

Methods for strategic liner shipping network design

Judith Mulder^{a,*}, Rommert Dekker^a

^a*Econometric Institute, Erasmus University Rotterdam, the Netherlands*

Abstract

In this paper the combined fleet-design, ship-scheduling and cargo-routing problem with limited availability of ships in liner shipping is considered. A composite solution approach is proposed in which the ports are first aggregated into port clusters to reduce the problem size. When the cargo flows are disaggregated, a feeder service network is introduced to ship the cargo within a port cluster. The solution method is tested on a problem instance containing 58 ports on the Asia-Europe trade lane of Maersk. The best obtained profit gives an improvement of more than 10% compared to the reference network based on the Maersk network.

Keywords: Transportation, liner shipping, network design, scheduling

1. Introduction

Seaborne shipping is the most important mode of transport in international trade. More than 80% of the international trade in 2010 is transported over sea (UNCTAD (2010)). In comparison to other modes of freight transport, like truck, aircraft, train and pipeline, ships are preferred for moving large amounts of cargo over long distances.

In the shipping market, three types of operations are distinguished: tramp shipping, industrial shipping and liner shipping (Lawrence (1972)). Tramp ships do not have a fixed schedule and are used for immediate deliveries where the most profitable freight is available. Therefore, the activities in tramp shipping are very irregular. In industrial shipping the cargo owner controls the ship and the objective becomes to minimize the cost of shipping. In liner shipping, ships follow a fixed route within a fixed time schedule; this is most common in the container trade.

The decision making in liner shipping can be distinguished on three different levels: the strategic, tactical and operational planning levels (Agarwal & Ergun (2008)). In the strategic planning level the optimal fleet-design is determined. This means that both the optimal number of ships in a fleet and the optimal ship sizes are determined in this level. This stage is very important, because the capital and operating cost in the (liner)

*Corresponding author, e-mail address: mulder@ese.eur.nl

shipping industry are very high. The ship-scheduling problem is solved in the tactical planning level. In this level, the service network is designed by creating ship routes and allocating the available ships to these routes. Finally, in the operational planning stage, it is determined which cargo is transported and which route(s) are used to ship the cargo. This problem is also referred to as the cargo-routing problem. The decisions made in a planning level influence the decision making in the other levels. Therefore, it could be profitable to solve the problems on the different levels simultaneously.

1.1. Literature review

Over the last decades, maritime transport has become a more popular field of research. In 1983 the first survey on ship routing and scheduling was published (Ronen (1983)). This survey gives a detailed overview on the research performed on ship routing and scheduling in the period before 1983. In a sequel, Ronen (1993) provides a detailed summary of published research on ship scheduling and related problem in the period from 1983-1993. Next, the survey of Christiansen et al. (2004) describes the major developments in the ship routing and scheduling problems in the period from 1994-2004. Finally, Wang & Meng (2011) give an overview of the most important existing literature on liner shipping studies and propose directions for further research.

Little research is performed on the determination of the optimal fleet design. Fagerholt (1999) develop a 3-phase solution approach to optimize the fleet size in liner shipping networks in which all feasible routes are generated and combined. Thereafter, the optimal fleet size is determined using a set partitioning problem, which is also solved in this phase. Powell & Perakis (1997) use an integer programming model to optimize the fleet deployment for a liner shipping company. They compare the results to the results obtained with a linear programming model. When using a linear programming model, manipulation of the results is needed to guarantee integer solutions. Both solution approaches become time consuming when the problem becomes larger.

Song et al. (2007) discuss a cargo allocation model with two objectives. The first objective is to minimize the unassigned cargo volume. The second objective is to minimize total costs corresponding to a given minimal unassigned cargo volume. Because the model is very difficult to solve by analytical methods, the solution space is first truncated. Thereafter, the authors select priority rules and make use of heuristics to find solutions of the model.

Most research is related to the combined ship-scheduling and cargo-routing problem. First, Rana & Vickson (1991) and Fagerholt (2004) present integer programming problems to solve the combined ship-scheduling and cargo-routing problem. They are able to solve small instances in a reasonable amount of time, but for larger instances their methods become too time consuming.

Next, some research exists in which mathematical programming methods, like Benders decomposition, are used to solve the combined ship-scheduling and cargo-routing problem (see for example Agarwal & Ergun (2008), Álvarez (2009) and Gelareh & Pisinger (2011)). These methods can be used to solve some very small instances to optimality, but for larger instances heuristic methods are still needed.

1.2. Contribution and outline

The objective of this study is to develop a service network in liner shipping. The service network should consist of a set of routes, the allocation of ships to the routes, the sailing speed of the ships on each route and the allocation of cargo over the routes. We assume that the liner shipping company is free to select the ships it needs to cover the routes, but we impose a maximum number of routes to which certain ship sizes can be allocated to prevent that only very large ships will be allocated in the network. In other words, our goal is to solve the fleet-design, ship-scheduling and cargo-routing problems simultaneously, where we consider the case with limited availability of ships.

In comparison to the current literature, our main contribution is to provide a solution approach for the integrated network design problem, in which we incorporate methods for fleet-design, ship-scheduling, speed-selection and cargo-routing. Since large liner shipping companies have to change their networks quite often, this integrated problem is essential in the current economy. In our test set, substantial savings can be obtained by solving the network design problem in an integrated way.

This paper is organized as follows. In Section 2 the problem is defined in more detail. Next, in Section 3 the methods used to solve the combined problem are proposed. Section 4 describes results from a case study and in Section 5 the main conclusions from this research are drawn.

2. Problem formulation

We consider the combined fleet-design, ship-scheduling and cargo-routing problem with limited availability of ships. In this section, first the three individual problems are described. Thereafter, a formulation of the combined problem is given.

2.1. Fleet-design problem

The goal of the fleet-design problem is to determine the optimal composition of the fleet. In this problem, both the number and the size of ships in the fleet have to be determined. For the shipping company it is important to determine the optimal fleet design, because the costs related to the fleet are very high. Costs related to the fleet composition can be distinguished in two types: fixed cost (e.g. Capital Expenditures (CAPEX)) and variable cost (e.g. Operating Expenditures (OPEX)).

In this paper, we investigate the optimal fleet design of a liner shipping company, so we do not consider an initial fleet. Therefore, the company can choose ships freely, although the number of routes that can be performed with ships is limited for each ship size. In this way, we prevent unrealistic outcomes with many large ships in the fleet. Finally, we assume that all ships are available at the beginning of the planning period.

The underlying route network and demand have to be considered when determining the fleet composition of a liner shipping company. However, the fleet design is determined for 10-20 years, because of the high cost incurred by replacing a ship. In such a period, the demand structure can change, which can cause changes in the route network. Therefore, when determining the optimal fleet design, both present and future demand have to be considered.

Economies of scale are another important factor in purchasing new ships. Larger ships usually have lower transportation cost per TEU than smaller ships. However, the fixed cost of larger ships are higher than that of smaller ships. The demand on the route that the ship will serve also influence the decision of the ship size.

2.2. Ship-scheduling problem

In the ship-scheduling problem, the service network has to be designed. A service network consists of a set of ship routes and the allocation of ships to the routes. Furthermore, the optimal sailing speed has to be determined for each ship route. A ship route is a sequence of ports that are visited by a ship. The ship routes are cyclic and consist of a westbound and an eastbound trip. Each port can be visited at most once on the westbound trip and at most once on the eastbound trip. So, ports can be visited twice on a ship route.

The allocation of ships to routes can be restricted, because for example a port on the route cannot handle a certain type of ship. Once a ship is allocated to a certain route, it will serve this route during the whole planning horizon. Most shipping companies operate schedules in which each route is served once a week to maintain a customer base and to provide customers with a regular schedule (Agarwal & Ergun (2008)). Therefore, in general the number of ships needed for one ship route has to be at least equal to the number of weeks needed to complete an entire round tour (rounded above). In this paper we will also require weekly route schedules.

2.3. Cargo-routing problem

In the cargo-routing problem, the shipping company makes two decisions. They decide which demands they accept and which routes are used to transport this cargo from the origin to the destination port. When the cargo-routing problem is solved as an individual problem, the service network is assumed to be known beforehand. Our goal is to maximize

the profit for one shipping company, so competition is not investigated. Revenues are obtained by transporting cargo between their origin and destination port. However, costs are also incurred by the transportation of the cargo. For some demand pairs the revenue that can be obtained will not exceed the cost incurred by transporting the cargo. This demand will then be rejected by the shipping company. Furthermore, it is possible that some profitable demands are rejected because other demands are more profitable.

When the demand of a demand pair is (partly) satisfied, the cargo will be picked up in the origin port and delivered at the destination port. When the origin port is visited on several ship routes, it has to be determined to which route the cargo is allocated. The same holds for the destination port. Some origin and destination ports will be visited on the same ship route, while other cargos have to be transhipped to other routes for which costs are incurred. All these decisions are made in the cargo-routing problem.

2.4. Combined fleet-design, ship-scheduling and cargo-routing problem

The decisions made in the three individual problems affect the decision making in the other problems as well. For example, when the service network is determined in the ship-scheduling problem, the network structure and capacity limits for the cargo-routing problem are set. This implies that a bad choice of service network in the ship-scheduling phase can result into lower profits in the cargo-routing phase. Therefore, it may be profitable to consider the individual problems at the same time.

In the combined fleet-design, ship-scheduling and cargo-routing problem, all decisions explained above in the three individual problems have to be taken at the same time. The problem becomes to construct a service network and determine the routes used to transport cargo such that the profit is maximized given a certain demand matrix and cost/revenue data. In this research, we limit the available ships that we can use in the service network. Therefore, we only try to select the best ships out of the available ships instead of solving the fleet-design problem from scratch. Here, we assume that the possible ship types are given with known characteristics.

In this paper we only include ship-related cost into the combined network design problem, which is essential to answer strategic questions for liner shipping companies. In practice, liner shipping companies also incur several other costs, such as administrative costs, costs related to the sales organization or headquarters. However, since we have no data on these costs, we leave them out of consideration.

3. Solution algorithm

In this section, we discuss the main ideas of the methods used to solve the combined fleet-design, ship-scheduling and cargo-routing problem. For a more detailed description of the methods, we refer to Electronic Supplementary Material E. Since the combined

fleet-design, ship-scheduling and cargo-routing problem is too large to solve efficiently and there is a lot of interaction between the different subproblems, we choose to aggregate the ports in the model and use a composite approach to solve the combined problem. Thereto, in Section 3.1, the method to aggregate ports is described. Next, in Section 3.2 we propose a method that generates initial feasible ship schedules and in Section 3.3 we construct a model that is able to optimally allocate the cargo to the available ships given these ship schedules. After solving the cargo routing model, the aggregated ports have to be disaggregated, which is explained in Section 3.4. To do this, we introduce feeder services to ship the cargo between the central port in a port cluster and all other ports of the same cluster. Finally, the genetic algorithm, described in Section 3.5, will generate new feasible ship schedules and this process will be repeated until a certain stopping criterion is met. Figure 1 shows a schematic overview of the solution algorithm.

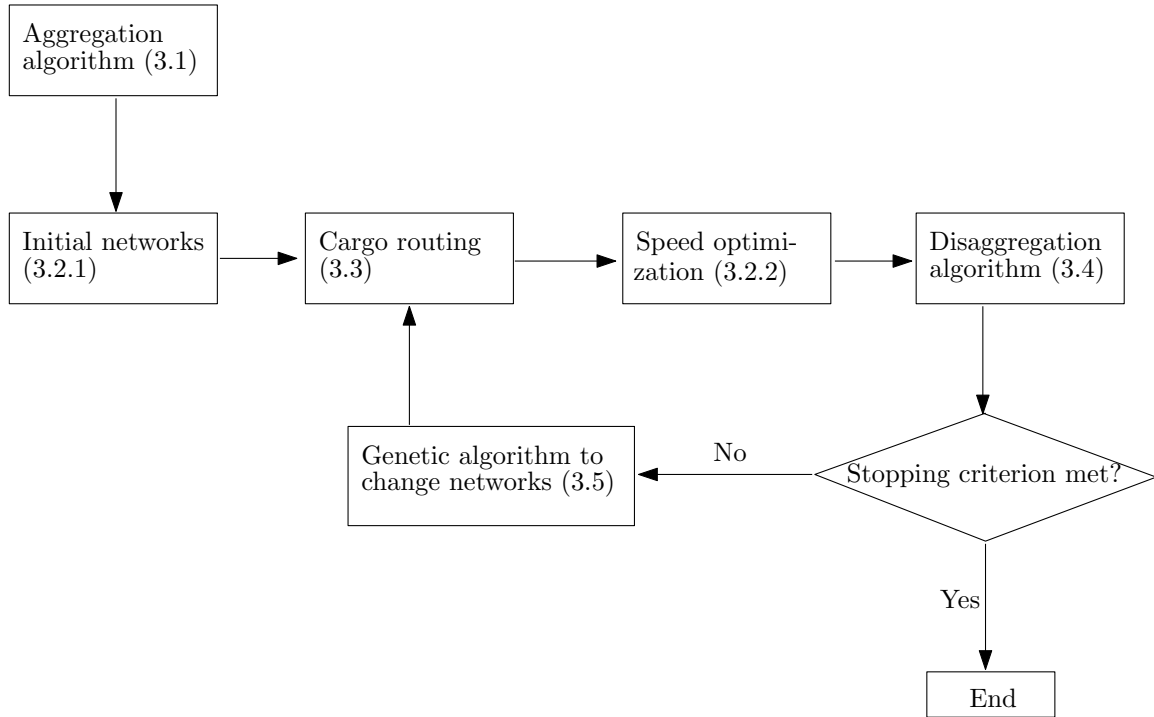


Figure 1: Schematic overview of the solution algorithm

3.1. Aggregation of ports

In Section 3.3, we will describe a linear programming model that can be used to solve the cargo routing over the ship routes. When the problem instance becomes larger, the computational time of the cargo routing model increases rapidly. Thus, for large problem instances, it is very time consuming to solve the cargo routing model repeatedly. The size of the problem instance decreases when the number of ports is reduced. Therefore, the

computational time of the cargo routing model can be decreased by reducing the number of ports. It depends on the computational time of the cargo routing model to which number of ports the problem has to be reduced.

In a model with aggregated ports, ships stop only once per cluster. For each port cluster, the stop should always be at the same place. Therefore, three major decisions have to be made during the aggregation process. First, the ports that are aggregated into a port cluster have to be determined. Next, one of the ports in a cluster has to be chosen as the central port (the port of the cluster where ships will make their port visit). Finally, the data of individual ports have to be aggregated to port cluster data.

Ports are aggregated based on their mutual distance. Ports that are relatively near to each other are clustered. An upper bound on the distance between two ports that belong to the same cluster can be imposed. To avoid problems, we consider the distance of a port that we will add to the cluster to the central port in the comparison. Thus, first the central port of the cluster has to be determined. Thereto, we make three lists of ports based on their expected yearly throughput. If a port has an expected yearly throughput that is at least twice as large as the average expected yearly throughput over all ports, the port is added to the list containing central ports. Furthermore, a list of ports that are not allowed to be central ports (noncentral ports) contains all ports that have a throughput that is at most one fifth of the average throughput. The last list consists of all remaining (intermediary) ports. At the end of the aggregation phase, when it is known which ports will be clustered into one port cluster, the decision of the central port in the cluster is reconsidered. The final central port in the cluster will be the port for which the expected cluster transshipment costs are minimized. The expected cluster transshipment cost will be determined by the product of the transshipment cost per TEU of the central port and the total expected yearly throughput of all noncentral ports in the cluster (as long as the maximum distance restriction is still satisfied). These lists can now be used to design the clusters as will be explained in the remainder of this section.

First, we create a cluster for each port on the central port list. The ports of the other two lists that are within the maximum distance to their closest central port are added to the corresponding cluster. If there are intermediary ports remaining that are not yet added to one of the clusters, a new cluster is created for the largest intermediary port, which becomes a central port. All intermediary and noncentral ports that are closest to this port (compared to the other central ports) and within the maximum distance are added to this new cluster (if a port was already allocated to one of the other clusters, it is removed from this old cluster). This is repeated until all intermediary ports are allocated to a cluster. Finally, if there are still some noncentral ports remaining, they are allocated to the cluster they are closest to. Note that in this last step the maximum

distance between ports in the same cluster is exceeded. However, since these ports have very little demand, it is probably quite costly to visit them on the main routes. Ships on the main network are generally larger (and thus more expensive) than ships on feeder services. Furthermore, the additional handling costs on the feeder network cannot be high, since the demand is low. Therefore, we decide to add these ports to an existing cluster instead of introducing a new cluster for them, forcing them to be visited on the main network.

Thus, the central port of a cluster is mainly chosen based on a combination of expected yearly throughput and transshipment cost. However, the distance to other ports also indirectly influences the central ports, because a maximum distance from a central port to all other ports in the cluster is imposed. Therefore, ports in regions with less throughput are still candidates to become a central port. Probably, some ports exist that are preferred to become central ports based on their location (such as Singapore), but are located in a region with less throughput. These ports will become central ports if they are the largest port in their region (unless the transshipment costs for these ports are too high). However, it is possible that another port in the region is larger, but has a worse location. In this case, our algorithm will select the larger port to be the central port. If we want to avoid this, it is possible to include a distance measure, that gives an indication of the amount of additional nautical miles that has to be sailed to visit the port on a route, in the determination of central ports. Furthermore, noncentral ports that are added to a cluster will in first instance be visited on a feeder network. However, in our solution approach, they can be added to the main route network in a later stage of the algorithm. Thus, if a noncentral port is located along a main route, this noncentral port will probably be added to the main route, saving (a part of) the feeder costs to serve this port on the feeder network.

Finally, we have to aggregate the individual port data into port cluster data. Relevant port data in the model are: distance, demand, port cost, transshipment cost, (un)loading cost and port time. The distance, costs and port time only depend on the port at which the ship stops. Therefore, for these data the port cluster data is the same as the individual port data of the central port. The demand data also depends on the demand of the other ports in the cluster. Cluster demand equals the sum of the individual port demand in the cluster. Note that demand between ports in the same cluster disappears during the aggregation process. This demand can be reviewed after the disaggregation process.

3.2. Designing an initial liner shipping network

In our research we consider routes in which ports can be visited at most once on the eastbound and at most once on the westbound trip. Furthermore, we clustered ports into port clusters, so we reduced the maximum number of visits on a route. Since there are

not that many port clusters, they are generally not very close to each other, such that a natural geographical order can be obtained. Using this order, we generate initial routes and determine the optimal speed on each route.

3.2.1. Generate initial routes

Initial routes are generated at random. First, we determine randomly the number of routes in the network between a minimum and maximum. Thereafter, for each route in the network, a cluster is called with a certain probability. Thus, we determine in geographical order at random whether a cluster will be visited or not. The routes obtained using this method are not always feasible routes. Therefore, we check during the generation of the routes whether they satisfy the following three conditions and change them when necessary.

The first condition is that the beginning cluster of each route should be equal to the end cluster, so that a round tour is made. If this is not satisfied, the beginning or end cluster is adjusted, so that this condition is met. The second condition is that the two middle clusters of the routes should be unequal, because otherwise the same cluster is visited twice in a row. we remove one of the middle clusters if this condition is violated. The last condition is that the route length is at least equal to the minimum route length. If this condition is violated for a certain route, this route is deleted.

Finally, a ship type is allocated to the ship route. Thereto, we select randomly a ship type from the set of available ship types. Note that this set has to be updated when the last available ship of a certain type is allocated to a route. In this case, the ship type is removed from the set of available ship types.

Using the above procedure a network is obtained with a random number of feasible routes. This network can be used to run the cargo allocation model and obtain the different flows. Later on, the networks will be changed using a genetic algorithm based method, so that better networks are constructed.

3.2.2. Determine optimal speed

When the routes and capacities of the liner ships are known, it is possible to determine the optimal speed of the liner ship serving a certain route. In this section, the method to determine this optimal speed will be explained.

Consider a route of the route network and the capacity of the liner ship used to serve this route. To maintain a weekly frequency, the number of ships needed will always equal the route duration, thus more ships are needed for lower sailing speeds. For each integer number of weeks between the minimum and maximum route duration (based on the maximum and minimum speed respectively, where the route duration may not exceed the maximum route duration imposed in the model), we calculate the cost of sailing the

route in the given number of weeks. These costs include the capital, operating and fuel costs. After we calculated the costs for all feasible route durations, we select the route duration and corresponding speed that is associated to the lowest costs.

Thus, we determine the optimal sailing speed by a simple enumeration procedure. We will repeat this procedure for each route to obtain the optimal sailing speed for each individual route.

3.3. Cargo Routing Model

In the cargo-routing problem, we want to allocate the cargo flows over the route network given the origin-destination demand matrix and capacities on the routes. The cargo-routing problem can then be formulated as a multi-commodity flow problem, for which the linear programming formulation is given below. We will refer to this model as the cargo routing model (CRM). Introduce the following sets:

- $h \in \mathcal{H}$ Set of ports.
- $t \in \mathcal{T} \subseteq \mathcal{H}$ Set of transshipment ports.
- $s \in \mathcal{S}$ Set of ship routes.
- $j \in \mathcal{J}$ Indicator set denoting whether a ship passes both ports $h_1 \in \mathcal{H}$ and $h_2 \in \mathcal{H}$ on ship route $s \in \mathcal{S}$, where $j = (h_1, h_2, s)$.
- $k \in \mathcal{K}$ Indicator set denoting whether port $h_2 \in \mathcal{H}$ is directly visited after port $h_1 \in \mathcal{H}$ on ship route $s \in \mathcal{S}$, where $k = (h_1, h_2, s)$.

The following parameters are used in the model:

- r_{h_1, h_2} Revenue of transporting one TEU from port $h_1 \in \mathcal{H}$ to port $h_2 \in \mathcal{H}$.
- c_t^t Cost of transshipping one TEU in transshipment port $t \in \mathcal{T}$.
- c_h^h Cost of (un)loading one TEU in origin or destination port $h \in \mathcal{H}$.
- d_{h_1, h_2} Demand with origin port $h_1 \in \mathcal{H}$ and destination port $h_2 \in \mathcal{H}$.
- b_s Capacity on ship route $s \in \mathcal{S}$.
- $I_{h_1, h_2, h_3, h_4, s}^{path}$ 0/1 parameter that takes the value 1 if a ship passes consecutive ports $h_3 \in \mathcal{H}$ and $h_4 \in \mathcal{H}$ when sailing from port $h_1 \in \mathcal{H}$ to port $h_2 \in \mathcal{H}$ on ship route $s \in \mathcal{S}$.

In the model formulation, we distinguish between direct flows and transshipment flows. Direct flows are cargo flows between the origin and destination port of a demand pair for which no transshipment movement has to be made. Cargo flows for which transshipment movements are necessary are called transshipment flows. Now, we introduce the following

decision variables:

- $x_{h_1, h_2, s}$ Cargo flow on ship route $s \in \mathcal{S}$ between consecutive ports $h_1 \in \mathcal{H}$ and $h_2 \in \mathcal{H}$.
- $x_{h_1, h_2, s}^{od}$ Direct cargo flow on ship route $s \in \mathcal{S}$ between ports $h_1 \in \mathcal{H}$ and $h_2 \in \mathcal{H}$.
- $x_{h_1, t, h_2, s}^{ot}$ Transshipment flow on ship route $s \in \mathcal{S}$ between port $h_1 \in \mathcal{H}$ and transshipment port $t \in \mathcal{T}$ with destination port $h_2 \in \mathcal{H}$.
- x_{t, h, s_1, s_2}^{td} Transshipment flow on ship route $s_2 \in \mathcal{S}$ between transshipment port $t \in \mathcal{T}$ and destination port $h \in \mathcal{H}$, where the flow to transshipment port $t \in \mathcal{T}$ was transported on ship route $s_1 \in \mathcal{S}$.
- $x_{t_1, t_2, h, s_1, s_2}^{tt}$ Transshipment flow on ship route $s_2 \in \mathcal{S}$ between transshipment port $t_1 \in \mathcal{T}$ and transshipment port $t_2 \in \mathcal{T}$ with destination port $h \in \mathcal{H}$, where the flow to transshipment port $t_1 \in \mathcal{T}$ was transported on ship route $s_1 \in \mathcal{S}$.
- $x_{h_1, h_2, s}^{tot}$ Total cargo flow on ship route $s \in \mathcal{S}$ between ports $h_1 \in \mathcal{H}$ and $h_2 \in \mathcal{H}$.

Now, we can give the linear programming formulation.

$$\begin{aligned}
\max \quad & \sum_{h_1 \in \mathcal{H}} \sum_{h_2 \in \mathcal{H}} \sum_{s \in \mathcal{S}} r_{h_1, h_2} \left(x_{h_1, h_2, s}^{od} + \sum_{t \in \mathcal{T}} x_{h_1, t, h_2, s}^{ot} \right) \\
& - \sum_{h_1 \in \mathcal{H}} c_{h_1}^h \left(\sum_{t \in \mathcal{T}} \sum_{h_2 \in \mathcal{H}} \sum_{s \in \mathcal{S}} [x_{h_1, t, h_2, s}^{ot} + x_{h_2, t, h_1, s}^{ot}] + \sum_{h_2 \in \mathcal{H}} \sum_{s \in \mathcal{S}} [x_{h_1, h_2, s}^{od} + x_{h_2, h_1, s}^{od}] \right) \\
& - \sum_{t_1 \in \mathcal{T}} c_{t_1}^t \left(\sum_{t_2 \in \mathcal{T}} \sum_{h_2 \in \mathcal{H}} \sum_{s_1 \in \mathcal{S}} \sum_{s_2 \in \mathcal{S}} x_{t_1, t_2, h_2, s_1, s_2}^{tt} + \sum_{h_2 \in \mathcal{H}} \sum_{s_1 \in \mathcal{S}} \sum_{s_2 \in \mathcal{S}} x_{t_1, h_2, s_1, s_2}^{td} \right) \quad (1)
\end{aligned}$$

subject to

$$\sum_{t \in \mathcal{T}} \sum_{s \in \mathcal{S}} x_{h_1, t, h_2, s}^{ot} + \sum_{s \in \mathcal{S}} x_{h_1, h_2, s}^{od} \leq d_{h_1, h_2} \quad h_1 \in \mathcal{H}, h_2 \in \mathcal{H} \quad (2)$$

$$x_{h_1, h_2, s} \leq b_s \quad (h_1, h_2, s) \in \mathcal{K} \quad (3)$$

$$\begin{aligned} \sum_{h_1 \in \mathcal{H}} x_{h_1, t_1, h_2, s_1}^{ot} + \sum_{t_2 \in \mathcal{T}} \sum_{s_2 \in \mathcal{S}} x_{t_2, t_1, h_2, s_2, s_1}^{tt} \\ - \sum_{s_2 \in \mathcal{S}} x_{t_1, h_2, s_1, s_2}^{td} - \sum_{t_2 \in \mathcal{T}} \sum_{s_2 \in \mathcal{S}} x_{t_1, t_2, h_2, s_1, s_2}^{tt} = 0 \quad (t_1, h_2, s) \in \mathcal{J} \end{aligned} \quad (4)$$

$$x_{h_1, h_2, s} - \sum_{h_3 \in \mathcal{H}} \sum_{h_4 \in \mathcal{H}} x_{h_3, h_4, s}^{tot} I_{h_3, h_4, h_1, h_2, s}^{path} = 0 \quad (h_1, h_2, s) \in \mathcal{K} \quad (5)$$

$$\begin{aligned} x_{h_1, h_2, s_1}^{tot} - x_{h_1, h_2, s_1}^{od} - \sum_{h_3 \in \mathcal{H}} x_{h_1, h_2, h_3, s_1}^{ot} \\ - \sum_{s_2 \in \mathcal{S}} x_{h_1, h_2, s_2, s_1}^{td} - \sum_{h_3 \in \mathcal{H}} \sum_{s_2 \in \mathcal{S}} x_{h_1, h_2, h_3, s_2, s_1}^{tt} = 0 \quad h_1 \in \mathcal{H}, h_2 \in \mathcal{H}, s_1 \in \mathcal{S} \end{aligned} \quad (6)$$

$$x_{h_1, h_2, s} \geq 0 \quad (h_1, h_2, s) \in \mathcal{K} \quad (7)$$

$$x_{h_1, h_2, s}^{od} \geq 0 \quad h_1 \in \mathcal{H}, h_2 \in \mathcal{H}, s \in \mathcal{S} \quad (8)$$

$$x_{t_1, t_2, h, s_1, s_2}^{tt} \geq 0 \quad h \in \mathcal{H}, s_1 \in \mathcal{S}, (t_1, t_2, s_2) \in \mathcal{J} \quad (9)$$

$$x_{t, h, s_1, s_2}^{td} \geq 0 \quad s_1 \in \mathcal{S}, (t, h, s_2) \in \mathcal{J} \quad (10)$$

$$x_{h_1, t, h_2, s}^{ot} \geq 0 \quad h_2 \in \mathcal{H}, (h_1, t, s) \in \mathcal{J} \quad (11)$$

The objective (1) of the cargo routing problem is to maximize total profit. Profit is given by the revenue minus the costs. The costs only consist of (un)loading cost and transshipping cost, because the route network is given, so all other costs are fixed and can be subtracted afterwards.

Constraints (2) ensure that the total cargo shipped from one port to another does not exceed the demand of that port combination. Next, Constraints (3) make sure that the total load of a ship between each two consecutive ports does not exceed the capacity of the ship. Constraints (4) ensure that the flow to a transshipment port with destination port $h_2 \in \mathcal{H}$ has to equal the flow from that transshipment port to port h_2 . In other words, they make sure that all flow unloaded to be transhipped, will also be loaded on another route. Constraints (5) define the the amount of flow between two consecutive ports and Constraints (6) define the total flow between each two ports in the same cycle. Finally, Constraints (7)-(11) guarantee a nonnegative flow between each two ports.

3.4. Disaggregation method

The cargo routing model can be executed with the clusters as determined in Section 3.1 as input. We will obtain cargo flows between the clusters as output from the cargo routing model. In practice, it is necessary to know the exact route of each load from

the origin port to the destination port. Therefore, we first have to determine the real origin and destination ports of each cargo flow. Thereafter, we will use feeder services to ship the cargo from the origin to the destination port in the cluster, so a feeder network has to be constructed. The problem of constructing a feeder network is similar to the vehicle routing problem with pickups and deliveries (see for example Berbeglia et al. (2007) for an overview). Nagy & Salhi (2005) provide some routines that can be used to improve existing routes for this problem. When constructing the feeder service, in first instance, a feeder service is added from the central port in the cluster to each other port belonging to the cluster. Thereafter, the feeder routes will be improved, for which we will use a method to decrease the ship sizes on the feeder routes, a method comparable to the SWITCH routine discussed in Nagy & Salhi (2005) and a method in which feeder ports are added to the main route network. Figure 2 shows an schematic overview of the disaggregation method.

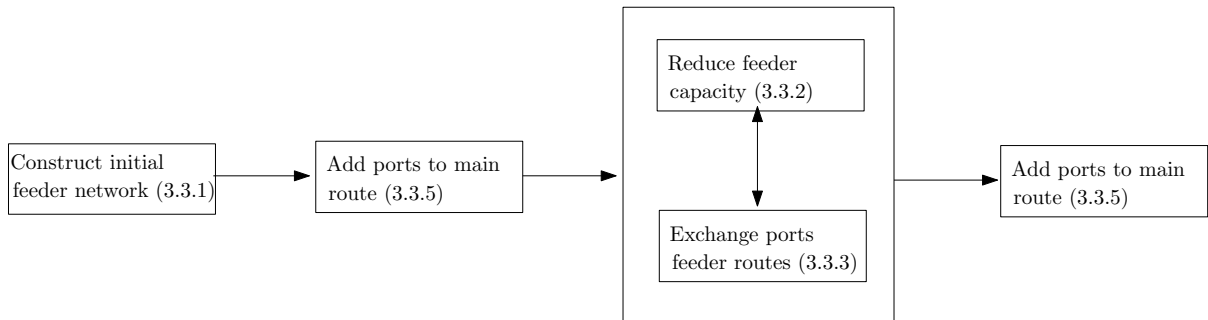


Figure 2: Schematic overview of the disaggregation method

3.4.1. Constructing initial feeder network

The cargo flow between each pair of clusters is obtained from the CRM model. For each cluster, we know which ports belong to this cluster. For each port pair with origin port in the origin cluster and destination port in the destination cluster, we can determine the revenue of transporting cargo between these ports. Furthermore, we know the demand between these two ports. Then, the disaggregation is done by repeatedly allocating as much cargo as possible to the port pair with the highest revenue until the total cargo flow is allocated. Because the cluster demand equals the sum of the individual port demand of the ports in the cluster and all possible combinations of port pairs with origin in the origin cluster and destination in the destination cluster are considered in the disaggregation method, the total transhipped cargo is always fully allocated using this method. Finally, this procedure is repeated for all combinations of two port clusters of the cargo routing model.

When all combinations are considered and the cargo routing between each two real ports is known, the size of the feeder services can be determined. In first instance, for each noncentral port in a cluster, a feeder service is made. This feeder ship will then sail from the central port of the cluster to the noncentral and back to the central port. The size of the ship can be obtained when considering the cargo transhipped from and to the noncentral port. These amounts will not be on the ship at the same time, so the maximum of the amount of cargo transhipped to and the amount transhipped from the noncentral port is the maximum load on the feeder ship. We will then use the ship that has minimal size but can still transport the maximum load. Note that only feeder services that sail with a frequency of once a week are considered. Exemptions are only made for ports that are placed in a cluster because their demand is too low, but cannot be served within one week with a direct feeder route from the central port of the cluster. For these ports, we can maintain a weekly frequency by assigning more than one ship to the route. However, this can be more costly, because more ships are needed. Therefore, in these cases we allow ports to be visited only once every two weeks if this is cheaper. Thus, we compare the costs of visiting the port every week (using two feeder ships for that route) with the costs of visiting the port every week (such that only one feeder ship is needed for the route).

3.4.2. Reducing the size of a feeder service

In this section, we describe the method used to reduce the size of the existing feeder services by reallocating the origin and/or destination port of a cluster cargo obtained from the cargo routing model. Each time, we select a port pair and consider the cargo flow from the origin to the destination port. When we start, all feeder services are direct services between the central port in a cluster and a noncentral port in the same cluster. In this case, only two cases have to be distinguished when reducing the size of a feeder service. The noncentral port can be the origin port of a cargo flow, in which case the cargo is on board when the ship sails from the noncentral port to the central port of the cluster or the noncentral port is the destination port of a cargo flow and the cargo is on board when the ship sails from the central to the noncentral port. In both cases, the cargo is only at one of the two legs on board, so only one leg has to be considered for each cargo flow. However, when ports are exchanged between feeder services, some feeder services are created that visit more than one noncentral port. In this case more legs have to be considered when a cargo flow is viewed. This makes the size reducing process more complicated.

The next algorithm describes the steps that have to be performed to determine the increase in profit when the size of a feeder service is reduced. The algorithm will be performed for each cluster separately. Note that no real changes are made in the algorithm.

So, when the algorithm starts over in step 1, the data is still the same as at the beginning. If a change is mentioned in the algorithm it is a temporarily change, which only holds during one iteration of the algorithm. In Electronic Supplementary Material E a more detailed explanation of the steps in the algorithm can be found.

1. Consider a feeder service in the cluster. Determine the capacity of the feeder service when it is reduced by one size.
2. Determine the reduction needed on each leg of the feeder service.
3. Repeat the following as long as the sum of the reduction needed over all legs is larger than 0 and not all port combinations are considered.
 - (a) Exchange as much cargo as possible between the port combinations that are not yet considered and have the lowest revenue decrease.
 - (b) Update the reduction needed on each leg.
4. Check whether the sum of the reduction needed is 0.
 - (a) If the sum is 0, determine the increase in profit when the exchanges are performed.
 - i. If the increase in profit is higher than the highest increase found earlier, save the new increase in profit and the reallocation of demand needed to decrease the feeder size.
 - (b) Else, the reduction is not possible.
5. If all feeder services are considered, then stop. Else, return to step 1.

3.4.3. *Exchanging a port between two feeder services*

In this step a port is exchanged between two feeder services. The cargo allocation does not change during this step, so the revenue and feeder handling costs will also not change. Because the costs on the main route will also stay the same, the only changes will occur in the feeder capital, operating and fuel costs. Finding a profitable exchange corresponds now to finding an exchange for which these feeder costs are reduced.

We will explain the method used to exchange ports using an algorithm. A more detailed description of the steps of the algorithm can again be found in Electronic Supplementary Material E. Again, the algorithm will be executed for each cluster separately and the changes made in the algorithm are only temporarily changes. In this case, each time the algorithm returns to step 4 or step 1, the data is the same as at the beginning. The real changes are only made in the third step of the method (the comparison).

1. Consider two feeder services, F_1 and F_2 in the cluster.
2. Determine all noncentral ports that are served by the feeder service F_1 .
3. Determine all consecutive port combinations on feeder service F_2 .

4. Repeat the following steps as long as not all combinations of a noncentral port and a consecutive port combination are considered.
 - (a) Select a combination of a noncentral port N and a consecutive port combination (P_1, P_2) .
 - (b) Remove port combination (P_1, P_2) from feeder service F_2 and add the combinations (P_1, N) and (N, P_2) to F_2 . Furthermore, remove port N from feeder service F_1 .
 - (c) Determine the new loads on and capacities of feeder services F_1 and F_2 .
 - (d) Determine the increase in profit obtained by adding port N between ports P_1 and P_2 on feeder service F_2 .
 - i. If the increase in profit is higher than the highest increase found earlier, save F_1, F_2, N, P_1, P_2 and the new highest increase in profit.
5. If all combinations of two feeder services in the same cluster are considered, then stop. Else, return to step 1.

3.4.4. Comparison

After the first two steps, both the most profitable reduction in the size of a feeder service and the most profitable exchange of a port between feeder services are known. Furthermore, the amount of increase in profit is known for both changes. Note, that the increase in profit can also be negative, which corresponds to a decrease in profit (loss), because it is not checked in the first two steps whether the increase in profit is positive or not. First check which increase in profit is the highest, that of the reduction in size or that of the port exchange. If the increase is positive, make the changes that corresponds to the increase. So, if the highest increase in profit is obtained by a reduction in the size of a feeder ship, reduce the saved feeder ship by one size and reallocate the demand necessarily to make this reduction possible. However, if the highest increase is obtained by a port exchange, remove the saved port from the first saved feeder service and add this port between the saved combination on the second saved feeder service. Finally, if a profitable change is made, go back to the first step, otherwise the feeder network cannot be improved further using this method, so stop.

Note that reducing the capacity on a feeder service or exchanging a port between feeder services influences the effect of future changes in the feeder network. For example, when the capacity on a feeder service is reduced, it might not be possible to add an additional port on that feeder service anymore, because the increase in flow over the feeder service would exceed the new lower capacity. Thus, if we do not make the right change in first instance, it is possible that we block some very profitable changes in the feeder network. Therefore, we consider first all possible changes, before we make the most profitable one and start over again. However, changes on different clusters can be performed at the same

time, because they are independent of each other. Thus, the algorithm is performed for each cluster separately.

3.4.5. Add ports to main route

Next, we investigate whether it is profitable to add some ports to the main route and thereby reducing the size of the feeder routes serving those ports. At the moment, only the central port of a cluster is visited on the main routes of the route network. All other ports are served by a feeder service. However, some noncentral ports exist, which also have a large amount of cargo handling movements. Now the flows are known, it can be calculated whether it is profitable to visit these noncentral ports on one of the main routes. A part of the cargo flows from and to these ports can then be transported over the main routes. This diminishes the flow on the feeder service networks, which can reduce the costs of the feeder network. Ports can be visited both on one or more main routes and on a feeder route.

The method is performed once before and once after the method to decrease the feeder network. When it is performed before reducing the feeder network, the exact feeder costs are not yet known. Therefore, in this case only the decrease in feeder (un)loading cost are seen as cost reduction, where also the capital, operating and fuel costs are considered when the feeder network is already decreased.

The next algorithm gives a brief description of the method used to investigate whether ports should be added to main routes or not. In this algorithm, changes are only made in step 6. So, changes in other steps of the algorithm are only temporarily. When the algorithm returns to a previous step, the changes are undone. In Electronic Supplementary Material E the method is explained in more detail.

1. Consider a main route and determine the clusters that are visited on that main route.
2. Consider one of those clusters.
3. Determine which (noncentral) ports that belong to the cluster are not yet visited on the considered main route.
4. Determine the consecutive port combinations on the main route for which at least one of the ports belongs to the cluster.
5. Repeat the following steps as long as not all combinations of noncentral ports and consecutive port combinations are considered.
 - (a) Select a combination of a noncentral port N and a port combination (P_1, P_2) .
 - (b) Remove port combination (P_1, P_2) from the main route and add the combinations (P_1, N) and (N, P_2) to the route.

- (c) Determine the new loads on the main route and on the feeder service serving port N after reallocating as much cargo from and to port N as possible to the main route.
 - (d) Determine the increase in profit obtained by adding port N between ports P_1 and P_2 at the main route. If the increase in profit is higher than the highest increase in profit obtained earlier, save the new highest increase, the considered cluster and main route and ports N, P_1 and P_2 .
6. In this step a port is finally added to a route.
- (a) Consider first all clusters and all routes and add thereafter the most profitable port to the main route.
 - i. Return to step 2 as long as not all clusters are considered.
 - ii. Return to step 1 as long as not all routes are considered.
 - iii. Add the most profitable port at the most profitable place to the main route if the increase in profit is larger than 0.
 - iv. If a profitable change is made in step 6(a)iii, return to step 1, else stop.

3.5. Designing a new route network

Even if only a few ports are included in the problem, a lot of different route networks can be constructed. Since it is impossible to generate and evaluate all possible networks and the performance of comparable networks will probably also be comparable, we use a genetic algorithm based method to change the route networks. The representation of a route in this study is a binary string consisting $R(2C - 1)$ 0/1 elements, where R is the number of routes in a network and C the number of port clusters found after the aggregation phase. So, $2C - 1$ are the possible stops of a ship on a route. When the element corresponding to port cluster $c \leq 2C - 1$ and route $r \leq R$ has value 1, port cluster c is called on route r . Furthermore, the capacity of the ship that will serve the route is stored in the representation. A route network then corresponds with a set of routes.

The following operators are used to change the route network: elitism, crossover and mutation. In the elitism step the best route network(s) of the current iteration are selected and are placed unchanged in the network set of the next iteration. Elitism ensures that the performance of the best network in the next iteration cannot be decreased in comparison to the best network in the current iteration.

For the crossover and mutation operators, we will use roulette wheel selection to select networks from the current set. Two different crossover methods are used: uniform and route crossover. In both crossover method two existing networks are selected and changed into two new networks. In the uniform crossover methods, each 0/1 value of the strings in each route of the network is with equal probability exchanged between

the two selected networks or not exchanged. Similar, the capacities of the routes are with equal probability exchanged or not. Finally, the routes are made feasible using the method earlier described and it is checked whether the ship types allocated to the routes in the networks still satisfy the available fleet constraint. In the route crossover method, existing routes are not changed, but only exchanged between the selected route networks. We randomly select a cut point, after which the routes including capacities of the two selected networks are exchanged. All routes that occur in the new route networks that are created using the route crossover method are always feasible, because they are unchanged according to the routes in the current iteration. However, the fleet composition of the networks can violate the available fleet constraint. Thus, we check whether this happens and exchange route with allocated ships where necessarily.

Finally, the mutation method changes the value of some elements. When a route network is selected, some elements corresponding to a route and a port cluster are selected at random. The selected ports are added to the route when they are not visited on the current route. On the other hand, when the ports are visited on the current route, they are deleted from the route. Furthermore, the feasibility of the route has to be checked, because the mutation operator can make routes infeasible. The capacity of a certain route will also be changed with a certain probability. The new capacity will then be randomly chosen from the existing, still available, capacities. When the capacity of the routes can also be changed, more feasible route networks will be considered.

4. Case study

We apply the proposed algorithm to design a service network that consists of 58 ports on the Asia-Europe trade lane. Most data is based on the service network of Maersk on the Asia-Europe trade lane during spring 2010. It is difficult to obtain real data, so demand is estimated using port throughput data and data obtained from the annual reports of Maersk. Furthermore, costs for different types of ships and sailing speeds are estimated using data obtained from Lachner & Boskamp (2011), Francesetti & Foschi (2002) and Notteboom (2006). The data can be found in Electronic Supplementary Material A.

4.1. Reference network

We define a reference network to which the best obtained networks (i.e. the networks with highest profits) can be compared. In the reference network, no feeder services are included. Comparing our best networks to a reference network will give some information on the performance of the obtained networks. The routes of the reference network can be found in Electronic Supplementary Material B.

The original Maersk route network, on which the data is based, is used as reference network. The network consist of nine routes. On each route a few ships with different

capacities are sailing to serve the demand. In our study, a route is served by ships of the same size and each ship size is a multiple of 1000 TEU. Therefore, we round the average capacities of the ships used on a route of the Maersk service network to the nearest multiple of 1000 TEU that is larger than the average to obtain the capacities of the reference network. The profit of the reference network can be determined after using the speed-optimization method and the cargo-routing model and is 1.707 billion USD. However, the capacities of the ships sailing on the reference network are fixed as explained above, while the liner shipping company is allowed to choose ships from an available set in our method. Thus, a comparison between our method and the reference network is not completely fair. Therefore, we used a simple improving heuristic to improve the allocated ships on the reference networks, such that the fleet satisfies the same restrictions as imposed in our method. In this heuristic, we first determine the optimal ship capacity on each route of the reference network, without taking the fleet restrictions imposed in this research into account. Thereafter, we make the allocation of ships to routes feasible by reducing the size of the ships on the routes for which the profit decrease is smallest. Finally, we repeatedly try to reduce the ship capacity on one of the routes until this is not possible anymore without causing a decrease in profit. The improved network has a profit of 1.772 billion USD, which is already an increase of 3.8% compared to the reference network with fixed capacities.

4.2. Characteristics of the best networks

Table 1 first shows the overall characteristics of the reference network and the best networks obtained using the described method with different numbers of cluster. We performed our methods with different number of clusters obtained from the aggregation step in order to investigate the importance of choosing the right number of clusters. Less clusters will result in lower computational time of the cargo-routing model, but a larger feeder network. Thus, by comparing different numbers of clusters, a trade off can be made between performance and computational time. The best network is found when including only ten clusters and results in a profit of 1.953 billion USD, which is an increase of 14.4% compared to the reference network and an increase of 10.2% compared to the improved reference network. The increase in profit is the result of a cost reduction. The total demand delivered and the revenue in the best networks are lower than in the reference network, but the routes are more cost efficient. Note that we only consider ship-related costs and revenues in this model. In reality, liner shipping companies will also face other types of costs. However, it is difficult to estimate these non-ship-related costs, so we leave them out of consideration here. Furthermore, we only use estimates on demand, revenue and costs in this research. Since these data are highly sensitive to fluctuations, the increase in profit can change over time.

	Reference network	8 clusters	10 clusters	12 clusters	15 clusters
Profit in billion USD	1.707	1.910	1.953	1.927	1.837
Revenue in billion USD	7.015	5.797	6.516	6.750	6.620
Cost in billion USD	5.307	3.895	4.563	4.823	4.783
Fraction demand delivered	0.832	0.669	0.739	0.793	0.777
Computational time in seconds	215.7	88.6	103.9	135.3	210.7
Main network					
Capital and operating cost in billion USD	1.150	0.840	0.952	1.077	1.039
Fuel cost in billion USD	1.453	1.064	1.174	1.338	1.316
Port cost in billion USD	0.189	0.140	0.160	0.211	0.194
Handling cost in billion USD	1.972	1.585	1.753	1.880	1.843
Transshipment cost in billion USD	0.544	0	0.100	0	0.035
Total liner cost in billion USD	5.307	3.630	4.139	4.506	4.426
Fleet size	91	64	71	80	79
Number of routes	9	6	7	7	7
Average number of ports per route	16.4	18.0	17.7	19.3	21.3
Distance traveled in nm	191,754	136,041	145,719	164,680	163,266
Feeder network					
Capital and operating cost in million USD		40.200	59.925	48.300	51.750
Fuel cost in million USD		35.153	61.520	43.399	41.648
Port cost in million USD		21.840	27.300	26.520	25.740
Handling cost in million USD		167.849	275.950	199.383	237.297
Transshipment cost in million USD		0	0	0	0
Total feeder cost in million USD		265.042	424.694	317.602	356.434
Fleet size		12	16	14	15
Number of routes		12	16	14	15
Average number of ports per route		3.3	3.2	3.4	3.2
Distance traveled in nm		21,469	27,555	25,729	22,259
Fraction of cargo on feeder network		0.148	0.220	0.149	0.180

Table 1: Characteristics of the best networks

The lower part of Table 1 shows some characteristics of the main and feeder routes of the best networks and the main routes of reference network. First note that compared to the reference network, the number of ships decreases in the best network with ten clusters, because the use of a feeder network leads to a more efficient use of ships. In the best network, only seven routes are included compared with nine in the reference network. Since, the number of ships needed for a route equals the round tour time, this will already lead to a significant reduction in the fleet size. The feeder services are all designed in such a way that they will have a round tour time of one week, so only one ship per feeder service is needed. Furthermore, the ships on the feeder network are in general smaller

than the ships on the main network. Therefore, the capital, operating and fuel costs on the feeder network are much lower than on the main network. Thus, by using smaller ships for the feeder network, a cost reduction in the ship costs will be obtained. The best networks with twelve and fifteen clusters have a larger fleet than the reference network, but because the ship costs on the feeder network are lower, the total ship costs of these networks are still lower than for the reference network.

Note that the total feeder costs of the best network with twelve clusters are lower than those corresponding to the best network obtained when using fifteen clusters. One would expect the total feeder costs to decrease when the number of included clusters increases as less ports have to be served on the feeder network. However, from Table 1 we can conclude that in both case 48 port visits are made on the feeder network ($15 \cdot 3.2 = 48$ and $14 \cdot 3.4 \approx 48$). For instances with more clusters, the initial main routes will probably contain more ports and thus more port combinations. Then, the average load on a route is also probably higher, which makes it more difficult to add noncentral ports to the main routes. Furthermore, less cargo can be allocated to the main route when a noncentral port is added to that route. Therefore, the total amount of cargo transhipped over the main routes is larger, which increases the feeder handling costs. We can indeed observe from the table that more handling costs are incurred in the instance with fifteen clusters than in that with twelve clusters.

Similarly, the total feeder costs of the best network with eight clusters is lower than the feeder costs in all other best networks. This time however, the fraction of demand delivered is also much lower than in the other networks. With less clusters, the initial routes are smaller and some ports are not visited at all on the best networks. Therefore, less cargo has to be transported over the feeder network and consequently less feeder costs are incurred.

It follows from Table 1 that a large cost reduction is obtained in the transshipment cost compared to the reference network. By clustering ports, less ports are included in the main route network and thus less transshipment movements are needed. However, additional handling costs on the feeder network are incurred, but in total the handling costs decrease, also since less demand is satisfied on the best networks than on the reference network.

In the table, also the fraction of cargo that is transported over the feeder network is given. Since each container can be transported twice over the feeder network (once from the origin port to the central port of the origin cluster and once from the central port of the destination cluster to the destination port), this fraction can vary between 0 and 2. The fraction of cargo transported over the feeder network gives some indication on the average transit times of containers in the network. Cargo that is only transported over the main network will in general have lower transit times than cargo that has to be

transhipped to the feeder network. Furthermore, the total distance travelled on the main network combined with the fleet size also give an indication of the average transit times in the network. If routes are shorter, less ships are needed and the travel times between two ports on the same route will on average be lower than when larger routes are used. Finally, the transshipment costs also give an indication of the average transit time. If more transshipment moves are needed, the transit times will generally be higher. Combining these three indicators, we can conclude from the table that the best networks obtained with eight and twelve clusters probably have lower average transit times than the best networks obtained with ten and fifteen clusters. In the networks with eight and twelve clusters, no transshipment moves take place on the main network and the fraction of cargo that is transported over the feeder network is also significantly lower than for the other networks. Furthermore, the network with eight clusters also has the smallest fleet size and distance travelled, so this network will probably perform best with respect to average transit time.

The computational times are given in seconds for the reference network and in seconds per iteration for the best network. In one iteration, the profit of all twenty route networks in the set is determined and a new set is made. The composite algorithm is stopped after eight hours of computational time in which an average improvement of 12% is found compared to the best initial network. The best network with ten clusters has both the highest profit and one of the lowest average computational times per iteration. Therefore, performing the methods with ten initial port clusters is preferred in this study. Furthermore, it can be seen that including too many clusters can have a negative effect on the performance. From Table 1, it follows that the computational time per iteration of the method increases when more port clusters are considered. Thus, less iterations can be performed in the maximum computational time and the profit of the best network obtained with fifteen clusters is 5.9% lower than the highest profit found with only ten clusters.

4.3. Best network

The routes of both the main and the feeder network of the best obtained route network can be found in Electronic Supplementary Material D. The reference network only consists of a main route network, so all ports are at least visited once on a main route. However, not all ports are visited on a main route in the best network. First, the ports in Japan are not visited anymore on the main route. Many ports in China are visited on a main route, but ships have to cross the ocean to visit ports in Japan. Because China is the turning point of the route, the crossing distance should be covered twice when ports in Japan are included in the main routes. Therefore, it is quite logical that the ports in Japan are only visited on the feeder services.

Many ports in Southern Europe are located in a cove, so that a lot of additional distance has to be covered to visit these ports on the main routes. Again, it is then logical that these ports are not visited on the main routes, but are fully served by the feeder network, since smaller and cheaper ships are used on the feeder routes.

Finally, some ports, for example Taipei, are not visited on the main route because they are relatively small. The additional distance that has to be covered to visit these ports is not very large (Taipei is located near Kaohsiung, which is visited on some main routes). However, since they are small ports, the additional costs incurred by adding them to a main route, will probably not be covered by the decrease in feeder costs.

Ports are mainly visited in geographical order on the routes in the best network found. However, sometimes small deviations from the geographical order are observed. These deviations are caused by the cargo flows. For example, on one of the routes first Ningbo is visited and thereafter respectively Qingdao, Busan, Xingang, Dalian and Shanghai. All these ports belong to the cluster with central port Shanghai. The geographical order would be Busan-Dalian-Xingang-Qingdao-Shanghai-Ningbo. Thus, for example, the location of Ningbo in the obtained route deviates from the geographical order. This can be explained in the following way. The method that adds ports to the main routes determines the best location to place a port on the existing route. The best location is defined as the location where the highest increase in profit can be obtained. The additional distance that has to be sailed is part of the decision, because additional costs are incurred when more distance has to be covered. However, the optimal place to add a port to the main route depends also on the reduction of the costs that can be obtained. A cost reduction can be obtained by reallocating cargo to the main routes, such that transshipment costs to the feeder services are saved. We have seen that the amount of cargo that can be loaded in the added port on the main route is bounded when a port is placed before the central port of the cluster, while the amount of cargo unloaded from the main route is bounded when the port is visited after the central port of the cluster. Thus, although the optimal geographical location of Ningbo would be after Shanghai, it can be more profitable to visit Ningbo before Shanghai when much cargo with destination Ningbo is on the ship.

4.4. Uncertain demand

The best networks are found using a given origin-destination demand matrix, while the reference network is probably constructed with other (uncertain) demand data. Thus, the comparison between the reference network and the best networks found in this case study, will not be completely fair. Therefore, we created ten instances with randomly generated origin-destination demand data, where each origin-destination pair has demand between 80% and 120% of the given demand matrix. For each of these ten instances, we solved the cargo-routing model for the reference network and for the best network (with ten clusters).

The average demand of the ten test cases is 0.2% higher than in the original scenario. The reference network has on average an increase in profit of 0.9% with respect to the reference network with given demand over these ten instances, while the profit increase of the best network is on average 0.8%. Also, the standard deviation of the profit does not differ that much: 1.0% for the reference network against 1.4% for the best network. These figures indicate that the best network is a bit more volatile for changes in the demand, but they do not suggest that the method is very sensitive to the used demand matrix.

5. Conclusion

In this study methods to solve the combined fleet-design, ship-scheduling and cargo-routing problem are developed. Thereto, an aggregation method is proposed, which aggregates ports into port clusters to reduce the size of the problem. Thereafter, initial route networks are constructed and a linear programming formulation (cargo routing model) is introduced that can be used to solve the cargo-routing problem to optimality. The number of clusters is chosen based on the computational time of the model. In our study, eight to fifteen clusters are appropriate to work with. The design of the clusters is based on the geographical location of the ports.

After the results are obtained in clustered ports, they have to be disaggregated again in individual port results. Some methods are developed and explained in this study. In these methods, a distinction is made between main services and feeder services. The feeder services are used to transport the cargo from the cluster centers to the other ports in the cluster. In first instance, only cluster centers can be part of the main service network. However, other ports are added to the main routes when this is profitable. Furthermore, we try to decrease the ship capacities on the feeder services in the proposed methods.

The above methods can be used to determine the profit of a certain route network. Then, methods are given to change existing networks into new networks, which can again be solved using above methods. We considered different numbers of port clusters obtained from the aggregation method, to investigate the importance of selecting the right number of clusters. When less clusters are considered, the average computational time per iteration reduces and the performance increases. The best results are obtained when ten port clusters are included, while including fifteen clusters results in more than 5% less profit in our case study.

The best network found using the overall model in our case study gives an improvement of 14.4% compared to the reference network. However, in the reference network, the used fleet is fixed, while we construct a new fleet under some restriction in our methods. Therefore, we also compare the best network to the reference network with improved ship capacities on the routes. The profit of the best network is 10.2% higher than the

profit of this reference network with improved fleet. The increase in profit is the result of a cost reduction obtained when integrating liner shipping network design. However, since we only consider ship-related costs and revenues and the data is highly sensitive to fluctuations, the profits of the liner shipping company for both the reference network and the best network will be lower in reality. The percentage of improvement will then also be lower.

The reference network only consists of a main service network. Therefore, all ports are visited on at least one of the main routes. However, in the best obtained network in our case study not all ports are visited on a main route anymore. There are basically three possible reasons for this. When the last ports before a turning port are noncentral ports in a network, the distance from the central port to these ports has to be covered twice when these ports are added to a main route. Thus, these ports can probably better be visited on a (cheaper) feeder route. Furthermore, some ports are located in a cove. The additional distance that has to be traveled to visit these ports can therefore become large. In this case, it is probably more profitable to serve these ports on feeder routes instead of a main route. Finally, some ports have very little demand. For these ports, the additional costs of visiting these ports on a main route are higher than the maximum reduction in costs that can be obtained. Therefore, these ports can also better be visited on a feeder route.

In our case study, the order in which the ports are visited on the main routes in the best obtained network corresponds most of times to the geographical order. Some deviations can be found, because ports are afterwards added to the main routes. The amount of cargo that can maximally be (un)loaded in a port that is added to a main route depends on whether the port is added before or after the central port of the cluster. Therefore, it can be more profitable to add a port after the central port of the cluster, even when the geographical order implies that the port should be added before the central port and vice versa.

Finally, the reference network is probably constructed with other (uncertain) demand, so we constructed ten scenarios with uncertain demand and compared the difference in performance of the reference network and the best obtained profit. In our study, the reference network and best network react comparable to changes in demand for the scenario, so this does not seem to be a large issue.

Only intra-regional demand is considered in this study, but it is possible to add also regional demand in the model. The idea behind the methods will stay the same when regional demand is included. The regional demand will not be considered in the methods discussed in the improvement steps. Because the revenue of the regional demand will be relatively low compared to intra-regional demand (because the distance between origin and

destination is much smaller), this will hardly influence the performance of the methods.

Acknowledgements

The authors want to thank Erasmus Smart Port BV for providing financial support for this research.

References

- Agarwal, R., & Ergun, O. (2008). Ship scheduling and network design for cargo routing in liner shipping. *Transportation Science*, *42*, 175–196.
- Álvarez, J. (2009). Joint routing and deployment of a fleet of container vessels. *Maritime Economics & Logistics*, *11*, 186–208.
- Berbeglia, G., Cordeau, J., Gribkovskaia, I., & Laporte, G. (2007). Static pickup and delivery problems: A classification scheme and survey. *TOP*, *15*, 1–31.
- Christiansen, M., Fagerholt, K., & Ronen, D. (2004). Ship routing and scheduling: Status and perspectives. *Transportation Science*, *38*, 1–18.
- Fagerholt, K. (1999). Optimal fleet design in a ship routing problem. *International Transactions in Operational Research*, *6*, 453–464.
- Fagerholt, K. (2004). Designing optimal routes in a liner shipping problem. *Maritime Policy & Management*, *31*, 259–268.
- Francesetti, D., & Foschi, A. (2002). The impact of hub and spokes networks in the mediterranean peculiarity. *IAME Annual Conference Panama*, .
- Gelareh, S., & Pisinger, D. (2011). Fleet deployment, network design and hub location of liner shipping companies. *Transportation Research Part E*, *47*, 947–964.
- Lachner, S., & Boskamp, V. (2011). Routing and scheduling in liner shipping with multi-start local search heuristics. Master’s thesis, Erasmus University Rotterdam.
- Lawrence, S. (1972). *International sea transport: the years ahead*, Lexington Books, Lexington, Mass.
- Nagy, G., & Salhi, S. (2005). Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. *European Journal of Operational Research*, *162*, 126–141.

- Notteboom, T. (2006). The time factor in liner shipping services. *Maritime Economics & Logistics*, 8, 19–39.
- Powell, B., & Perakis, A. (1997). Fleet deployment optimization for liner shipping: an integer programming model. *Maritime Policy & Management*, 24, 183–192.
- Rana, K., & Vickson, R. (1991). Routing container ships using lagrangean relaxation and decomposition. *Transportation Science*, 25, 201–214.
- Ronen, D. (1983). Cargo ships routing and scheduling: Survey of models and problems. *European Journal of Operational Research*, 12, 119–126.
- Ronen, D. (1993). Ship scheduling: The last decade. *European Journal of Operational Research*, 71, 325–333.
- Song, D., Zhang, J., Carter, J., Field, T., Marshall, J., Polak, J., Schumacher, K., Sinha-Ray, P., & Woods, J. (2007). On cost-efficiency of the global container shipping network. *Maritime Policy & Management*, 32, 15–30.
- UNCTAD (2010). United Nations Conference on Trade And Development, Review of maritime transport 2010, Geneva.
- Wang, S., & Meng, Q. (2011). International maritime-port technology and development conference (mtec 2011). chapter Liner Ship Routing and Scheduling: State of the Art. (pp. 69–74).