Cargo allocation and vessel scheduling on liner shipping with synchronization of transshipments

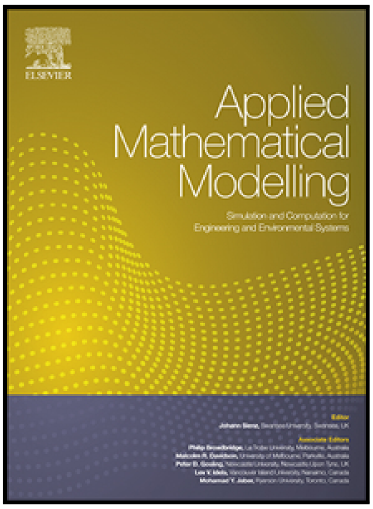
Ozcan, Sel; Eliiyi, Deniz Türsel; Reinhardt, Line Blander

Published in:
Applied Mathematical Modelling

DOI (link to publication from Publisher):
10.1016/j.apm.2019.06.033

Publication date:
2020

Document Version
Accepted author manuscript, peer reviewed version



Cargo Allocation and Vessel Scheduling on Liner Shipping with Synchronization of Transshipments

Sel Ozcan , Deniz Türsel Eliiyi , Line Blander Reinhardt

PII: S0307-904X(19)30395-6
DOI: https://doi.org/10.1016/j.apm.2019.06.033
Reference: APM 12901

To appear in: *Applied Mathematical Modelling*

Received date: 25 September 2018
Revised date: 18 June 2019
Accepted date: 25 June 2019

Please cite this article as: Sel Ozcan , Deniz Türsel Eliiyi , Line Blander Reinhardt , Cargo Allocation and Vessel Scheduling on Liner Shipping with Synchronization of Transshipments, *Applied Mathematical Modelling* (2019), doi: https://doi.org/10.1016/j.apm.2019.06.033

Highlights

- A mathematical model for the joint cargo allocation and vessel scheduling problem is developed.
- A benders-decomposition algorithm as well as valid inequalities are proposed to solve the problem.
- The model reflects the practical dynamics with realistic estimations for the arrival/departure times of vessels.
- The problem is tested over 24 realistic instances.
- The results indicate the proposed model helps decision makers to examine different solution alternatives for each shipment.

Cargo Allocation and Vessel Scheduling on Liner Shipping with Synchronization of Transshipments

Sel Ozcan [a,*], Deniz Türsel Eliiyi [a] , Line Blander Reinhardt [b]

[a] Izmir Bakircay University, Department of Industrial Engineering, Gazi Mustafa Kemal Mahallesi, Kaynaklar Caddesi Seyrek, Menemen, Izmir, Turkey

[b] Aalborg University, Department of Mechanical and Manufacturing Engineering, A.C. Meyers Vænge 15 København SV, Denmark

[*] Corresponding Author. e-mail: sel.tatari@bakircay.edu.tr (Sel Ozcan). Tel: +90 232 493 00 00

ABSTRACT

A mixed integer linear programming model is presented for the operational level cargo allocation and vessel scheduling problem of a liner shipping company in Turkey, where flow-dependent port-stay lengths, transit times and transshipment synchronizations are considered. The proposed model aims to assign shipments to routes to decrease total tardiness and construct partial vessel schedules for establishing coordination with port authorities to comply with the berthing time windows. In addition to the mathematical model, novel valid inequalities and benders decomposition algorithm are implemented. Performance of the developed algorithm is evaluated on real-life problem instances. The results show that benders decomposition with valid inequalities yields the best performance.

Keywords Liner shipping; Cargo allocation; Vessel Scheduling; Transshipment; Benders decomposition.

1. INTRODUCTION AND MOTIVATION

With over 80% of world merchandise trade being carried by sea, maritime transport is the backbone of international trade and globalization. As it is clearly stated in the recent review of maritime transport by the United Nations Conference on Trade and Development (UNCTAD), global seaborne shipments have increased by 4% in 2017, and the 2018-2023 projection indicates that this percentage will increase up to 3.8% per year. Furthermore, total containerized trade volumes are estimated at 148 million twenty-foot equivalent units (TEUs) [1].

Among the various seaborne transportation alternatives for containers, liner shipping is preferred mostly due to its cheaper freight rates, higher safety level and less environmental hazard [2]. A liner shipping company (LSC) has a fleet of different vessels deployed on its services, where a predetermined number of homogeneous vessels are operating on each service at regular frequencies. Each service makes round trips and visits predetermined ports in a fixed sequence. In most cases, the shipment of a container through the liner shipping network may include the use of several services to reach from its origin to its destination port. As a result, many containers need to be transshipped. Due to the competitive business environment, LSCs try to provide efficient cargo routing solutions, i.e., an increase in the vessel utilization, by improving service design and/or fleet deployment. At the same time, LSCs work on achieving effective cargo routing solutions through on-time delivery [3].

When the freight route is planned, various critical factors need to be taken into account simultaneously. Although there are some studies [4, 5, 6] having multi-objective optimization models in which both vessel scheduling and schedule recovery are studied, the most common objective is to perform the transportation with minimum cost. On the other hand, when the sole objective is the on-time delivery of the shipments, which is especially crucial for highly competitive businesses, objective becomes minimizing the *transit time*, i.e., the time it takes to travel from the origin to the destination. For example, for perishable or time-sensitive products having economic/technical depreciation (fashion, computers, etc.), where shorter transit times are strictly enforced, the second type of objective is more relevant [7, 8, 9].

In general, the LSCs prepare their routing plans in terms of shipments. A *shipment* is defined as a bundle of consolidated containers, each having the same characteristics such as origin-destination (O-D) pair and the desired transit time. The route of a shipment may include either a sub-path of a single service or a combination of multiple sub-paths of multiple services, requiring transshipments. As manufacturers aim to minimize their inventory holding costs, they prefer to send their just-finished products to an LSC as shipments. These shipments are stored temporarily by the LSC until the destined vessel for the container arrives at the origin port [3]. In addition to the storage times of the shipments at the origin port, the waiting times at the transshipment ports required for connections to other services on the route should also be considered for routes having multiple sub-paths.

For products requiring short transit times, the LSC may prefer the route having the minimum transit time, and if this route includes transshipments, the synchronization of the connections, i.e., schedule coordination, plays a critical role on timely delivery. For instance, if the path of a shipment requires transshipment at port A from vessel i to vessel j, synchronization between the departure of vessel i from port A, and the arrival of vessel j to port A will yield minimum transshipment time. If synchronization is not achieved by the two vessels, the shipment should either be stored at port A until the next vessel of the same service as vessel j arrives, or the shipment should be rerouted. Both yield serious costs to the LSC.

On the other extreme, for shipments having loose deadlines, the LSC may prefer to send through alternative routes in order to increase its services' profit. These alternative routes usually include sub-paths of the underutilized vessels. Such shipments can be sent without any delay despite their waiting times at the origin and/or transshipment ports. In many cases, it is more profitable for the LSC to store or transfer such containers at the ports having low demurrage costs.

As stated in [10], as many alliances were established in order to avoid underutilized vessels operating on transatlantic and transpacific routes, LSCs within these alliances started to determine the best ship size to deploy for these routes. As a result, the market shares of the smaller LSCs started to diminish. Therefore, the small LSCs should seek alternative ways to increase their market shares. Although vessel scheduling is a tactical level decision, the number of TEUs loaded/unloaded have a huge impact on the port stay durations and influences the arrival/departure times of the vessels on an operational basis. Therefore, it is vital for especially small LSCs to link their shipment assignment and vessel scheduling decisions in a systematic way to increase its long-term profitability and schedule reliability.

In this study, we are motivated by our real-life case partner, the liner shipping agency in Izmir, Turkey, which mainly operates on the Mediterranean, West Africa and the Black Sea. Our partner has a relatively small market share and tries to improve its businesses. We focus on a demand flow problem with transshipments on a given (fixed) liner shipping network, considering flow dependent port stay lengths and transit times, in addition to transshipments and arrival time constraints. With the proposed mathematical model, we aim to assign the shipments to the routes to decrease the total tardiness of the shipments, and to construct the partial schedules of the vessels to facilitate the LSC's coordination with the port authorities for the berthing time windows.

Our study has the following differences from the existing literature:

1. Different than most existing studies, our study allows late arrivals of shipments to the destination port. As opposed to the studies of [3, 11], where the time spent at each port is assumed to be fixed, our proposed formulation calculates the actual schedule of the vessels on each service by computing the port durations of each vessel. We believe that taking non-fixed port stay durations will better reflect the practical dynamics of the liner shipping industry. This is realistic since the time spent at each port is a function of the number of TEUs handled, hence cannot be fixed through the entire planning horizon.
2. Except for the cases where unplanned rare events, such as long delays due to labor strikes or hurricanes, which lead to major deviations from the planned schedules of the vessels; the real arrival and departure times of the vessels can have minor deviations from the planned schedules due to the variable port durations and uncertainties at ports. Our formulation therefore provides more realistic estimations for the arrival and departure times, which will help the information flow between the LSC and the port authorities prior to berthing.
3. We enforce every candidate shipment within the planning horizon to be sent via a candidate route. We assume that the LSC do not reject sending any shipment. In many real-life cases, due to fierce market competition among small LSCs and the low vessel utilizations, small LSCs prefer to accept all shipment requests. Hence, our problem environment differs from the existing cases in the literature.

4. Our study helps the decision maker to examine different solution alternatives for each shipment, which was not considered in the literature previously. Although the LSC may prefer to send the shipments via the route having the minimum transit time, storing some of the shipments at a transshipment port for some days at no cost can decrease the total cost of the LSC in some cases. Such a strategy may also end up decreasing the total tardiness of the shipments planned. This flexibility is allowed in our solution approach.

Due to the aforementioned differences, this study aims to contribute to the shipment planning literature via the development of a more realistic model. The remaining of the paper is organized as follows: Section 2 contains reviews of the related literature. Section 3 introduces the notation used, explains the mathematical formulation in detail and defines the problem through an illustrative example. The solution framework, as well as the strengthening constraints and the Benders decomposition algorithm are presented in Section 4. Section 5 reports the computational experiments. Finally, conclusions and future research opportunities are discussed in Section 6.

2. LITERATURE REVIEW

The decision-making problems on liner shipping can be categorized into three levels. At the *strategic level*, an LSC makes long-term decisions such as ship fleet size and mix, strategic alliances, and network design. At the *tactical level*, frequency determination of services, fleet deployment, speed optimization and schedule construction are considered. On the *operational level*, cargo allocation (i.e., cargo assignment), cargo booking and rescheduling decisions are taken into consideration. It is necessary to state that the problems examined at different decisions levels are interrelated within each other [12, 13].

Cargo routing/allocation problem seeks for the decision of which cargo to accept or reject for shipping, and which route(s) to use for shipping the selected cargo [12]. Although the cargo routing problem is treated as an operational level problem, the planning horizon may contain a strategic or tactical planning period by considering monthly or weekly cargo flows. They are similar to the multi-commodity network flow (MCNF) problems; the formulation of routing in the MCNF problem can be either origin-destination-based (O-D-based) link-flow [12, 14] or path-based flow, where each variable indicates a path for a certain commodity [9, 14, 15, 16, 17].

Transshipment and transit time considerations are rarely studied in the cargo routing and assignment problems. To the best of our knowledge, the first study considering transshipment operations in liner shipping is presented by Agarwal and Ergun [12]. They propose a MILP model to solve the integrated ship scheduling and cargo assignment problems simultaneously. In addition, they present a column-generation-based heuristic and a two-phase Benders-decomposition-based algorithm to solve these two problems separately on tactical level, rather than operational. The proposed model does not consider the transshipment costs while designing the service routes. Unlike our study, there is no transit time restriction in their formulation. They prove that the decision version of the simultaneous ship scheduling and cargo assignment problem is NP-complete by reducing the problem into the well-known 0-1 Knapsack problem [12].

Karsten et al. [9] formulate the cargo routing problem as a time-constrained MCNF problem by employing a maximum transit time for each commodity on tactical level. They examine the trade-off between the reductions on bunker cost versus offering shorter transit times for commodities. Without considering the design of the network, they only focus on how cargo should flow through the network. Moreover, they assume constant transshipment times, transshipment costs and constant sea durations at each leg. Their two-phase solution methodology includes the generation of the routes in the first phase, and the decision of how much cargo should be transported through these routes in the second phase. After proposing both arc-flow and path-flow formulations, they indicate that path-flow formulation performs faster than the arc-flow formulation in the delayed column-generation algorithm.

Transit time considerations have been usually incorporated in liner shipping network design problems rather than the cargo routing problems in literature. For example, as a tactical level study, Wang and Meng [18] present a non-linear MILP for the network design problem, taking the transit times into account. They formulate a column generation-based heuristic for solving the problem for a Europe-Asia network with 12 ports and assume predetermined port rotations without considering transshipments.

There are a limited number of studies where scheduling decisions of vessels are taken into consideration. Wang and Meng [16] present a liner ship route schedule design model that focuses on determining the arrival time of a ship at each port call and the sailing speed on each leg, by considering uncertainties during sailing and port calls. A tactical level mixed-integer non-linear stochastic programming model is developed with the objective of minimizing the ship cost and expected bunker cost while satisfying a required transit time service level. They assume that a container route is either a part of one particular ship route or a combination of several ship routes for delivering containers from the origin to destination. An exact cutting-plane based solution algorithm is proposed as a solution methodology.

Wang et al. [3] consider the liner shipping route schedule design problem and decide the arrival and departure times at each port call of the route. They assume that the ports are available within the defined time windows of a week, and these should be taken into account while constructing the schedule of the vessels. Other assumptions indicate that a port can be visited at most twice in a week on the ship route, and a ship can only be served by one berth. These assumptions implicitly define the set of possible arrival days in a week at the port of call. However, as the schedule of a single service is optimized, transshipments are not considered in this study. The problem is formulated as a tactical level, mixed-integer non-linear non-convex optimization model, and an efficient holistic solution approach is proposed to reach global optimality.

Recently, Reinhardt et al. [19] propose a speed optimization problem on liner shipping by adjusting the berth times of vessels on tactical level. Their study covers both transit time restrictions and transshipment times at ports. The proposed model ensures that the overall transit times for the cargo is retained. They define the minimum required time for transshipment of a container as *Connecting Time Windows (CTW)*, usually measured from the departure of a vessel, from which the container is unloaded, to the arrival of another vessel onto which the container is loaded. In this case, a crane first unloads the container from one vessel. The container is then transported by a truck to another crane, which performs the loading operation of the container on another vessel. This arrangement of operations is referred as *hot berthing*, which may transpire when a container is transshipped from one vessel to another. The authors claim that this configuration is especially useful when the berth locations of the two vessels are close to each other.

A chance-constrained optimization model is developed by Wang et al. [20] which simultaneously attempts to determine the optimal fleet capacities, cargo allocation and vessel route schedules through schedule coordination. In addition to the weekly deterministic demand coming from the contracted customers, they consider the daily spot demand. They construct a state-augmented shipping network where each task on the container shipment activities is represented by a unique link. They emphasize the requirement of an efficient solution methodology for this problem which will be able to solve large networks in a reasonable computation time and their study is on tactical level, rather than operational.

Guericke and Tierney [7] study the cargo routing problem with service levels and leg-based speed optimization. They allow multiple port calls of vessels on a single service. Similar to the study of Karsten et al. [9], the transshipment times, transshipment costs and the transit time restrictions are employed, and a path-flow formulation is utilized. In order to decrease the complexity of the problem, the total number of available paths for each O-D pair is assumed to be limited. As they assume half of the vessel is loaded and unloaded at each port visited, the studied problem is at the tactical level. Neither of these studies [7] and [9] simultaneously decide the arrival and departure times of the vessels.

Different than the liner shipping studies discussed so far, Wang et al. [21] present a connectivity reliability-cost approach for path selection in maritime transportation of crude oil. The authors introduce an uncertainty variable defining the connectivity reliability of each node in the case of extreme events, and an uncertain bi-objective programming model with connectivity reliability maximization and transportation cost minimization objectives is formulated for path selection. Moreover, Güven and Türsel Eliiyi [22] propose a recent study for the increase in the port performances by reducing the unproductive movements of a container through online stacking policies.

In addition to approaches including mathematical formulations, Wang and Yeo [23] present an integrated Fuzzy Delphi and Fuzzy (Elimination Et Choice Translating Reality (ELECTRE) I method for finding the best route for transporting cargo from Korea to Central Asia under China's Silk Road Economic Belt. Firstly, for the intermodal route selection, the hierarchy within the factors is achieved through clustering opinions from experts using Fuzzy Delphi method. Secondly, the Fuzzy ELECTRE I is applied for the evaluation of the five alternative routes. The results show that, total cost is the most important factor, and the others are followed by reliability, transportation capability, total time, and security, respectively.

Different than the cargo routing problems on liner shipping, there are also several studies concentrated on the vessel scheduling problem, in general. Chuang et al. [24] propose a fuzzy genetic algorithm where uncertainty in market demand, shipping and berthing time of the vessels are handled with fuzzy sets theory. Brouer et al. [25] introduce a novel formulation for the Vessel

Schedule Recovery Problem (VSRP) and it is the first to apply optimization techniques for disruption management in liner shipping. In this study, they evaluate a given disruption scenario and try to select a recovery action with the consideration of the bunker consumption, the impact on cargo in the remaining network and the customer service level. Alharbi et al. [26] address a mixed integer, non-linear, non-convex model for the liner shipping schedule design problem considering port time windows. The problem is reformulated as an integer linear optimization model and an iterative solution framework is presented. Different than existing studies in literature where the liner shipping service is served by a predetermined number of homogeneous vessels, Dulebenets [27] proposes a novel mixed integer non-linear mathematical model for the vessel scheduling problem with heterogeneous fleet to minimize the total vessel turnaround cost. Similarly, Gürel and Shadmand, [28] study the vessel scheduling problem with a heterogeneous fleet implying different fuel burn functions on vessels. Their formulation also incur different port capacities, demands, and many other characteristics, which yield different handling and waiting times at ports.

There are also recent studies where vessel scheduling and speed optimization decisions are considered simultaneously. Dulebenets and Ozguven [29] propose a novel mixed integer non-linear mathematical model for the vessel scheduling problem in a liner shipping route with perishable assets where the objective is to minimize the total route service cost. They consider variable sailing speed and employ the linear secant approximation method for the approximation of the bunker consumed. They adopt an exponential decay function to capture deterioration of perishable assets on board the vessels over time. Fagerholt et al. [30] study the joint speed optimization and routing problem for a single vessel in the case of Emission Control Areas (ECAs), where the objective is to minimize the fuel cost. They determine which leg options to use between the ports (i.e. the sailing path across the sea) and analyze whether the ship operators prefer to sail longer distances in order to decrease the sailing within the ECAs. Similarly, Qi and Song [31] address vessel scheduling problem in liner shipping with the objective of minimization of total expected fuel consumption where port-related uncertainties are taken into account. In addition, a recent study addresses a mixed integer non-linear mathematical model for the vessel scheduling problem, where a novel collaborative agreement is proposed between the liner shipping companies and port authorities with the objectives of total vessel operational cost, total fuel consumption cost, total port handling cost, total container inventory cost, and the total late arrival penalty [32].

Our study differs from the existing papers in the literature as it focuses on the joint vessel scheduling and cargo allocation problem as a demand flow problem where transshipments are allowed, and the transit time constraints of shipments as well as schedule coordination restrictions are considered. In this respect, our study concurrently determines the port stay lengths and maintains schedule coordination with the port authorities.

3. THE SHIPMENT ASSIGNMENT AND VESSEL SCHEDULING PROBLEM

In this section, the assumptions and the formulation of the Shipment Assignment and Vessel Scheduling Problem (SAVSP) is explained in detail. First, the problem environment and the notation are described.

We consider an LSC that operates mainly on the Mediterranean, West Africa and the Black Sea, where all its services, denoted by the set S, provide a regular frequency, i.e. daily or weekly. The set V represents the vessels operating on services. We assume that a subset of homogeneous vessels, denoted by V_s are operating on each service s, $V_s \subseteq V$. The set P includes all feasible routes (path) and we define the subset P_b, $P_b \subseteq P$, for each shipment b, where the route $p \in P_b$ includes the ordered set of ports visited through by vessel v. The first port on each route p represents the origin port of the corresponding shipment, whereas the last port indicates the destination port of that shipment. In addition to the origin and destination ports, loading/unloading operations are performed at the transshipment ports. There is no capacity restriction on the arcs of each service. That is, no restriction exists on the amount of shipment transported through each leg.

In most liner shipping networks, a port may be visited several times through the vessel's round trip. In order to differentiate these unique vessel-port pairs, we replicate some ports depending on the total number of visits of the corresponding vessel during its round trip and generate a *modified* service network. For a better understanding, consider the illustrative example in Figure 1 (a). Assume that the shipment will be sent from the origin port *Thessaloniki*, *Greece* (GRSKG) to the transshipment port *Marport*, *Turkey* (TRMRP) using vessel 2 and then from TRMRP to the destination port *Koper*, *Slovenia* (SIKOP) using vessel 1. To represent the aforementioned unique vessel-port subsets for this specific route, denoted as p, we first replicate the ports in the visiting sequence and then relabel them with unique numbers. As depicted in

Figure 1 (b), the double-circled ports with the corresponding vessels are included in the subset for route p. Afterwards, the unique vessel-port pairs on the network are defined. The origin port GRSKG, for example, is represented with 6 in Figure 1 (c) and the ordered subset of vessel-port pairs for route p are constructed as $\{(2,6),(2,7),(2,8),(1,8),(1,9),(1,10)\}$ where, for instance, $(2,6)$ corresponds to vessel 2 and port 6 and $(2,7)$ corresponds to vessel 2 and port 7 etc. (see Figure 1 (c)).

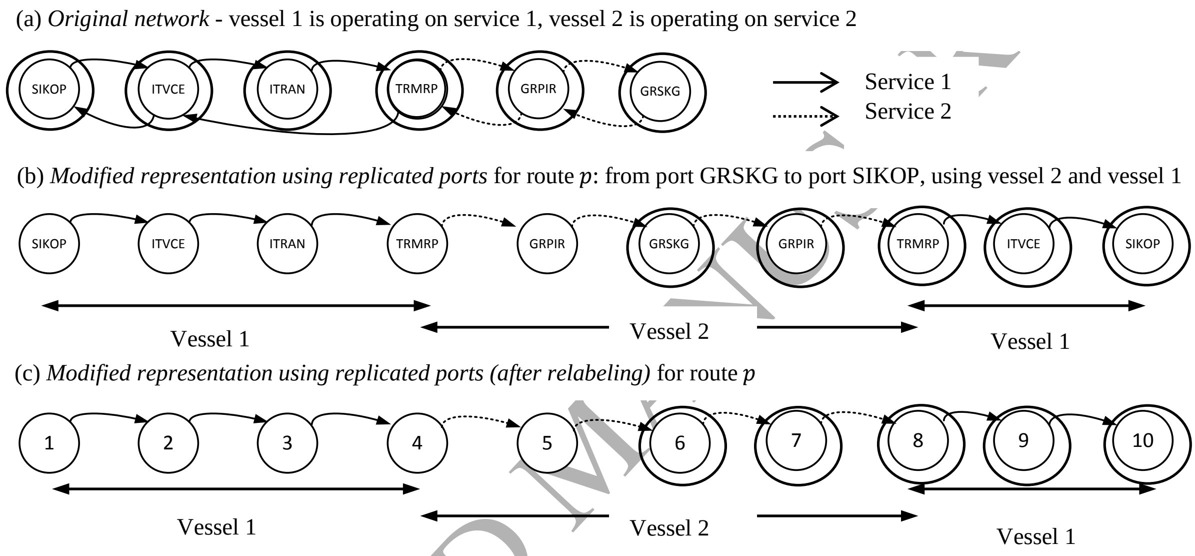


Figure 1. Representation of unique vessel-port pairs for route *p*

3.1. An Illustrative Example

For a better understanding of how the mathematical model behaves, we provide an illustrative example where we compare the two scenarios to show that the shortest path strategy is not always the best choice. Consider an LSC having two services, namely service 1 and service 2. Figure 2 depicts this small network. Service 1 has the port rotation of *Koper-Venice-Ravenna-Istanbul-Venice-Koper*, and service 2 has *Istanbul-Piraeus-Thessaloniki-Piraeus-Istanbul*. Assume that there is a fixed number of homogeneous vessels operating on each service, and the services operate on a 3-day frequency. That is, the difference of the departure times of vessel 1 and 2 from *Koper*, and of vessel 3 and 4 from *Istanbul* is 72 hours. In a 15-day planning horizon, at least two homogeneous vessels operate on each service. To simplify calculations, we assume that the sea durations between ports are deterministic and fixed at 48 hours. In addition, the port stays of vessels depend on the number of TEUs loaded/unloaded; the departure time can be determined by adding the current port stay to the port arrival time, whereas the arrival time to the next port can be found by adding the leg duration to the departure time.

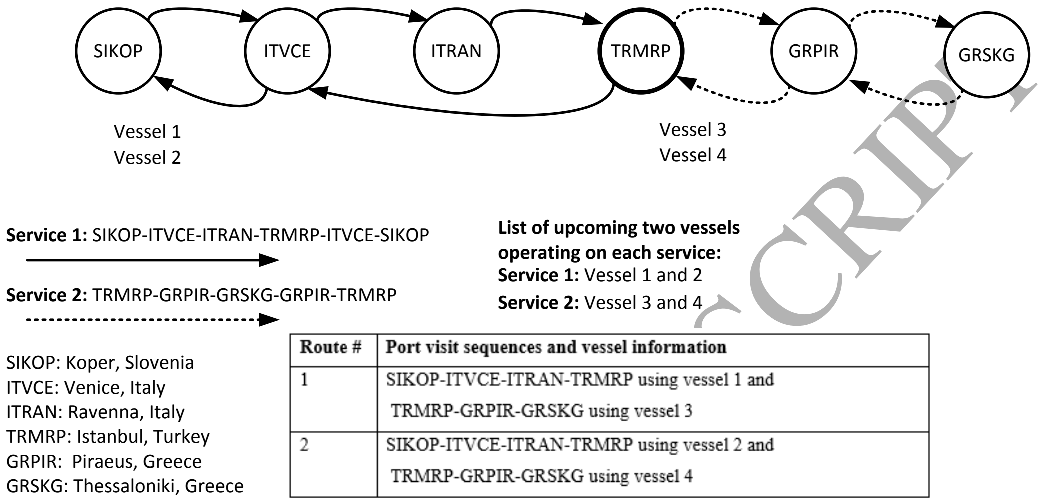


Figure 2. An illustrative example of two services, four vessels and two routes

The liner shipping company should decide on which route the shipment will be sent to minimize the total tardiness of shipments. By selection of any route, the arrival and departure times of the vessel(s) operating on that route will also be determined. We elaborate on two alternative scenarios below to clarify the impact of port duration on tardiness.

In the first scenario, assume that the shipment from *Koper* to *Thessaloniki* has a quantity of 100 TEUs and a desired delivery time of 360 hours. For the sake of simplicity, consider the case where no loading/unloading operation takes place at the intermediate ports. Namely, the port stay durations at any port except the origin, destination or transshipment ports, is zero. Also assume a constant loading/unloading time for this shipment, e.g. 4 hours.

Figure 2 indicates two feasible routes, namely route 1 and 2, for this shipment. Assume that route 1 is selected. In order to provide on-time delivery, vessel 1 arrives to the origin port *Koper* at time zero. After 4 hours of loading it departs from the port, and after 48 hours of sailing the vessel arrives at port *Venice* at time 52. Following the same reasoning, the vessel arrives at port *Ravenna* and *Istanbul* at times 100 and 148, respectively. The unloading operation from vessel 1 starts immediately and finishes after 4 hours at time 152. Next, the loading of vessel 3 starts without delay and finishes at time 156. Vessel 3 departs from port *Istanbul* at time 156 and arrives at ports *Piraeus* and *Thessaloniki* at times 204 and 252, respectively. The total transit time is thus calculated as 256. As the transit time is less than the desired delivery time, the shipment is delivered on-time.

Due to the 3-days frequency enforcement of each service, the arrival and departure times of the unused vessels can be derived easily. For example, for vessel 2, the arrival time to port *Koper* will be 72, and the arrival time of vessel 4 to port *Istanbul* will be 320 (see Table 1).

Table 1. Arrival and departure times of vessels for route 1 (Scenario 1).

		Arrival times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
1	SIKOP	0	72		
	ITVCE	52	120		
	ITRAN	100	168		
	TRMRP	148	216	152	224
	GRPIR			204	272
	GRSKG			252	320
		Departure times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
1	SIKOP	4	72		
	ITVCE	52	120		
	ITRAN	100	168		
	TRMRP	152	216	156	224
	GRPIR			204	272
	GRSKG			**256**	320

As an alternative route for this shipment, assume that route 2 is selected (see Figure 2). In this case, the port stays of vessels 1 and 3 become zero, and the arrival and departure times of vessels 2 and 4 are summarized in Table 2. Due to the 3-days frequency, vessel 2 arrives at port *Koper*, and after 4 hours of loading time it departs from *Koper* at time 76. Since vessel 2 has no planned loading/unloading operation at ports *Venice* and *Ravenna*, the arrival and departure times are identical. Vessel 2 arrives at *Istanbul* at time 220 and finishes with the unloading operation at time 224, while vessel 4 arrives at the same time and departs at time 228 after 4 hours of loading. After 2 days at sea, vessel 4 arrives at *Piraeus* at time 276, and departs without delay. As a last leg, vessel 4 arrives at the destination port *Thessaloniki* at time 324, and the total transit time of this shipment is calculated as 328.

Table 2. Arrival and departure times of vessels for route 2 (Scenario 1).

		Arrival times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
2	SIKOP	0	72		
	ITVCE	48	124		
	ITRAN	96	172		
	TRMRP	144	220	144	224
	GRPIR			192	276
	GRSKG			240	324
		Departure times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
2	SIKOP	4	76		
	ITVCE	48	124		
	ITRAN	96	172		
	TRMRP	144	224	144	228
	GRPIR			192	276
	GRSKG			240	**328**

As in the solution of route 1, the shipment is not delayed when route 2 is selected. Hence, these two solutions are indistinguishable in terms of the tardiness objective.

In the second scenario we consider two shipments, the same shipment in the previous example and another shipment between the same O-D pair with a quantity of 400 TEUs and a desired delivery time of 305 hours. We assume a constant loading/unloading time at each origin, destination and transshipment port for shipments 1 and 2, as 4 and 16 hours respectively. When route 1 is used for both shipments, the transit time of the shipments will be 320 hours, since the total time spent at ports *Koper, Istanbul* and *Thessaloniki* is increased to 24 hours. It can be observed that the second shipment is delayed by 15 hours. The arrival and departure times of the vessels can be examined in Table 3.

Now consider the following solution, the shipment with the 360-hour deadline is sent through route 2 and the shipment with the 305-hour deadline is sent through route 1. Table 4 summarizes the arrival and departure times for this solution, and it can be seen that, with this second solution, the delay for both shipments is decreased to zero. The first shipment arrives at *Istanbul* at time 220 and should wait 28 hours for the arrival of vessel 4.

Table 3. Arrival and departure times of vessels for route 1 (Scenario 2).

		Arrival times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
1	SIKOP	0	72		
	ITVCE	68	120		
	ITRAN	116	168		
	TRMRP	164	216	184	256
	GRPIR			252	304
	GRSKG			300	352
		Departure times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
1	SIKOP	20	72		
	ITVCE	68	120		
	ITRAN	116	168		
	TRMRP	184	216	204	256
	GRPIR			252	304
	GRSKG			**320**	352

Table 4. Arrival and departure times of vessels for routes 1&2 (Scenario 2).

		Arrival times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
1 and 2	SIKOP	0	72		
	ITVCE	64	124		
	ITRAN	112	172		
	TRMRP	160	220	176	248
	GRPIR			240	300
	GRSKG			288	348
		Departure times			
Route #	**Port**	**Vessel 1**	**Vessel 2**	**Vessel 3**	**Vessel 4**
1 and 2	SIKOP	16	76		
	ITVCE	64	124		
	ITRAN	112	172		
	TRMRP	176	224	192	252
	GRPIR			240	300
	GRSKG			**304**	**352**

The illustrative example in this section indicates how port durations affect the delay of shipments, and how a solution without any delay can be achieved regardless of the waiting times at the origin and transshipment ports. In many real-world cases, it is more profitable for the LSC to store containers at interim ports having low demurrage costs. Moreover, although vessel scheduling is a tactical level decision, the number of TEUs loaded/unloaded have a huge impact on the port durations and hence, it influences the arrival/departure times of the vessels on an operational basis. Therefore, it is significant for the LSC to consider shipment assignment and vessel scheduling decisions in a systematic way so as to increase its profitability and schedule reliability.

3.2. The Mathematical Model

The proposed formulation assigns the shipments to the routes to decrease total tardiness, while concurrently constructing the partial schedules of the vessels to facilitate the LSC's coordination with the port authorities for the berthing time windows. The sets and parameters used in the formulation are defined below.

Sets:

B = set of shipments $b = 1,2, \ldots$

P = set of routes for all shipments $p = 1,2, \ldots$

S = set of services $s = 1,2, \ldots$

N = set of ports $n = 1,2, \ldots$

V = set of vessels $v = 1,2, \ldots 3$

L = set of legs $l = 1,2, \ldots$

P_b = set of candidate routes for shipment b, $P_b \subseteq P$

VN_p = set of vessel-port pairs that are visited through route p

$P_{(v,n)}$ = set of routes that the vessel-port pair (v, n) is included, $P_{(v,n)} \subseteq P$

L_p = set of legs on route p, $L_p \subseteq L$

TS_p = set of transshipment ports on route p, $TS_p \subseteq N$

B_v = set of shipments that are transported through vessel v, $B_v \subseteq B$

B_n = set of shipments that are either loaded or unloaded at port n, $B_n \subseteq B$

V_s = set of vessels operating on service s, $V_s \subseteq V$

Parameters:

H_p = the deadline of shipment b transported via route p, $b \in B, p \in P_b$

Y_p = the latest possible arrival time of shipment b transported via route p, $b \in B, p \in P_b$

T_p = the maximum required time between legs of route p to make the planned transshipment for shipment b, $b \in B, p \in P_b$

S_{pl} = the sailing time for leg l of route p, $p \in P, l \in L_p$

$HT_p^{(v,n)}$ = the handling time of the shipment transported through vessel v to port n on route $p, p \in P, n \in N_p$

F_{v_1,v_2} = the minimum required time for vessels v_1 and v_2 to maintain the frequency of service s, $v_1, v_2 \in V_s$

ε = a positive number, close to zero

$dest_p$ = destination port of route $p, p \in P$

M = a very large positive number

Below are the decision variables of our mathematical model.

Decision Variables:

$a^{(v,n)}$ = the arrival time of vessel v to port n, $p \in P_{(v,n)}$, $(v,n) \in VN_p$

$a_p^{(v,n)}$ = the arrival time of vessel v to port n on route p, $p \in P_{(v,n)}$, $(v,n) \in VN_p$

$d_p^{(v,n)}$ = the departure time of vessel v from port n on route p, $p \in P_{(v,n)}$, $(v,n) \in VN_p$

$a_p^{(v,[i])}$ = the arrival time of vessel v to the port, where the corresponding port is in the i^{th} sequence in the port call of the related service operating on route p, $p \in P_{(v,[i])}$, $(v,[i]) \in VN_p$

$d_p^{(v,[i])}$ = the departure time of vessel v to the port, where the corresponding port is in the i^{th} sequence in the port call of the related service operating on route p, $p \in P_{(v,[i])}$, $(v,[i]) \in VN_p$

q_p = delay of route p, $p \in P_b, b \in B$

$$x_p = \begin{cases} 1, & \text{if route } p \text{ is selected} \\ 0, & \text{otherwise} \end{cases}, \quad p \in P_b, b \in B$$

Note that, all time-related parameters and decision variables are in hours. Based on the above definitions, the MILP model for the SAVSP is as follows:

$$min \sum_{b \in B} \sum_{p \in P_b} q_p + \sum_{(v,n) \in VN_p} \varepsilon a^{(v,n)} \quad (1)$$

subject to

$$\sum_{p\in P_b} x_p \geq 1 \qquad b \in B \qquad (2)$$

$$d_p^{(v,[n])} + x_p S_{pl} \leq a_p^{(v,[n+1])} \qquad p \in P, (v,[n]), (v,[n+1]) \in VN_p, l \in L_p \qquad (3)$$

$$\sum_{b\in B} \sum_{p\in P_b} \left(HT_p^{(v,n)} x_p \right) + a_p^{(v,n)} \leq d_p^{(v,n)} + \left(1 - x_p\right) M \qquad p \in P,\ (v,n) \in VN_p \qquad (4)$$

$$\left(x_p\right) M \geq a_p^{(v,n)} \qquad p \in P,\ (v,n) \in VN_p \qquad (5)$$

$$Y_p - H_p x_p \leq q_p \qquad p \in P \qquad (6)$$

$$a_p^{(v_2,m)} - a_p^{(v_1,m)} \leq T_p x_p \qquad p \in P,\ \left((v_1,m),(v_2,m) \in VN_p\right) \ni m \in TS_p \qquad (7)$$

$$a_p^{(v_1,m)} - d_p^{(v_2,m)} \leq 0 \qquad p \in P,\ \left((v_1,m),(v_2,m) \in VN_p\right) \ni m \in TS_p \qquad (8)$$

$$a_p^{(v,n)} \leq a^{(v,n)} \qquad p \in P,\ (v,n) \in VN_p \qquad (9)$$

$$a^{(v_2,[0])} - a^{(v_1,[0])} \geq F_{v_1,v_2} \qquad v_1, v_2 \in V_s, s \in S \qquad (10)$$

$$a^{(v,n)}, a_p^{(v,n)}, d_p^{(v,n)} \geq 0 \qquad p \in P,\ (v,n) \in VN_p \qquad (11)$$

$$x_p \in \{0,1\},\ q_p \geq 0, \text{integer} \qquad p \in P \qquad (12)$$

The first term in the objective function (1) minimizes the total tardiness whereas the second term fine-tunes vessel schedules by forcing the arrival time of the vessel-port pair (v,n) to be equal to the maximum arrival time of the routes of the vessel-port pair (v,n). With this type of multi-objective function scheme, the main goal is to find the best assignment that gives the minimum total tardiness of the shipments and as a secondary goal the schedules of the vessels are determined. Constraint set (2) enforces that every shipment should be sent via a candidate route. Constraint set (3) controls the arrival and departure times of the vessels on every route. Constraint set (4) calculates the departure time of each vessel from each visited port. Constraint set (5) forces the arrival time of a vessel to any non-visited port to zero. The parameter M in (4) and (5) will be further explained in Section 4.1. Constraint set (6) calculates the tardiness of each shipment. Since our model allows late arrivals of shipments, tardiness becomes positive once a shipment misses its deadline. The minimum connection time required for the synchronization of the transshipment is achieved through constraint set (7). If the arrival time of the loading vessel of the shipment is at least T_p time units larger than the arrival time of its discharging vessel, the synchronization of the transshipment is achieved. Our formulation guarantees this minimum time allowance if the selected route includes a transshipment. Constraint set (8) ensures that, if a transshipment decision is given, the arrival time of the discharging vessel to the transshipment port should be strictly less than the departure time of the loading vessel. Constraint set (9) together with the objective (1) ensures that the arrival time of a vessel at each port equals the maximum of the arrival times of the shipments on the vessel. Constraint set (10) dictates the desired frequency for the operating vessels for each service. Finally, constraint sets (11)-(12) define the decision variables.

Calculating the partial vessel schedules and port stays

The largest arrival and departure times among the shipments on each route determine the arrival and departure times of the vessels operating on these routes. For a better understanding, consider the illustrative example in Figure 3 where the flow of vessels and shipments are represented on a timeline. We assume that there are two shipments, namely shipment 1 and 2, both have the origin as port 1, however their destination ports are different. Shipment 1 has to be unloaded at the next port call of vessel 1, i.e., at port 2, whereas shipment 2 will be dropped off at later port visits of vessel 1. The two shipments are planned to be sent through the routes 1 and 2, respectively.

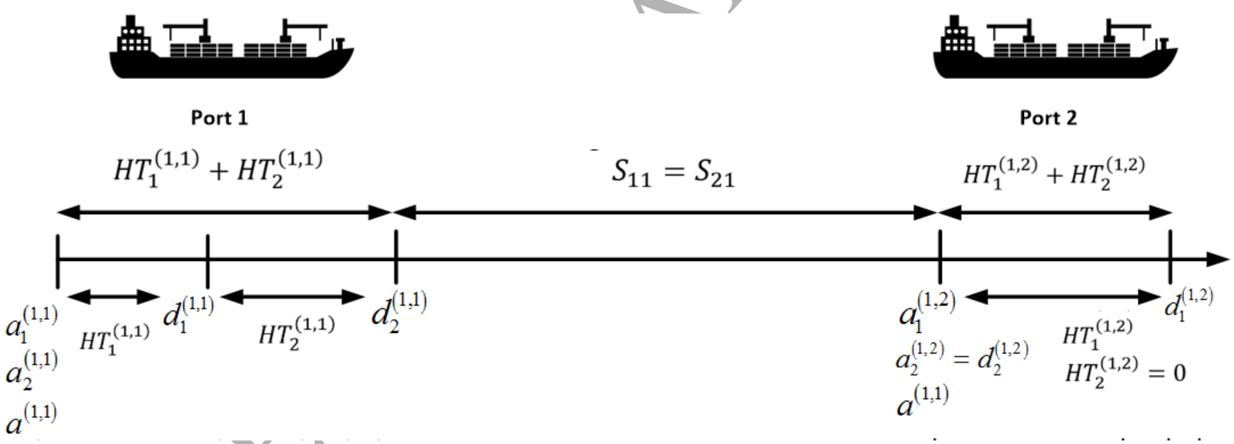


Figure 3. An example for the port stay, arrival and departure time calculations.

In this example, we assume that both routes use vessel 1 for transporting the two shipments. We assume that both shipments are ready before the vessel arrives, hence our formulation yields, $a_1^{(1,1)} = a_2^{(1,1)} = 0$, leading the arrival time of the vessel $a^{(1,1)}$ to be set to zero. Since the parameter $HT_p^{(v,n)}$ indicates the total time spent on route p at port n visited by vessel v, only for the ports the shipment is planned to be loaded or unloaded, the value of $HT_p^{(v,n)}$ has a positive value. Moreover, the values of the decision variables $d_p^{(v,n)}$ depend on the arrival time of the vessel and the total time spent at the corresponding port. Since we assume loading/unloading operations at a port can be performed one by one, the largest among the departure times of the routes, determines the departure time of the vessel v from port n. The arrival time of vessel v on route p to its next port call is determined by adding its departure time from the current port call to the constant sailing time for each leg l of route p, i.e., S_{pl}. For routes having the same vessel and port call, the sailing times should be equal to each other. Hence, S_{11} = S_{21} (see Figure 3).

We assume that, when vessel 1 arrives to port 2, the unloading operation of shipment 1 starts immediately. Since there is no unloading operation for shipment 2, $HT_2^{(1,2)}$ is zero, which yields to equal arrival and departure times of route 2 for the vessel-port pair $(1,2)$, i.e., $a_2^{(1,2)} = d_2^{(1,2)}$. As only shipment 1 will be unloaded at port 2, the corresponding port duration equals to $HT_1^{(1,2)}$ and vessel 1 will depart from port 2 after the unloading operation finishes i.e., at $d_1^{(1,2)}$. The partial vessel schedules are determined accordingly.

4. SOLUTION METHODOLOGY

An optimization algorithm with two phases is proposed for solving the SAVSP. In the first phase, all feasible routes are generated for each candidate O-D pair with a shipment during the planning horizon. A depth-first-search algorithm for searching the routes between any given O-D pair is implemented for this purpose. In order to obtain a search tree with a finite number of nodes, it is assumed that a port visited in one of the previous nodes of the tree cannot be visited again. Note that the number of routes is exponential in input size. The generated routes are used to construct the set of routes P as inputs of the SAVSP model proposed in Section 3. With this two-stage approach, the complexity gathered through routing is decreased. Note that the infeasible routes due to the limited number of the transshipment ports are also eliminated via this two-stage approach.

In the second phase, the model is solved and the best route for each shipment is determined. As the model includes a large number of variables, a group of tightening constraints are introduced in order to improve computational time performance. In addition, a Benders decomposition algorithm is developed for the problem. These are presented in the following subsections.

4.1 Introducing bounds and valid inequalities for the SAVSP model

Instead of assigning an arbitrary large value to the parameter M defined in constraint sets (4) and (5), we introduce two tight upper bounds.

$PS^{UB}_{(v,n)}$ is defined as an upper bound on the port stay length of vessel v at port n. The port stay length is maximized when the number of shipments (un)loaded at the same port from/to the same vessel is maximized. To calculate this upper bound for each vessel-port pair, the related port stay durations, $HT^{(v,n)}_p$ values are added, assuming that all shipments in vessel v are handled at port n. This leads to an upper bound as, $PS^{UB}_{(v,n)} = \sum_{p:\{p\in P_b, b\in B_v\}} HT^{(v,n)}_p$.

Another upper bound is introduced on the arrival time of vessel v to port n on the route p as $A^{UB}_{p,(v,n)}$. There are two components affecting the value of $A^{UB}_{p,(v,n)}$. The first is the sailing time of vessel v until reaching port n, i.e., $\sum_{l=1}^{l=n-1} S_{pl}$, where consecutive $n-1$ legs need to be sailed until port n. Let $SC_p = \sum_{l=1}^{l=n-1} S_{pl}$ be the constant related with the sailing time for route p. The second component includes the upper bound on the total port stay length for $n-1$ port visits, which was computed above. The second component is then expressed as $\sum_{k=1}^{n-1} PS^{UB}_{(v,k)}$. Hence, the upper bound becomes $A^{UB}_{p,(v,n)} = SC_p + \sum_{k=1}^{n-1} PS^{UB}_{(v,k)}$.

Based on these bounds, we replace constraint sets (4) and (5) with valid inequalities (4') and (5'), as below:

$$\sum_{b\in B}\sum_{p\in P_b}\left(HT^{(v,n)}_p x_p\right) + a^{(v,n)}_p \le d^{(v,n)}_p + PS^{UB}_{(v,n)}\left(1-x_p\right) \quad p\in P,\ (v,n)\in VN_p \qquad (4')$$

$$A^{UB}_{p,(v,n)}\left(x_p\right) \ge a^{(v,n)}_p \quad p\in P,\ (v,n)\in VN_p \qquad (5')$$

We also introduce $C^{LB}_{p,(v,n)}$ as a lower bound on the arrival time of vessel v to port n on the route p. This lower bound is composed of two parts. The first part includes the sailing time of vessel v until reaching port n, i.e., $\sum_{l=1}^{l=n-1} S_{pl}$. The second part is the lower bound on the total port stay length for the $n-1$ port visits. This latter part of $C^{LB}_{p,(v,n)}$ is computed by finding the minimum port stay on port n for every shipment as $\min_p\left\{HT^{(v,n)}_p\right\}$. Hence, vessel v will spend at least $(|B_v|)\min_p\left\{HT_p^{(v,n)}\right\}$ time units at port n, where $|B_v|$ represents the cardinality of set B_v. As a result, the lower bound $C_{p,(v,n)}^{LB}$ is computed as $C_{p,(v,n)}^{LB} = \sum_{l=1}^{l=n-1} S_{pl} + |B_v| \sum_{k=1}^{n-1} min_p\left\{HT_p^{(v,k)}\right\}$, and the following tightening constraint (13) is introduced into the model:

$$a_p^{(v,n)} \geq x_p C_{p,(v,n)}^{LB} \qquad p \in P,\ (v,n) \in VN_p \qquad (13)$$

4.2 Benders Decomposition Algorithm

As the number of paths and shipments are far above hundreds, solving the MILP model in (1) – (12) becomes increasingly difficult due to the large number of decision variables and constraints in the problem. In the computational experiments, we have observed that the computation time grows exponentially with the number of constraints in the problem. However, the MILP model in (1) – (12) can be decomposed in order to obtain a pair of problems (an LP subproblem and an IP master problem) that can be solved in a faster and more effective manner. We utilize the advantage of the boundedness of the LP subproblem, which provides only the optimality cuts to the IP master problem, hence yielding remarkable reductions on the computation time. In this section, we present a Benders decomposition algorithm for our model.

Benders decomposition (BD) [33] is an algorithm for solving MILPs with linking constraints and is preferred when the master problem has all the integer variables and it is difficult to treat them in subproblems [12]. When the integer variables are fixed, the original problem is decomposed into several LP subproblems, which iteratively generate optimality and/or feasibility cut(s) to the master problem. BD is usually preferred to reduce the number of variables at the expense of an increase in the number of constraints. Cordeau et al. [34] applied BD to simultaneously solve the aircraft routing and crew scheduling problems, while an alternative use of BD on power transmission network design problems is addressed by Binato et al. [35]. There are a few studies where BD is applied to the problems in liner shipping. As discussed earlier, Agarwal and Ergun [12] presented a BD-based algorithm to solve the ship scheduling and cargo assignment problems simultaneously. Moreover, Gelareh and Pisinger [36] addressed BD approach to solve the liner shipping network design and fleet deployment problem simultaneously. The interested reader is referred to a recent by study Rahmaniani et al. [37] for a comprehensive literature review on BD.

For given nonnegative values of $\bar{x}_p$ and $\bar{q}_p$ $(p \in P_b, b \in B)$ satisfying constraints (2) and (6), our model reduces to the following *primal subproblem* (PSP) including only the scheduling variables:

$$min \sum_{(v,n)\in VN_p} a^{(v,n)} \quad (14)$$

subject to

$$d_p^{(v,[n])} + \bar{x}_p S_{pl} \leq a_p^{(v,[n+1])} \qquad p \in P, (v,[n]),(v,[n+1]) \in VN_p, l \in L_p \quad (15)$$

$$\sum_{b\in B}\sum_{p\in P_b}\left(HT_p^{(v,n)}\bar{x}_p\right) + a_p^{(v,n)} \leq d_p^{(v,n)} + \left(1-\bar{x}_p\right)M \qquad p \in P,\ (v,n) \in VN_p \quad (16)$$

$$\left(\bar{x}_p\right)M \geq a_p^{(v,n)} \qquad p \in P,\ (v,n) \in VN_p \quad (17)$$

$$a_p^{(v_2,m)} - a_p^{(v_1,m)} \leq T_p\bar{x}_p \qquad p \in P,\ \left((v_1,m),(v_2,m) \in VN_p\right) \ni m \in TS_p \quad (18)$$

$$a_p^{(v_1,m)} - d_p^{(v_2,m)} \leq 0 \qquad p \in P,\ \left((v_1,m),(v_2,m) \in VN_p\right) \ni m \in TS_p \quad (19)$$

$$a_p^{(v,n)} \leq a^{(v,n)} \qquad p \in P,\ (v,n) \in VN_p \quad (20)$$

$$a^{(v_2,[0])} - a^{(v_1,[0])} \geq F_{v_1,v_2} \qquad v_1, v_2 \in V_s, s \in S \quad (21)$$

$$a^{(v,n)}, a_p^{(v,n)}, d_p^{(v,n)} \geq 0 \qquad p \in P,\ (v,n) \in VN_p \quad (22)$$

Let $\alpha_p^{(v,n)},\ \beta_p^{(v,n)},\ \gamma_p^{(v,n)}, \pi_p^{(v,n)}, \phi_p^{(v,n)},\ \delta_p^{(v,n)}, \theta^{(v,n)} \geq 0 \quad p \in P, (v,n) \in VN_p$ be the dual variables associated with constraints (15) – (21), respectively. Then, the dual of (14) – (21) is the following *dual subproblem* (DSP):

$$\max \sum_{b\in B}\sum_{(v,n)\in VN_p}\sum_{l\in L_p}\Bigg(\alpha_p^{(v,n)}\bar{x}_p S_{pl} + \beta_p^{(v,n)}\left(\left(1-\bar{x}_p\right)M - \left(HT_p^{(v,n)}\bar{x}_p\right)\right) + \left(\gamma_p^{(v,n)}M\bar{x}_p\right) - \left(\pi_p^{(v,n)}T_p\bar{x}_p\right)\Bigg) + \sum_{s\in S}\sum_{v:v1,v2\in V_s}\left(F_{v1,v2}\right)\theta^{(v,0)} \quad (23)$$

subject to

$$\alpha_p^{(v,[n+1])} - \beta_p^{(v,n)} - \gamma_p^{(v,n)} + \pi_p^{(u1,m)} - \pi_p^{(u2,m)} - \phi_p^{(u1,m)} - \delta_p^{(v,n)} \leq 0$$

$$p \in P, (v,n) \in VN_p\left((u_1,m),(u_2,m) \in VN_p\right) \ni m \in TS_p \quad (24)$$

$$- \alpha_p^{(v,[n])} - \beta_p^{(v,n)} + \phi_p^{(u,m)} \leq 0 \qquad p \in P, (v,n) \in VN_p\big((u,m) \in VN_p\big) \ni m \in TS_p$$

(25)

$$\delta_p^{(v,n)} + \theta^{(u2,[0])} - \theta^{(u1,[0])} \leq 1 \qquad p \in P, (v,n) \in VN_p, u_1, u_2 \in V_s, s \in S \qquad (26)$$

For given values of $\bar{x}_p$ and $\bar{q}_p$, there will always be a feasible schedule independent of the assignment of the shipments, since the null vector 0 satisfies constraints (23) – (26). Therefore, the PSP always yields bounded and feasible solutions. So does the DSP, and therefore only optimality cuts will be added to the *Benders master problem* (BMP).

The BMP is formulated as follows:

$$\min \sum_{b \in B} \sum_{p \in P_b} q_p \qquad (27)$$

subject to

$$\sum_{p \in P_b} x_p \geq 1 \qquad b \in B \qquad (28)$$

$$Y_p - H_p x_p \leq q_p \qquad p \in P \qquad (29)$$

$$x_p \in \{0,1\},\ q_p \geq 0, \text{integer} \qquad p \in P \qquad (30)$$

After partitioning the original model as BMP and DSP, the BD algorithm solves the two problems repeatedly. It first starts by solving the BMP to optimality, after which the lower bound of the original problem is updated with the objective function value of the master problem. Next, the DSP is solved by taking $\bar{x}_p$ and $\bar{q}_p$ values from the optimal solution of the BMP as input. At each iteration, a new constraint (Benders cut) extracted by the DSP is added to the BMP. Since there will always be a feasible subproblem, the sum of both the objective function value of the master problem and the objective function value of the subproblem provides an upper bound for the original problem. The algorithm terminates when the upper and lower bound of the original problem converge. The computational time performance gains of the BD algorithm are further discussed in Section 5.

5. COMPUTATIONAL STUDY

The two solutions, one obtained by the MILP solution by CPLEX 12.8.0, and the other found by the built-in BD algorithm available in CPLEX 12.8.0, are tested on the real-life service network example of an LSC. We employ two variants of BD via including and excluding the bounds introduced in Section 4.1 in the model, where the optimality gap is set to 10^{-6} and there is no limitation on the computation time, i.e., the global time limit is set to its default value 10^{75}. Various problem instances are generated to assess the performances of the solutions. We explain the characteristics of the test problems in Section 5.1 and discuss performances in Section 5.2.

5.1 Test Problems

Four different-sized networks of real data are used, provided by the shipping agency in Izmir, Turkey. The network sizes vary between 15 ports with 2 services and 39 ports with 8 services. The services have various frequencies, daily or weekly. The complete network of the LSC is illustrated on Figure 4.

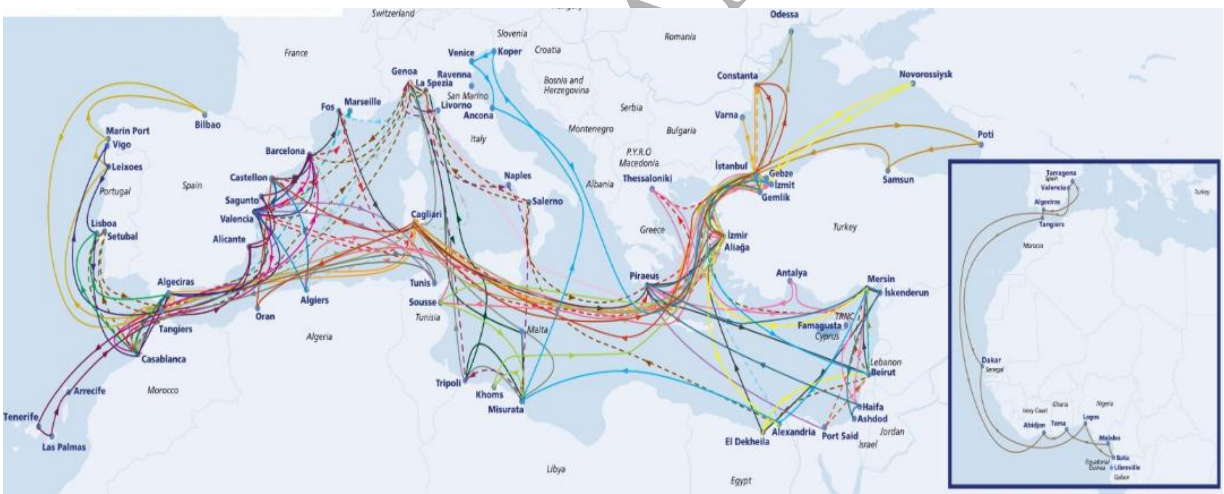


Figure 4. The complete network of the shipping agency

The characteristics of the network for each problem instance is summarized in Table 5. The main characteristics of the test problems are defined as follows: The *number of unique ports* is the union of the ports within the considered subset of service, whereas the *total number of ports* includes the duplicated ones. The *maximum number of ports visited on a single service* is an indicator on how the length of the voyages on each single service can vary. Although the *number of ports intersecting by any services* reflects the number of transshipment alternatives within the considered services from one aspect; the *Average number of ports on services* together with the *total number of unique ports* and the *Number of unique ports* creates another point of view on the complexity of the paths generated.

Table 5. Characteristics of the test problems.

Problem	Number of unique ports	Total number of ports	Number of services	Maximum number of ports visited on a single service	Number of ports intersecting by any services	Average number of ports on services
1	15	29	2	10	2	8
2	18	35	3	9	1	8
3	29	80	5	12	1	10
4	39	118	8	14	1	11

For the sake of simplicity, we assume similar-sized shipments. The generated data reflect up to 80% of the real O-D pairs of the mentioned shipping agency. Hourly average container handlings of each port are obtained and used to represent distinct port performances. In addition, we take individual transit times of all shipments as the current transit times, as suggested by the agency. Based on obtained data, the maximum transshipment time is taken as a full 24-hour working day. Due to privacy reasons, we provide only the sizes of our problem instances, and summarize their characteristics in Table 6.

Table 6. Characteristics of the scenarios tested.

Problem	Scenario	Limit on the number of routes	Number of Transshipments	Number of different O-D pairs	% covered of the considered real network
1	1	-	1	10	20
1	2	-	1	20	60
1	3	-	1	50	100
2	1	20	2	10	30
2	2	20	2	20	65
2	3	20	2	50	100
3	1	20	2	10	20
3	2	20	2	20	35
3	3	20	2	50	100
4	1	20	2	10	20
4	2	20	2	20	60
4	3	20	2	50	100

For each test problem, we tested our solution algorithms under three different scenarios. For the Problem 1, there is no limitation on the number of routes. There are only two services operating on the network for this instance, hence at most one transshipment can take place. For the Problem 1, we consider different number of O-D pairs as 10, 20 and 50, which respectively mimic 20, 60 and 100% of the real shipping network. For the remaining test problems, where the complexity of the network increases immensely, we limit the number of routes for each O-D pair to 20, which covers most dense and efficient routes on the network. On the other hand, the maximum number of transshipments is set to two, as suggested by the agency.

5.2 Computational Results

We first observe the performance of the MILP defined in (1) – (12). Afterwards, the valid inequalities explained in Section 4.1 are implemented and the computational performance of the strengthened MILP is reported. As both MILP variants yield low performances within the operational time limits, the BD algorithm is also implemented. All problems are solved using CPLEX 12.8.0, and the programming code is compiled by Java on a computer having a i7-5500 CPU @2.40Ghz processor and a 16.0 GB RAM. Details and examples of implementing BD in CPLEX can be found in Rudin [38]. Table 7 summarizes the computational time performances of all algorithms as well as percent improvements on the objective function value achieved through our formulation.

Table 7. Performances of the instances tested.

Test instance	Problem	Scenario	Number of different O-D pairs	Number of shipments for each O-D pair	# of feasible routes	$MILP_N$	$MILP_{VI}$	BD_N	BD_{VI}	% improvement
1	1	1	10	1	60	1.7	1.12	1.15	0.95	4.05
2	1	1	10	2	120	4.47	1.09	1.08	0.82	25.39
3	2	1	10	1	174	1.84	1.06	1.11	0.91	86.48
4	2	1	10	2	348	513.02	329.07	94.8	63.25	36.06
5	3	1	10	1	180	385.09	196.63	74.6	32.2	80.72
6	3	1	10	2	360	858.04	451.16	240.33	81.97	92.27
7	4	1	10	1	200	0.21	0.17	0.09	0.08	86.95
8	4	1	10	2	400	102.15	72.03	17.67	9.54	92.09
9	1	2	20	1	102	5.12	1.13	1.14	0.88	42.42

10	1	2	20	2	204	15.19	5.37	2.04	1.15	32.00
11	2	2	20	1	400	901.16	380.46	237.07	83.2	54.16
12	2	2	20	2	800	10126.81	8843.72	7910.63	5360.6	58.33
13	3	2	20	1	400	1323.32	395.68	257.77	87.26	92.16
14	3	2	20	2	800	10835.08	8499.59	7735.06	5026.4	94.43
15	4	2	20	1	388	366.63	203.24	81.07	44.99	88.18
16	4	2	20	2	776	11354.16	8778.09	6991.72	5870.2	82.80
17	1	3	50	1	264	628.15	475.03	129.32	94.98	0.53
18	1	3	50	2	528	103253.2	100125.8	9942.22	8614.4	0.26
19	2	3	50	1	982	130135.1	100744.1	9975.06	9065.8	1.82
20	2	3	50	2	1964	244174.6	225093.33	24349.1	21976	2.34
21	3	3	50	1	748	7010.54	5411.52	5980.15	2186.1	87.16
22	3	3	50	2	1496	347016.7	29554.43	29171.6	26315	93.30
23	4	3	50	1	986	422.05	269.41	82.08	52.19	91.14
24	4	3	50	2	1972	372590.4	34407.8	33308.8	30542	89.26

The first column in Table 7 enumerates the tested instances. Columns (2) – (4) are identical to those in Table 6. Column 5 represents the total number of shipments for each O-D pair, which is limited to 2 in our test problems. This implies that the maximum number of TEUs between each O-D pair is 100. Based on the analysis on the past demand data, this assumption is realistic for the agency. Column 6 shows the total number of feasible routes, i.e. the cardinality of set P. All test problems are solved to optimality by all algorithms. Column 7 presents the exact solution computation times (in seconds) for the SAVSP model in (1) – (12) ($MILP_N$). Columns 8 to 10 lists computation times for the model with the valid inequalities ($MILP_{VI}$), the BD algorithm without valid inequalities (BD_N), and the BD algorithm with the valid inequalities (BD_{VI}), respectively. Finally, the last column calculates the percent improvements gained through our formulation. Here, we compare our formulation's performance with the shortest path strategy. To do so, we select the shortest route alternative for each candidate shipment, i.e., the minimum transit time among all route alternatives for each shipment. Then, we compare the total tardiness of the shortest path strategy with the results of our formulation.

Note that, our formulation dominates the shortest path strategy in terms of the tardiness values, that is, it yields a smaller tardiness for each test instance. The impact of improvement increases with increasing network size, and the percent improvements are above 80% for 13 test instances. Although small percent improvements are obtained for some instances, their corresponding tardiness differences are quite large. For example, in the test instance 19, the total improvement on the tardiness value is 102 hours whereas the corresponding percent is 1.82%.

We also visualize the relative computational time deviations from the best, for all algorithms over the 24 test instances on Figure 5.

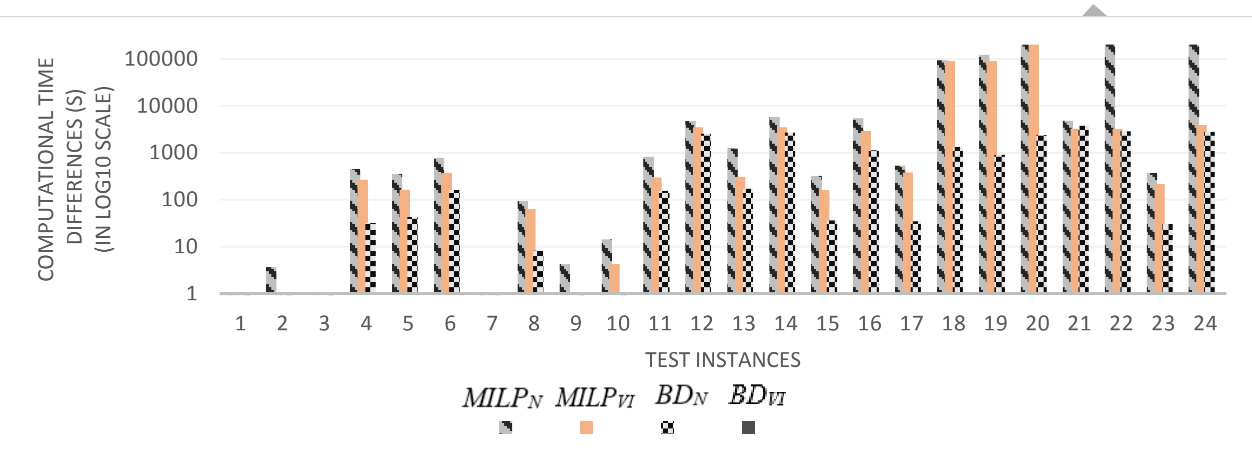


Figure 5. Relative performances of algorithms over the test instances

As expected, the computation time increases as the problem size increases, in parallel with the increase in the number of routes. Although each solution approach has a different characteristic, they act similarly on a quarter of the test instances. For example, for the test instances 1, 2, 3, 7, 9 and 10, where the number of feasible routes is below 204, the computational time performance of the algorithms is comparable. On the other hand, $MILP_N$ has the worst performance over all instances, implying that both the valid inequalities in (4'), (5') and (13), and the decomposition approach have a huge impact on the computational time performance.

The sole impact of the valid inequalities can be observed when $MILP_{VI}$ and $MILP_N$ are compared. $MILP_{VI}$ always outperforms $MILP_N$, meaning that the valid inequalities strengthen the formulation and tighten the polyhedron defined by the constraints (2) – (12). The major effect of the proposed valid inequalities can be observed in the largest test instances, namely instances 22 and 24, where the percent improvements on the runtime are 91% and 90%, respectively.

For the test instances 18 - 20, in which the number of different O-D pairs is set to 50, the impact of BD can be best observed. When $MILP_N$ and BD_N are compared, the percent improvements on the runtime for instances 18 – 20 are respectively as 90%, 92% and 90%. Similarly, the dominancy of BD_N over $MILP_{VI}$ for the same test instances is quantified with percent improvements on the runtime. 90%, 90% and 89% decrease in runtime are achieved for the same test instances when BD_N is employed instead of $MILP_{VI}$. Note that BD_N always outperforms $MILP_{VI}$ in terms of computational time performance.

A clear dominance of the BD_{VI} over the remaining three algorithms prevails. BD_{VI} achieved optimality on 15 test instances in less than 100 seconds, and its performance on the average is around 4800 seconds. The reason of BD_{VI}'s high performance is mainly due to elimination of the unboundedness in the dual subproblems through the introduction of the valid inequalities, thereby eliminating the need to add feasibility cuts to the master problem.

We report the average performances in Table 8. Note that the decrease in solution time with the valid inequalities and BD can be observed even for the case of 10 different O-D pairs. The improvement is particularly pronounced for the 50 O-D pair instances. On the average, all instances are solved to optimality in around 14 hours once $MILP_N$ is employed. The average computational time decreases to nearly 6 hours with the $MILP_{VI}$. Once BD_N is used, the average solution time reduces to less than 2 hours. The best performance with an average of 1.5 hours is obtained with the BD_{VI}.

Table 8. Average computational time performances of each algorithm.

Number of different O-D pairs	$MILP_N$	$MILP_{VI}$	BD_N	BD_{VI}
10	233.32	131.54	53.85	23.72
20	4365.93	3388.41	2902.06	2059.33
50	150653.83	62010.18	14117.28	12355.79
average (s)	*51751.03*	*21843.38*	*5691.07*	*4812.94*

Table 9 reports pairwise comparisons of each solution approach in terms of average solution times. The first and the last column in Table 9 indicate the sole performance of the proposed valid inequalities in (4'), (5') and (13), whereas the average performance of the BD can be observed from columns 2 and 5. By comparison of the first two columns, we can conclude that BD_N always outperforms $MILP_{VI}$. This implies that BD has relatively higher impact on computational time performance than the valid inequalities. Except the comparison of BD_N and BD_{VI}, the computational time gain is biggest for the largest number of O-D pairs, as expected.

Table 9. Average pairwise computational time comparisons of the algorithms tested.

Number of different O-D pairs	$MILP_N$ vs $MILP_{VI}$	$MILP_N$ vs BD_N	$MILP_N$ vs BD_{VI}	$MILP_{VI}$ vs BD_N	$MILP_{VI}$ vs BD_{VI}	BD_N vs BD_{VI}
10	43.62%	76.92%	89.84%	59.06%	81.97%	55.96%
20	22.39%	33.53%	52.83%	14.35%	39.22%	29.04%
50	58.84%	90.63%	91.80%	77.23%	80.07%	12.48%

We believe that our results indicate fruitful directions for managerial use. Our formulation will provide flexibility to LSC to adjust the routes and select the best alternatives in a broader spectrum. The optimal solutions indicate that the shortest route is not the best route for many cases, hence it is relevant and necessary to consider all feasible route alternatives.

6. CONCLUSION

In this study, the shipment assignment and vessel scheduling problem motivated by a liner shipping agency in Izmir, Turkey is considered. The problem is formulated as a novel mixed integer linear programming model and solved by a two-stage algorithm. The algorithm generates all feasible routes for each shipment in the first stage, while in the second stage the proposed model is solved optimally by taking the routes generated as input. Valid inequalities are proposed to restrict the feasible region and reduce the computational time. We also implemented a Benders decomposition approach for the problem.

The performances of all algorithms are tested on 24 problem instances, which are generated in line with the real practice based on past data of the liner shipping agency. The computational results indicate that the Benders decomposition algorithm including the proposed valid inequalities yields the best computational time performance, solving 21 of the test instances to optimality within 2.5 hours. The algorithm also achieves optimality in less than 100 seconds on 15 of the test instances, promising fast optimal solutions for real instances of the problem.

The study has two major contributions. Our MILP model allows the agency to determine which shipment will be sent through which route. Concurrently, our formulation determines the arrival and departure times of the vessels while maintaining a desired service frequency. We believe that our approach can handle the stochasticity of the port stays and sailing times by adjusting the arrival and departure times of the vessels and may provide an insight to the practitioners in order to redesign their routes for every O-D pair, as the solution suggests alternative routes for every shipment. The improvement achieved through our formulation is twofold. Our first contribution to the LSCs becomes apparent when the LSC's main objective is to increase customer satisfaction through increasing the schedule reliability. The LSC can further evaluate alternative routes for the shipments with loose deadlines, leading an increase in the usage of underutilized services. In addition, the LSC can evaluate various routes for each candidate shipment which, in fact, have alternatives other than the shortest ones. Doing so, the LSCs may achieve remarkable gains. We also believe that our solution framework can be practically used via a user-friendly interface.

Although the 2.5-hour time performance is manageable for the shipping company and the time horizon it would still be attractive to improve the computation time. Thus, as a future work, we plan to focus on strengthening the developed algorithm and compare it with more sophisticated and efficient existing algorithms that are more efficient than the shortest path.

A natural extension of this study lies in developing sophisticated heuristic algorithms for obtaining high quality solutions faster than the discrete optimization techniques applied in this study. Due to the high complexity of the proposed model, a realistic approach for practical use is to utilize heuristic approaches, providing near-optimal solutions in short computation time.

Another interesting extension of our current work is to enhance the proposed formulation by including speed optimization as well as bunker consumption decisions. LSCs are dealing with high fuel prices as well as concerns related with the greenhouse gas emissions and carbon footprint. The sailing speed of a vessel has a remarkable effect on bunker consumption, and the bunker cost accounts for a large proportion of the total operation cost of the vessel [39]. The bunker cost is estimated to be more than 60% of the total operating cost of a liner shipping company. Hence, slow steaming is preferred as it yields a reduction in the bunker consumption, i.e., paying less on bunker cost. On the other extreme, slow steaming may lead an increase in the delivery times, resulting in unattractive service times for the customers [19]. As a future study, we plan to investigate the problem of reducing the bunker consumption as well as maintaining the service quality, with the consideration of vessel capacities and utilizations.

An alternative approach can be to reformulate the problem through stochastic programming; where “act of god”, i.e., uncertainties such as whether conditions, congestions at ports or port strikes, which slash the container traffic; can be treated explicitly. To incorporate such uncertainties, we plan to address a the two-stage (planning and recourse) stochastic programming shipment assignment and vessel scheduling model. In the first stage, the model is solved according to the realized port performances. The solution is then revised in the later stage. For example, in the second stage, the LSC may reassign the shipments at intermediate ports in the route, when a delay is observed in the first stage. At the recourse level, the model will consider the shipment assignments made in the first stage, its realizations and the instant port performances at the adjustment point. The adjustment points will be the candidate transshipment ports that are going to be visited by the vessels through their upcoming port visits.

Acknowledgements

This work was supported within the scope of the scientific research project which was accepted by the Project Evaluation Commission of Yasar University under the project number BAP038 and title "Solution Approaches for the Integrated Liner Shipping Network Design and Container Terminal Operations".

REFERENCES

[1] UNCTAD, Review of Maritime Transport, 2018.

[2] S. Gelareh, D. Pisinger, Fleet deployment, network design and hub location of liner shipping companies, Transp. Res. Part E Logist. Transp. Rev. (2011). doi:10.1016/j.tre.2011.03.002.

[3] H. Wang, S. Wang, Q. Meng, Simultaneous optimization of schedule coordination and cargo allocation for liner container shipping networks, Transp. Res. Part E Logist. Transp. Rev. (2014). doi:10.1016/j.tre.2014.07.005.

[4] D.P. Song, D. Li, P. Drake, Multi-objective optimization for planning liner shipping service with uncertain port times, Transp. Res. Part E Logist. Transp. Rev. (2015). doi:10.1016/j.tre.2015.10.001.

[5] M.A. Dulebenets, A comprehensive multi-objective optimization model for the vessel scheduling problem in liner shipping, Int. J. Prod. Econ. (2018). doi:10.1016/j.ijpe.2017.10.027.

[6] F. Cheraghchi, I. Abualhaol, R. Falcon, R. Abielmona, B. Raahemi, E. Petriu, Modeling the speed-based vessel schedule recovery problem using evolutionary multiobjective optimization, Inf. Sci. (Ny). (2018). doi:10.1016/j.ins.2018.03.013.

[7] S. Guericke, K. Tierney, Liner shipping cargo allocation with service levels and speed optimization, Transp. Res. Part E Logist. Transp. Rev. (2015). doi:10.1016/j.tre.2015.10.002.

[8] T.E. Notteboom, The Time factor in liner shipping services, Marit. Econ. Logist. (2006). doi:10.1057/palgrave.mel.9100148.

[9] C.V. Karsten, D. Pisinger, S. Ropke, B.D. Brouer, The time constrained multi-commodity network flow problem and its application to liner shipping network design, Transp. Res. Part E Logist. Transp. Rev. (2015). doi:10.1016/j.tre.2015.01.005.

[10] S. Gelareh, S. Nickel, D. Pisinger, Liner shipping hub network design in a competitive environment, Transp. Res. Part E Logist. Transp. Rev. (2010). doi:10.1016/j.tre.2010.05.005.

[11] S. Wang, Q. Meng, C.Y. Lee, Liner container assignment model with transit-time-sensitive container shipment demand and its applications, Transp. Res. Part B Methodol. (2016).

doi:10.1016/j.trb.2016.04.014.

[12] R. Agarwal, Ö. Ergun, Ship Scheduling and Network Design for Cargo Routing in Liner Shipping, Transp. Sci. (2008). doi:10.1287/trsc.1070.0205.

[13] M. Christiansen, K. Fagerholt, D. Ronen, Ship Routing and Scheduling: Status and Perspectives, Transp. Sci. (2004). doi:10.1287/trsc.1030.0036.

[14] B.D. Brouer, D. Pisinger, S. Spoorendonk, Liner Shipping Cargo Allocation with Repositioning of Empty Containers, INFOR Inf. Syst. Oper. Res. (2011). doi:10.3138/infor.49.2.109.

[15] S. Wang, Q. Meng, Schedule Design and Container Routing in Liner Shipping, Transp. Res. Rec. J. Transp. Res. Board. (2011). doi:10.3141/2222-04.

[16] S. Wang, Q. Meng, Liner ship fleet deployment with container transshipment operations, Transp. Res. Part E Logist. Transp. Rev. (2012). doi:10.1016/j.tre.2011.10.011.

[17] S. Wang, Q. Meng, Z. Liu, Containership scheduling with transit-time-sensitive container shipment demand, Transp. Res. Part B Methodol. (2013). doi:10.1016/j.trb.2013.04.003.

[18] S. Wang, Q. Meng, Liner shipping network design with deadlines, Comput. Oper. Res. (2014). doi:10.1016/j.cor.2013.08.014.

[19] L.B. Reinhardt, C.E.M. Plum, D. Pisinger, M.M. Sigurd, G.T.P. Vial, The liner shipping berth scheduling problem with transit times, Transp. Res. Part E Logist. Transp. Rev. (2016). doi:10.1016/j.tre.2015.12.006.

[20] H. Wang, X. Zhang, S. Wang, A joint optimization model for liner container cargo assignment problem using state-augmented shipping network framework, Transp. Res. Part C Emerg. Technol. (2016). doi:10.1016/j.trc.2016.05.001.

[21] S. Wang, D. Yang, J. Lu, A connectivity reliability-cost approach for path selection in the maritime transportation of China’s crude oil imports, Marit. Policy Manag. (2018). doi:10.1080/03088839.2018.1445308.

[22] C. Güven, D. Türsel Eliiyi, Modelling and optimisation of online container stacking with operational constraints, Marit. Policy Manag. (2019). doi:10.1080/03088839.2018.1450529.

[23] Y. Wang, G.T. Yeo, Intermodal route selection for cargo transportation from Korea to Central Asia by adopting Fuzzy Delphi and Fuzzy ELECTRE I methods, Marit. Policy Manag. (2018). doi:10.1080/03088839.2017.1319581.

[24] T.N. Chuang, C.T. Lin, J.Y. Kung, M. Da Lin, Planning the route of container ships: A fuzzy genetic approach, Expert Syst. Appl. (2010). doi:10.1016/j.eswa.2009.09.040.

[25] B.D. Brouer, J. Dirksen, D. Pisinger, C.E.M. Plum, B. Vaaben, The Vessel Schedule Recovery Problem (VSRP) - A MIP model for handling disruptions in liner shipping, Eur. J. Oper. Res. (2013). doi:10.1016/j.ejor.2012.08.016.

[26] A. Alharbi, S. Wang, P. Davy, Schedule design for sustainable container supply chain networks with port time windows, Adv. Eng. Informatics. (2015). doi:10.1016/j.aei.2014.12.001.

[27] M.A. Dulebenets, The Vessel Scheduling Problem in a Liner Shipping Route with Heterogeneous Fleet, Int. J. Civ. Eng. (2018). doi:10.1007/s40999-016-0060-z.

[28] S. Gürel, A. Shadmand, A heterogeneous fleet liner ship scheduling problem with port time uncertainty, Cent. Eur. J. Oper. Res. (2018). doi:10.1007/s10100-018-0554-7.

[29] M.A. Dulebenets, E.E. Ozguven, Vessel scheduling in liner shipping: Modeling transport of perishable assets, Int. J. Prod. Econ. (2017). doi:10.1016/j.ijpe.2016.11.011.

[30] K. Fagerholt, N.T. Gausel, J.G. Rakke, H.N. Psaraftis, Maritime routing and speed optimization with emission control areas, Transp. Res. Part C Emerg. Technol. (2015). doi:10.1016/j.trc.2014.12.010.

[31] X. Qi, D.P. Song, Minimizing fuel emissions by optimizing vessel schedules in liner shipping with uncertain port times, Transp. Res. Part E Logist. Transp. Rev. (2012). doi:10.1016/j.tre.2012.02.001.

[32] M.A. Dulebenets, Minimizing the Total Liner Shipping Route Service Costs via Application of an Efficient Collaborative Agreement, IEEE Trans. Intell. Transp. Syst. (2019). doi:10.1109/TITS.2018.2801823.

[33] J.F. Benders, Partitioning procedures for solving mixed-variables programming problems, Numer. Math. (1962). doi:10.1007/BF01386316.

[34] J.-F. Cordeau, G. Stojković, F. Soumis, J. Desrosiers, Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling, Transp. Sci. (2003). doi:10.1287/trsc.35.4.375.10432.

[35] S. Binato, M.V.F. Pereira, S. Granville, A new Benders decomposition approach to solve power transmission network design problems, IEEE Trans. Power Syst. (2001). doi:10.1109/59.918292.

[36] S. Gelareh, D. Pisinger, Simultaneous Fleet Deployment and Network Design of Liner Shipping, 2010.

[37] R. Rahmaniani, T.G. Crainic, M. Gendreau, W. Rei, The Benders decomposition algorithm: A literature review, Eur. J. Oper. Res. (2017). doi:10.1016/j.ejor.2016.12.005.

[38] Rudin, P. (2016). OR in an OB World. https://orinanobworld.blogspot.com.tr/2016/12/support-for-benders-decomposition-in.html accessed in April 04, 2019

[39] D. Ronen, Ship scheduling: The last decade, Eur. J. Oper. Res. (1993). doi:10.1016/0377-2217(93)90343-L.