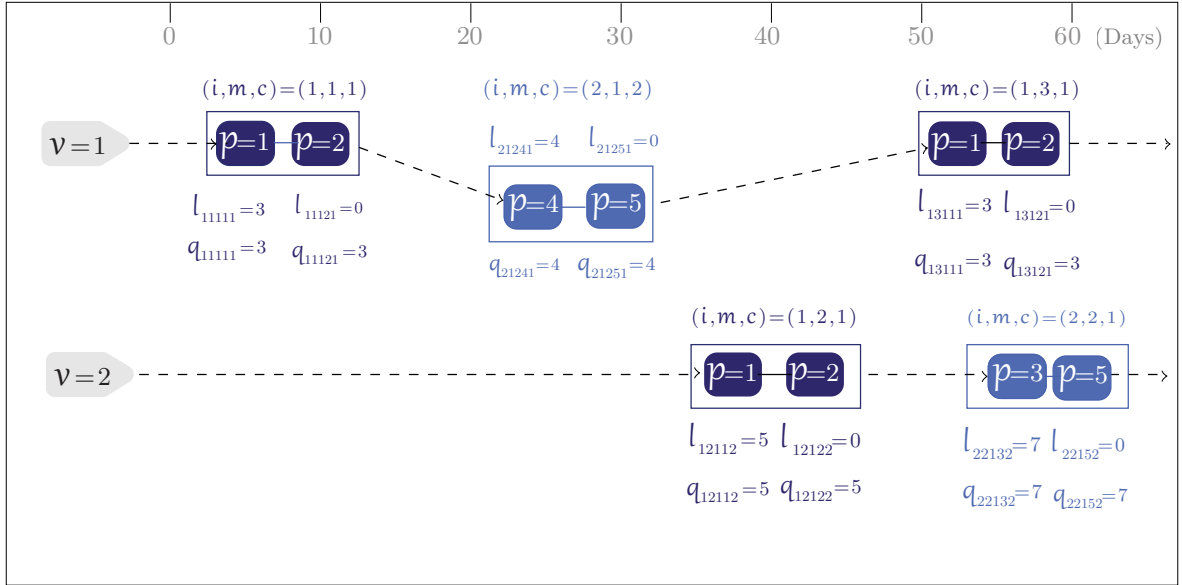


Figure 6: Illustration of the indicated load variables for the example in Figure 4. The values for the load variables are made up for illustrational purposes.

Figure 6 illustrates the relationship between the load variables continuing from the example in Figures 4 and 5. Ship 1 sails in ballast from its origin to port 1 to start servicing voyage 1 of trade 1 choosing its first configuration. In port 1, three units of products (e.g. 3000 CEUs) are loaded onto ship 1 before it sails to port 2 to unload. Then, the ship sails in ballast to port 4 to start servicing voyage 1 of trade 2 choosing its second configuration (with only ports 4 and 5). In port 4, the ship loads four units and then sails to port 5 to unload, before finally sailing to port 1 again to service voyage 3 of trade 1 (choosing its first configuration). In this example, both ship 1 and 2 use configurations with only two ports.

11

If we instead had chosen configuration 3 when servicing voyage 1 of trade 2 with ship 1 (see Figure 4), then the ship could for example have loaded two units of products both at loading ports 3 and 4, before unloading all four units at port 5.

*Inventory Management Constraints.*

$$s_{imp_r} - S_{ip_r}^O - R_{ip_r} t_{im} = 0, \quad i \in \mathcal{I}, m = 1, r = 1, p_r \in \mathcal{P}_i \tag{15}$$

$$s_{imp_r} - S_{ip_r}^O - R_{ip_r} \Big( t_{im} + \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} (T_{icp_r v}^R w_{imcv} + \sum_{r'=1}^{r-1} T_{ip_{r'}}^P w_{imcv}) \Big) = 0,$$
$$i \in \mathcal{I}, m = 1, r \in \mathcal{R}_i \setminus \{1\}, p_r \in \mathcal{P}_i \tag{16}$$

$$s_{imp}^E - s_{imp} + I_{ip} \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} q_{imcpv} - \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} T_{ip}^P R_{ip} w_{imcv} = 0,$$
$$i \in \mathcal{I}, m \in \mathcal{M}_i, p \in \mathcal{P}_i \tag{17}$$

$$s_{imp_r} = s_{i(m-1)p_r}^E + R_{ip_r} \Big( t_{im} + \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} (T_{icp_r v}^R + \sum_{r'=1}^{r-1} T_{ip_{r'}}^P) w_{imcv} -$$
$$t_{i(m-1)} - \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} (T_{icp_r v}^R + \sum_{r'=1}^{r-1} T_{ip_{r'}}^P) w_{i(m-1)cv} \Big),$$
$$i \in \mathcal{I}, m \in \mathcal{M}_i \setminus \{1\}, r \in \mathcal{R}_i, p_r \in \mathcal{P}_i \tag{18}$$

$$\underline{S}_{ip} \leq s_{imp} \leq \bar{S}_{ip}, \quad i \in \mathcal{I}, m \in \mathcal{M}_i, p \in \mathcal{P}_i \tag{19}$$

$$\underline{S}_{ip} \leq s_{imp}^E \leq \bar{S}_{ip}, \quad i \in \mathcal{I}, m \in \mathcal{M}_i, p \in \mathcal{P}_i \tag{20}$$

$$\underline{S}_{ip_r} \leq s_{imp_r}^E + R_{ip_r} \Big( T - t_{im} - \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}_i} (T_{icp_r v}^R - \sum_{r'=1}^{r-1} T_{ip_{r'}}^P) w_{imcv} \Big) \leq \bar{S}_{ip_r},$$
$$i \in \mathcal{I}, m = |M_i|, r \in \mathcal{R}_i, p \in \mathcal{P}_i \tag{21}$$

Constraints (15) - (16) calculate the inventory upon getting to each port along trade $i$ for its first voyage. Constraints (15) define the inventory at the first port on the trades, while constraints (16) do the same for the remaining ports. Constraints (17) calculate the inventory when operation finishes at any port visit $(i, m, p)$. This is measured by the inventory upon arrival at the port, adjusted for the loaded/discharged amount and production/demand during the operating time at the port. The inventory upon arriving at any port visit $(i, m, p)$ is defined in constraints (18) based on the inventory after service at the port in the preceding

visit $(i, m-1, p)$, adjusted for the amount production/demand in between the port visits. Constraints (19) – (20) indicate that the inventory at each port is always maintained within its given lower and upper limits, while constraints (21) make sure that the inventory levels at the end of the planning horizon are kept within the limits.

*Binary Requirements.*

$$z_v^O \in \{0, 1\}, \quad v \in \mathcal{V} \tag{22}$$

$$x_{imcv}^O, w_{imcv}, z_{imcv} \in \{0, 1\}, \quad v \in \mathcal{V}, (i, m, c) \in \mathcal{N}_v \tag{23}$$

$$x_{imcjndv} \in \{0, 1\}, \quad v \in \mathcal{V}, ((i, m, c), (j, n, d)) \in \mathcal{A}_v \tag{24}$$

The binary requirements for the variables are provided in constraints (22) – (24).

### 3.2. Linearization

Constraints (8) are non-linear and need to be linearized for the model to be solvable with a mixed integer programming solver. Therefore, we linearize and replace them with the following constraints (25), following Agra et al. (2017).

$$\sum_{v \in \mathcal{V}} \max\{\bar{T}_{im} + T_{icjdv} - \underline{T}_{jn}, 0\}x_{imcjndv} + t_{im} - t_{jn} - \bar{T}_{im} - \underline{T}_{jn} \leq 0,$$
$$((i, m, c), (j, n, d)) \in \mathcal{A} \tag{25}$$

Here, $\mathcal{A}$ is the complete set of arcs that exist across all ships.

### 3.3. Preprocessing

Several reduction tests can be used to remove elements from $\mathcal{A}_v$. The arc reductions for ship $v$ removing elements of $\mathcal{A}_v$ is based on the fact that a ship can only service a voyage if it can arrive at the starting point of the trade within its time window, i.e.:

- If $\underline{T}_{im} + T_{icjdv} > \bar{T}_{jn}$, then the arc $((i, m, c), (j, n, d))$ is eliminated.

- If $T_{icv}^O > \bar{T}_{im}$, then the arc $(o(v), (i, m, c))$ is eliminated.

## 4. Computational Study

This section presents the tests performed to evaluate the proposed model. Section 4.1 gives a short description of the test instances used in the computational study. In Section 4.2 results from solving the full model in Section 3 using commercial software are presented. Section 4.3 outlines computational results by comparing the results for the flexible model (Flex) with the base model (Base) without flexibility in skipping ports. Section 4.4 studies the effects on the solution quality of having different inventory limits.

The proposed model was implemented in Xpress Mosel ver 8.5.3, using the solver Xpress version 33.01.02. The model is designed to run on a standalone computer, but due to the number of tests it was decided to run all tests on a cluster. The nodes in the cluster are Lenovo NextScale nx360 M5 computers with an Intel E5-2643v3 6 core 2x 3.4GHz processor and 512 GB of RAM running on a Linux operating system.

## 4.1. Test Instance Generation

In order to test the mathematical model with flexibility in skipping ports, an instance generator was developed in Python. The instance generator is based on realistic ship data and shipping routes.

The dimension of the ship capacity is considered in CEUs (Car Equivalent Units). The vessel set contains two real types of ships, S and L, with capacities of 5800 and 7600 CEUs, respectively. A mixture of the two types are selected randomly for each test instance. A fixed sailing speed is assumed at 20 knots for all ships. The sailing distances between the ports in the trade are found in Ports.com (2018). The length of the trades and the chosen sailing speed results in about 10 to 50 sailing days along the different trades and about 0 to 30 sailing days for ballast sailing.

The ships' sailing costs are calculated based on the distances, their fuel consumption and the fuel prices. The fuel price is assumed to be 300 USD per ton. The cost of servicing a voyage depends on the type of ship chosen. Based on the ships' fuel consumption, we have approximated the sailing cost for ships of type S to 59 and 53 USD per nautical mile when loaded and in ballast, respectively, while the corresponding numbers for ships of type L are 77 and 69, respectively. Initial position of ships are distributed randomly among all possible loading ports of all selected trades.

The ports are geographically divided into four regions, namely Europe, North America, Asia, and Oceania. It is assumed that each region can be selected either as loading region or unloading region. A trade is generated by choosing a loading and an unloading region, and then connecting them together. It is assumed that the loading and unloading regions for a trade must be different from each other. The number of loading ports equals the number of unloading ports on a given trade.

Since we do not have real data for the different ports' production/consumption rates, we have generated these randomly within a range with the aim to provide reasonable values. The production/consumption rates are randomly chosen from the interval [0.8*avg, 1*avg], where avg= estimated total amount of cargo to be transported from/to the port divided by the length of the time horizon. The estimate is based on the average ship capacity, the number of voyages on the trade, and the number of selected loading/unloading ports on the trade.

We have generated test instances with three, six, and nine trades, respectively. Every three trades are selected such that they can form a closed loop. For example, Europe-North America, North America-Asia, Asia-Europe. The six trades are selected such that least three of them can form a closed loop, such as for example Europe-North America, North America-Asia, Asia-Europe, and Oceania-Europe, Europe-Asia, Asia-Oceania. The nine trades are also selected such that at least three of them can form a closed loop.

For each trade, we assume there are two voyages to be performed in each month (30-day period), and that these are evenly distributed over the month with disjunctive time windows. This means for example that voyage 1 on a given trade have the time window [0, 15] in days, voyage 2 [15, 30], and so on.

For our test, we consider 16 instances with different characteristics. Details about the instances (Inst.), number of trades (Trades), number of ports per trade (Ports), number of time periods in days (Time), total number of voyages to be serviced (Voyages), and number of ships (Ships) are presented in Table 1. The instances vary in terms of the number of trades, the number of ports per trade, the length of the planning horizon, the number of voyages, and the number of ships deployed. However, even though for example instances 1-4 all have three trades and four ports, each of them is generated as a totally new one with different trades, ports, and demands. For all instances, the service time at all ports is set to one day.

Table 1: Test Instances.

| Inst. | Trades | Ports | Planning horizon | Voyages | Ships |
|-------|--------|-------|------------------|---------|-------|
| 1 | 3 | 4 | 90 | 18 | 6 |
| 2 | 3 | 4 | 120 | 24 | 6 |
| 3 | 3 | 4 | 150 | 30 | 6 |
| 4 | 3 | 4 | 180 | 36 | 6 |
| 5 | 3 | 6 | 90 | 18 | 8 |
| 6 | 3 | 6 | 120 | 24 | 8 |
| 7 | 3 | 6 | 150 | 30 | 8 |
| 8 | 3 | 6 | 180 | 36 | 8 |
| 9 | 6 | 4 | 90 | 36 | 21 |
| 10 | 6 | 4 | 120 | 48 | 21 |
| 11 | 6 | 4 | 150 | 60 | 21 |
| 12 | 6 | 4 | 180 | 72 | 21 |
| 13 | 9 | 4 | 90 | 54 | 29 |
| 14 | 9 | 4 | 120 | 72 | 29 |
| 15 | 9 | 4 | 150 | 90 | 29 |
| 16 | 9 | 4 | 180 | 108 | 29 |

As the solution quality heavily depends on the inventory limits, we set these at three different levels, i.e. tight (T), medium (M), and loose (L) for each instance. The maximum inventory limit in the medium setting is calculated as $2 \times$ (the length of the planning horizon/the number of voyages per trade) $\times$ the average daily production (consumption) at the port. The maximum inventory limits in the T instances is set to half the inventory limits in the M instances, while the maximum inventory limits in the L instances is set to two times the limits in the M instances.

*4.2. Computational Results*

Here the computational results of solving the mathematical model given in Section 3 by commercial optimization software are presented. We set a maximum running time of 36,000 seconds (ten hours) for all instances, and default parameter settings were used for the Xpress Optimizer. In this section, we use the medium (M) inventory limits, and then test the effects of changing these to tight (T) and loose (L) in Section 4.4.

Table 2 summarizes the main results for the 16 test instances. The table gives information about the best (mixed) integer solution found (Total Costs) in ten million USD, the optimality gap (Gap) in percentage, the total numbers of rows (Rows) and columns (Columns) generated. The optimality gap (Gap), specified as $\text{Gap} = (\text{sol} - \text{bb})/\text{sol} \times 100\%$ where sol is the best integer solution value, acquired by the FICO Xpress optimizer, while bb is the value of the best bound on the optimal value at the end of the search. $\text{Gap} = 0\%$ correspond to the fact that the instance is solved to optimality for all practical purposes. The number of rows and columns reported is the number after the presolve.

Table 2: Computational results for medium (M) inventory limits.

| Inst. | Total Costs | Gap (%) | Rows | Columns |
|---|---|---|---|---|
| 1 | 1.09 | 0.00 | 26,888 | 36,326 |
| 2 | 1.42 | 1.78 | 46,823 | 81,575 |
| 3 | 2.11 | 0.20 | 71,042 | 146,354 |
| 4 | 2.55 | 0.16 | 80,170 | 201,006 |
| 5 | 1.24 | 0.11 | 752,638 | 921,463 |
| 6 | 1.44 | 0.15 | 1,221,399 | 2,115,777 |
| 7 | 1.74 | 0.01 | 2,025,265 | 4,438,077 |
| 8 | 2.14 | 0.02 | 2,651,931 | 7,090,732 |
| 9 | 2.09 | 0.00 | 204,408 | 271,212 |
| 10 | 3.09 | 0.00 | 384,344 | 650,636 |
| 11 | 3.47 | 0.04 | 814,826 | 1,783,755 |
| 12 | 4.16 | 0.00 | 2,107,606 | 5,610,795 |
| 13 | 3.26 | 0.12 | 740,106 | 960,158 |
| 14 | 4.10 | 0.01 | 2,544,758 | 4,429,125 |
| 15 | 5.41 | 0.00 | 3,584,370 | 7,846,644 |
| 16 | 6.44 | 0.00 | 5,813,197 | 15,487,622 |

The solver found integer solutions with relatively small gaps to all instances within the maximum running time. Instance 1 was solved to optimality in 20,601 seconds, while for all other instances the solver spent the maximum allowed time of 36,000 seconds. For instances 9, 10, 12, 15, and 16 we obtained gaps of less than 0.005% (shown as 0.00 in Table 2), while four other instances (i.e. instances 7, 8, 11, and 14) had gaps less than 0.05%. Actually, all instances, except for instance 2, which had a gap of 1.78%, had gaps of less than 0.2%. We also tried using a maximum running time of 3,600 seconds (one hour), but experienced

that the solver could not find a feasible integer solution for most of the instances. For the ones where the solver could find at least one feasible integer solutions, the gaps ranged from 0.07% to 14.75% after one hour.

As we can see from Table 2, the problem size is large for most of the instances. In fact, for seven of the instances the number of rows generated exceeds 1,000,000. Moreover, the number of columns generated is also considerable for some of the instances. More than one million columns were generated for eight of the instances. The number of columns generated is extremely large, more than 15 million for instance 16. For instances 5-8 the number of rows and columns generated is large, which indicates more complexity on the six port instances compared to the four port instances. This is because the number of possible trade configurations, and hence the number of variables, increases drastically with increased number of ports. We also note that the number of rows and columns generated is large for the instances 8, 12, and 16, which have longer planning horizons (180 days).

### 4.3. Comparison with the Base Model

Figure 7 presents the comparison of the solution quality for the Base model without skipping ports flexibility (as in Dong et al. (2017)) and the proposed Flex model. The Base model correspond to the Flex model in the case where only one configuration visiting all ports is generated for each trade.
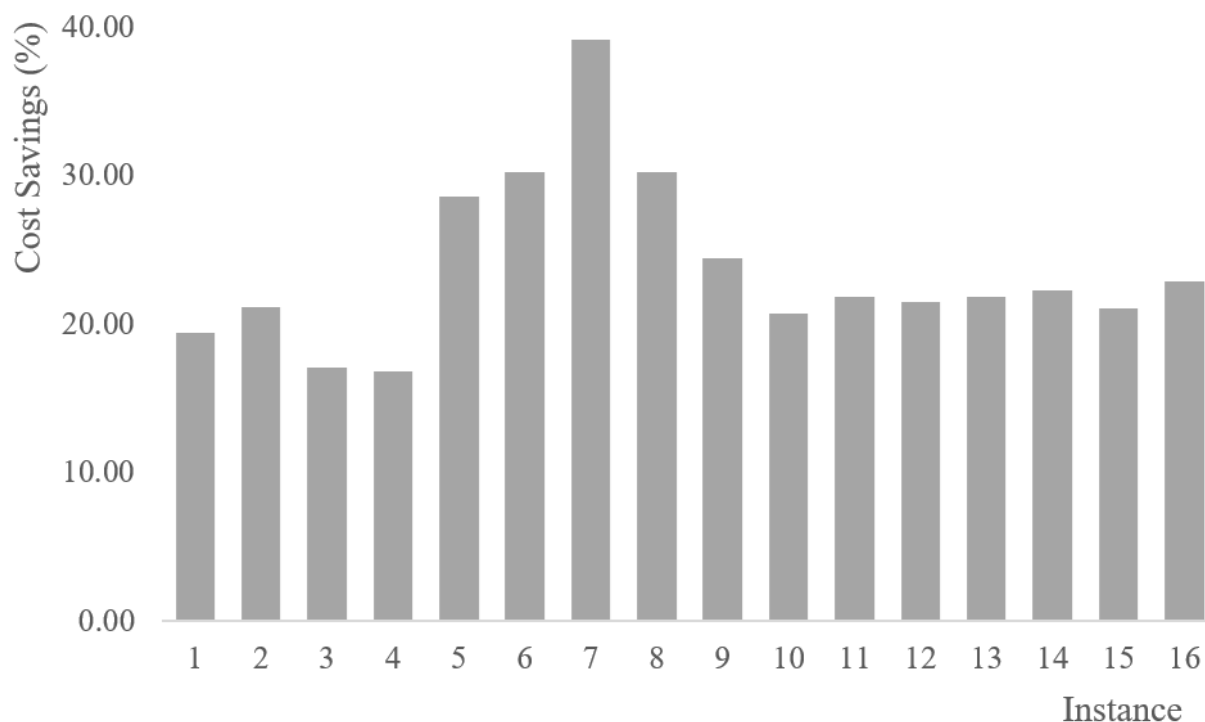


Figure 7: Cost savings achieved by utilizing the Flex model rather than Base model for medium (M) inventory limits.

Figure 7 gives details on the cost savings (the percentage savings in Total Costs) achieved by adding flexibility in skipping ports and utilizing the Flex model rather than Base model for medium (M) inventory limits in percentage (%). In terms of the computational times, the Base model is, as expected, much faster to solve when compared with the Flex model. Furthermore, the Base model found the optimal solution for all the 16 instances where the Flex model found the optimal solution for six of the 16 instances.

Even though the Flex model takes much longer time to solve than the Base model, the Flex model found significantly better solutions than the Base model for all the 16 instances. For these 16 instances, the cost saving is on average 23.73% (ranging from 16.86% to 39.19%). Thus, there can be substantial cost savings in utilizing the flexibility in skipping some port visits. This means there is a significant potential economic impact from changing the way of planning, i.e. by introducing the possibility of skipping ports, which possibly can be obtained if one gets a tighter collaboration with the shippers (customers).

The cost components for each instance with medium (M) inventory levels are reported in Table 3. For each instance we report the total costs (TC), the sailing costs along voyages (SC), the ballast sailing costs between voyages (BC), and the port visit costs (PVC) for both Base and Flex model. The total costs are the sum of the three parts: sailing costs, ballast sailing costs, and port visit costs.

Table 3: Analysis of the cost component for medium (M) inventory limit.

| Inst. | Base | | | | Flex | | | |
|---|---|---|---|---|---|---|---|---|
| | TC | SC | BC | PVC | TC | SC | BC | PVC |
| 1 | 1.36 | 1.08 | 0.07 | 0.22 | 1.09 | 0.96 | 0.00 | 0.14 |
| 2 | 1.80 | 1.39 | 0.12 | 0.29 | 1.42 | 1.21 | 0.04 | 0.17 |
| 3 | 2.54 | 1.94 | 0.24 | 0.36 | 2.11 | 1.67 | 0.24 | 0.20 |
| 4 | 3.07 | 2.33 | 0.31 | 0.43 | 2.55 | 2.00 | 0.32 | 0.23 |
| 5 | 1.74 | 1.34 | 0.07 | 0.32 | 1.24 | 1.04 | 0.05 | 0.15 |
| 6 | 2.07 | 1.47 | 0.16 | 0.43 | 1.44 | 1.22 | 0.03 | 0.18 |
| 7 | 2.87 | 2.01 | 0.32 | 0.54 | 1.74 | 1.48 | 0.04 | 0.22 |
| 8 | 3.06 | 2.21 | 0.20 | 0.65 | 2.14 | 1.86 | 0.01 | 0.27 |
| 9 | 2.76 | 2.13 | 0.20 | 0.43 | 2.09 | 1.78 | 0.05 | 0.25 |
| 10 | 3.90 | 3.05 | 0.28 | 0.58 | 3.09 | 2.63 | 0.14 | 0.32 |
| 11 | 4.45 | 3.47 | 0.25 | 0.72 | 3.47 | 3.03 | 0.04 | 0.40 |
| 12 | 5.31 | 4.16 | 0.28 | 0.86 | 4.16 | 3.64 | 0.05 | 0.47 |
| 13 | 4.17 | 3.21 | 0.31 | 0.65 | 3.26 | 2.79 | 0.09 | 0.38 |
| 14 | 5.28 | 4.12 | 0.30 | 0.86 | 4.10 | 3.59 | 0.03 | 0.49 |
| 15 | 6.85 | 5.30 | 0.47 | 1.08 | 5.41 | 4.59 | 0.23 | 0.59 |
| 16 | 8.35 | 6.45 | 0.61 | 1.30 | 6.44 | 5.52 | 0.21 | 0.70 |

It can be seen from Table 3 that the sailing costs play a significant role in total costs for both the Base and Flex model. When it comes to ballast sailing costs, it does not necessarily

increase when we increase the size of the instance. For example, instance 11 with a planning horizon of 150 days has less ballast sailing costs than instance 10 with 120 days. The reason is that the fleet is wisely utilized, reducing the ballast sailing substantially even though the whole planning horizon is longer. In addition, the Flex model reduces the port visit costs significantly compared with the Base model for all instances.

## 4.4. The Effects of the Inventory Limits

As described in Section 4.1, it can be expected that the solution quality heavily depends on the setting of the inventory limits. In this section we therefore test the different settings of the inventory limits to see their effects on the solutions. We test the tight (T) and loose (L) inventory limits, as defined in Section 4.1, in addition to the the medium (M) one, which was used in the previous tests.

Table 4 summarizes the results for the 16 test instances for all the three inventory limits, including information about the percentage gaps (Gap (%)) after the maximum running time. Furthermore, the table reports the Total Costs (TC) for medium (M) inventory limit, and the percentage cost increase in total costs for loose (L) and tight (T) inventory limits ($\Delta$ TC (%)), respectively, when compared with the medium (M) inventory limit.

Table 4: Comparison of gaps (%) and costs using inventory limits M, T, L.

| Inst. | Inv = M | | Inv = T | | Inv = L | |
|---|---|---|---|---|---|---|
| | Gap (%) | TC | Gap (%) | $\Delta$ TC (%) | Gap (%) | $\Delta$ TC (%) |
| 1 | 0.00 | 1.09 | 1.66 | 14.22 | 0.00 | -1.40 |
| 2 | 1.78 | 1.42 | 3.31 | 0.47 | 0.21 | -1.84 |
| 3 | 0.20 | 2.11 | 1.19 | 4.84 | 0.89 | 0.77 |
| 4 | 0.16 | 2.55 | 1.23 | 4.39 | 2.66 | 2.72 |
| 5 | 0.11 | 1.24 | 0.33 | 10.13 | 2.66 | 1.43 |
| 6 | 0.15 | 1.44 | 0.15 | 9.51 | 0.21 | -0.72 |
| 7 | 0.01 | 1.74 | 0.11 | 14.00 | 0.00 | -0.94 |
| 8 | 0.02 | 2.14 | 0.41 | 14.70 | 0.00 | -1.49 |
| 9 | 0.00 | 2.09 | 0.98 | 3.36 | 0.00 | -0.12 |
| 10 | 0.00 | 3.09 | 0.13 | 4.74 | 0.00 | -0.32 |
| 11 | 0.04 | 3.47 | 0.01 | 5.82 | 0.00 | -0.11 |
| 12 | 0.00 | 4.16 | 0.00 | 7.63 | 0.00 | -0.06 |
| 13 | 0.12 | 3.26 | 0.12 | 2.93 | 0.11 | -0.26 |
| 14 | 0.01 | 4.10 | 0.13 | 4.64 | 0.00 | -0.12 |
| 15 | 0.00 | 5.41 | 0.00 | 4.28 | 0.00 | -0.09 |
| 16 | 0.00 | 6.44 | 0.01 | 4.77 | 0.00 | -0.16 |

We see that the gaps are on average slightly higher for the tight and loose instances compared to the medium ones, but still very reasonable. In terms of costs, the medium and loose outperform the tight inventory limits for all instances, showing that reduced

transportation costs of up to 14.70% can be achieved with increased inventory limits from the tight to the medium ones. However, it should be noted that inventory costs are not included here, and these must be expected to increase with higher inventory limits. Furthermore, we see from Table 4 that increasing the inventory limits from medium to loose only gives marginal cost reduction on average. Actually, for instances 3, 4 and 5, the costs increase when going from medium to loose inventory limits. This should not be possible, as the solutions for the medium instances also should be feasible in the loose ones. However, the cost increase can be explained by the large gaps for the solutions with the loose limits for these instances.

The cost components with different inventory levels are analyzed in Figure 8. The costs are based on the average of the 16 instances for each model. The total costs are the sum of the three parts: sailing costs, ballast sailing costs, and port visit costs. The figure shows that the sailing costs is the dominant cost component in the solutions from both models and all inventory limits. However, we see that it can be reduced by utilizing the flexibility because of the reduced detouring to visit ports. Moreover, it can also be noted that the cost for ballast sailing is much higher for the Base model's solutions than for the Flex model's for all the three inventory limits. This implies that the way of planning results in a wisely utilized fleet, reducing the ballast sailing substantially even for the tight inventory limits. In addition, the port visit costs consist of only a small part of the total costs. Similar to the ballast sailing, it is obvious that the Flex model saves considerably in port visit costs compared with the Base model for all inventory limits.
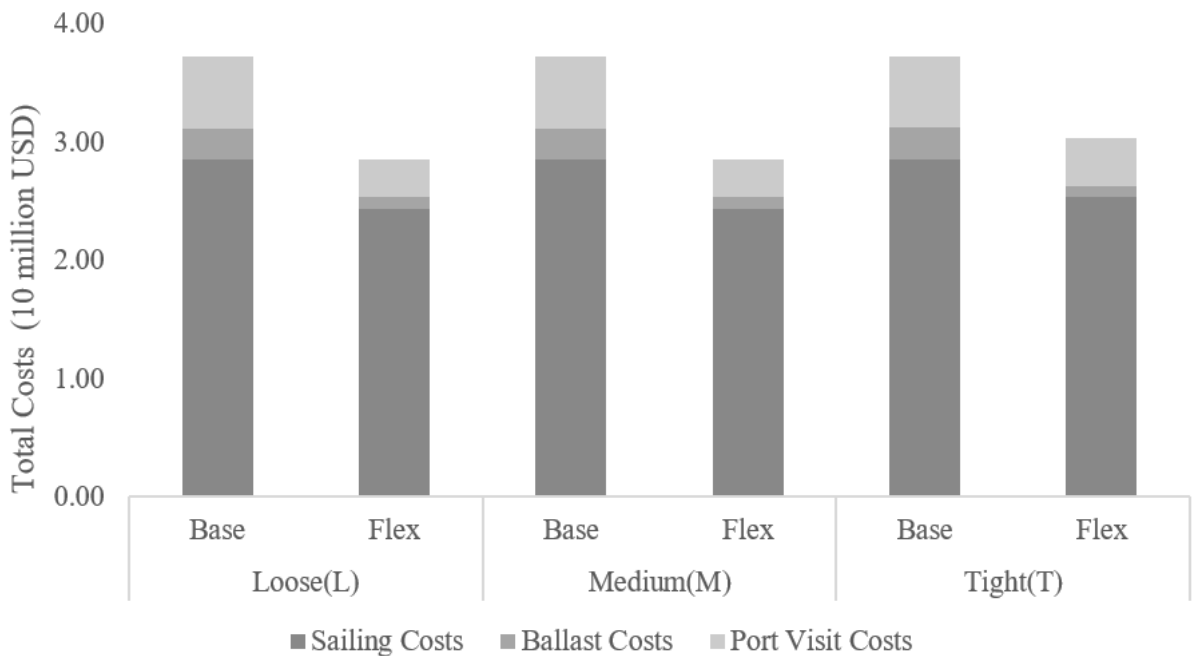


Figure 8: Cost Component Analysis for loose (L), medium (M) and tight (T) inventory limits.

Instead of looking at the total costs, the cost per unit transported is probably a more

fair criterion to use for comparing the different solutions. This can be explained by the fact that the unit cost criterion adjusts the cost factor with the quantities transported, which can vary among the different solutions (especially due to end of horizon effects). Figure 9 shows the average value of the unit cost for both Base and Flex model for the three inventory limits.
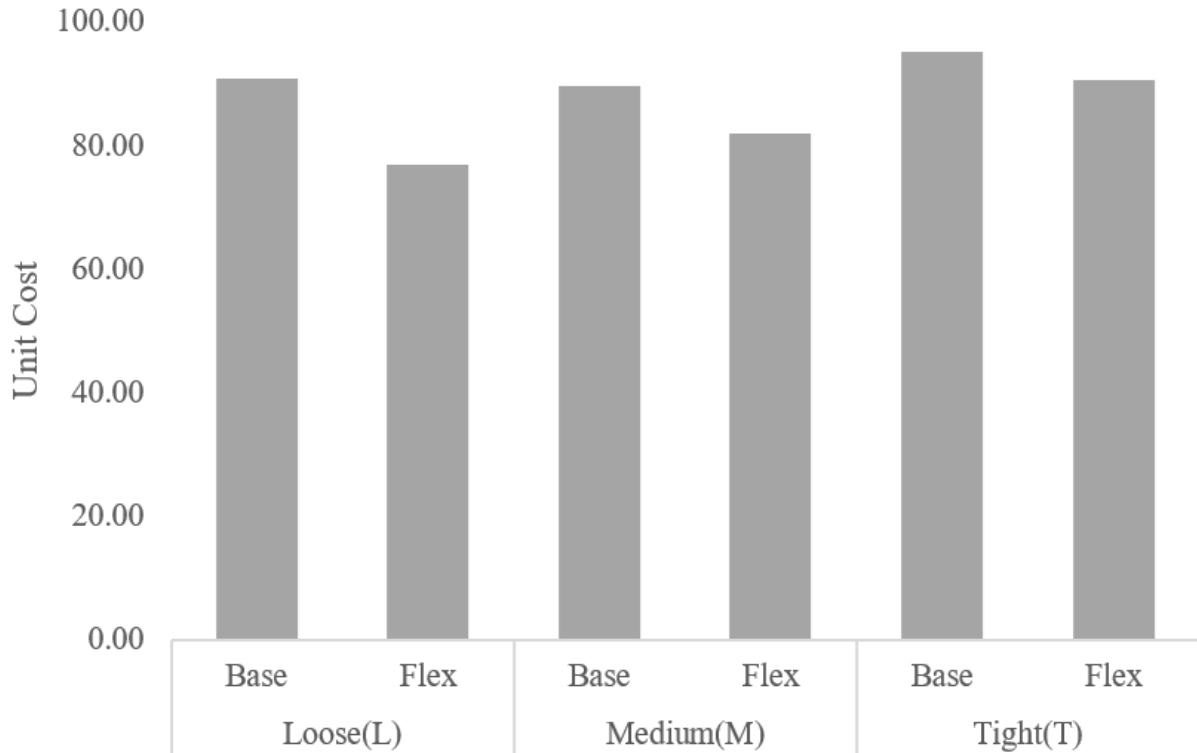


Figure 9: Unit cost for both Base and Flex model at three inventory limits.

On average, the solutions from the Flex model have smaller unit costs than the solutions from the Base model for all the three inventory limits. Looking at the three results with the Base model, the unit cost at tight inventory limits is the largest at 95.1, while the unit cost at medium inventory limits is the smallest at 89.7. Looking at the three results with the Flex model, the unit cost at tight inventory limits is the largest at 90.5, while the unit cost at loose inventory limits is the smallest at 76.8. Therefore, the tight inventory limit is, as expected, the most costly setting for both the Base and Flex models also in terms of unit cost.

To summarize, the results show the potential positive economic impact from changing the way of planning, i.e. by introducing the possibility of skipping ports, which possibly can be obtained if one gets a tighter collaboration with the shippers (customers).

## 5. Conclusion

We have in this paper considered a combined fleet deployment and inventory management problem (FDIMP) emerging in Roll-on Roll-off (Ro-Ro) shipping. Along a set of given trade routes with a sequence of port calls, a number of loading and unloading ports have inventories that should be kept within their limits. The current planning practice is that all ports are visited every time a trade route is serviced. Here, we instead aim at determining the sailing routes of each voyage along the trade route, i.e. which ports to be visited, where some ports along the trade route can be skipped on certain voyages. We propose a novel mixed integer programming model for this new and more flexible version of the FDIMP in Ro-Ro shipping. The problem is originally formulated as a nonlinear mixed-integer linear program which is then linearized.

This flexible model is promising as it finds good integer solutions with small gaps to most of the instances tested using a commercial solver. Furthermore, the flexible model gives much better results (23.73% savings in total costs on average over the tested instances) than the base model without any port skipping flexibility (resembling current planning practice). The results show the potential economic impact from changing the way of planning, i.e. by introducing the possibility of skipping ports, which possibly can be obtained if one gets a tighter collaboration with the shippers (customers). One achieves significant cost savings going from no flexibility (the Base model) to even the tight inventory limits, and also gets substantial further savings going to medium inventory levels, while one cannot gain much from extending these even further.

An interesting model extension for the FDIMP would be to include speed optimization, see for example Andersson et al. (2015). In some planning situations, it would also be relevant to extend the model to consider multiple products. However, we leave this as possibilities for future research.

## References

Agra, A., Christiansen, M., Delgado, A., 2017. Discrete time and continuous time formulations for a short sea inventory routing problem. Optimization and Engineering 18, 269–297.

Andersson, H., Fagerholt, K., Hobbesland, K., 2015. Integrated maritime fleet deployment and speed optimization: Case study from RoRo shipping. Computers & Operations Research 55, 233–240.

Chandra, S., Christiansen, M., Fagerholt, K., 2016. Combined fleet deployment and inventory management in roll-on/roll-off shipping. Transportation Research Part E: Logistics and Transportation Review 92, 43–55.

Christiansen, M., 1999. Decomposition of a combined inventory and time constrained ship routing problem. Transportation Science 33, 3–16.

Christiansen, M., Hellsten, E., Pisinger, D., Sacramento, D., Vilhelmsen, C., 2019. Liner shipping network design. European Journal of Operational Research doi:10.1016/j.ejor.2019.09.057.

Dong, B., Bektaş, T., Chandra, S., Christiansen, M., Fagerholt, K., 2017. A New Formulation for the Combined Maritime Fleet Deployment and Inventory Management Problem, in: Bektaş, T., Coniglio, S., Martinez-Sykora, A., Voß, S. (Eds.), Computational Logistics, Springer International Publishing. pp. 321–335.

Fagerholt, K., Johnsen, T.A.V., Lindstad, H., 2009. Fleet deployment in liner shipping: a case study. Maritime Policy & Management 36, 397–409.

Fischer, A., Nokhart, H., Olsen, H., Fagerholt, K., Rakke, J.G., Stålhane, M., 2016. Robust planning and disruption management in roll-on roll-off liner shipping. Transportation Research Part E: Logistics and Transportation Review 91, 51–67.

Gelareh, S., Meng, Q., 2010. A novel modeling approach for the fleet deployment problem within a short-term planning horizon. Transportation Research Part E: Logistics and Transportation Review 46, 76–89.

ISL, 2017. Shipping statistics and market review. Technical Report 9/10. Institute of Shipping Economics and Logistics.

Liu, X., Ye, H.Q., Yuan, X.M., 2011. Tactical planning models for managing container flow and ship deployment. Maritime Policy & Management 38, 487–508.

Meng, Q., Wang, S., Andersson, H., Thun, K., 2013. Containership Routing and Scheduling in Liner Shipping: Overview and Future Research Directions. Transportation Science 48, 265–280.

Ng, M., 2015. Container vessel fleet deployment for liner shipping with stochastic dependencies in shipping demand. Transportation Research Part B: Methodological 74, 79–87.

Ports.com, 2018. Sea routes and distances. `http://ports.com/sea-route/`. URL: `http://ports.com/sea-route/`. accessed: 2019-09-16.

Rodrigues, F., Agra, A., Christiansen, M., Hvattum, L.M., Requejo, C., 2019. Comparing techniques for modelling uncertainty in a maritime inventory routing problem. European Journal of Operational Research 277, 831–845.

Song, J.H., Furman, K.C., 2013. A maritime inventory routing problem: Practical approach. Computers & Operations Research 40, 657–665.

Wang, S., Meng, Q., 2012. Liner ship fleet deployment with container transshipment operations. Transportation Research Part E: Logistics and Transportation Review 48, 470–484.

Zhang, C., Nemhauser, G., Sokol, J., Cheon, M.S., Keha, A., 2018. Flexible solutions to maritime inventory routing problems with delivery time windows. Computers & Operations Research 89, 153–162.