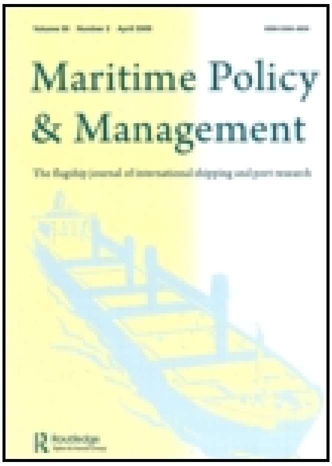

Maritime Policy & Management: The flagship journal of international shipping and port research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tmpm20

A combined tramp ship routing, fleet deployment, and network design problem

I. K. Moon[a], Z. B. Qiu[b] & J. H. Wang[c]
[a] Department of Industrial Engineering, Seoul National University, Seoul 151-744, Korea
[b] LG Electronics Hi Logistics (China) Co., Ltd., No. 9 Wangjing Beilu, Chaoyang District, Beijing 100102, China
[c] Department of Industrial Engineering, Pusan National University, Busan 609-735, Korea
Published online: 08 Jan 2014.

Maritime Policy & Management, 2015
Vol. 42, No. 1, 68–91, http://dx.doi.org/10.1080/03088839.2013.865847

A combined tramp ship routing, fleet deployment, and network design problem

I. K. MOON†*, Z. B. QIU‡ and J. H. WANG§

†Department of Industrial Engineering, Seoul National University, Seoul 151-744, Korea
‡LG Electronics Hi Logistics (China) Co., Ltd., No. 9 Wangjing Beilu, Chaoyang District, Beijing 100102, China
§Department of Industrial Engineering, Pusan National University, Busan 609-735, Korea

In this paper, a tramp ship routing model of fleet deployment in a hub-and-spoke network is presented. This model simultaneously determines the selection of hubs, the assignment of spokes to hubs, the deployment of feeder-containerships as well as containership routing between spokes and spokes, hubs and spokes, and hubs and hubs. Even though some parts have been studied, this complex combination of shipping problems has never been addressed. Because the problem is NP-hard, a genetic algorithm (GA) with local search is proposed. In the algorithm, a cut-off procedure is applied to fleet deployment in a sub-route strategy. A number of randomly generated problem instances are solved by both a mathematical program and the GA with local search. A simple but realistic heuristic algorithm is also developed. Both the GA with local search and the heuristic algorithm are used to solve a number of real case instances. A comparison of the results shows the efficiency of the GA with local search. The developed model can be used as a route-decision support tool for shipping companies that provide long-haul shipping services in a hub-and-spoke network.

1. Introduction

As the global economy increases and container transportation becomes progressively more popular, containerships from small feeders to the Panamax, Post-Panamax, and ultra-large containerships (ULCSs) show increased capacity. Examination of cellular ship deliveries for the period 2008 to 2011 reveals that the shift toward larger ships is continuing, as shown in Figure 1 (BRS 2008). The major reason for this popularity is that the large containerships possess the advantage of economies of scale; that is, the operating cost of a containership does not increase proportionally with its capacity. Currently, some ULCSs feature capacities of over 10 000 TEU. For example, since early 2010, the Maersk Line from Denmark owned and operated the eight largest container ships, each with a nominal capacity of 14 770 TEU (UNCTAD 2010).

Travel of large containerships is severely constrained by specifications of ports, many of which have small navigational channels, limited berth depths, and other restrictive factors related to equipment and infrastructure. The ports capable of servicing large containerships are known as ‘hubs’. To maximize the benefits from economies of scale, the large

*To whom correspondence should be addressed. E-mail: ikmoon@snu.ac.kr

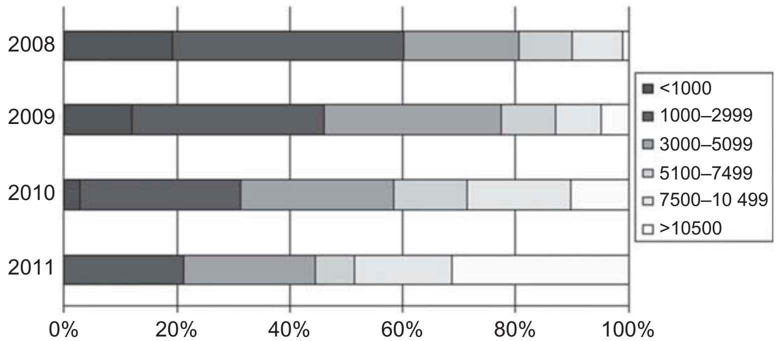

Figure 1. Cellular ship deliveries for the period 2008 to 2011.

containerships are usually used in transoceanic routes, such as the Trans-Pacific Service, the Trans-Atlantic Service, the Asia–Europe Service, and the Asia–Australia Service.

To attain the benefits of economies of scale, the capacity for large containerships must be extremely well utilized; low utilization and high fixed and travel costs render the service uneconomical. Therefore, with the rare exception in which one single load is large enough to meet the capacity of the containership, the demand should be consolidated for large containerships. This need for consolidated demand means that large containerships are usually used in a hub-and-spoke network. In the geographical region around each hub, feeder-containerships, utilizing ports called 'spokes', bring goods to the hub, where they are consolidated and picked up by large containerships.

Traditionally, a containership that travels on a predefined route for a significant length of time is called 'liner shipping'. Because demand seldom reaches economies of scale, a typical ULCS is not used for liner shipping on a route with a consistent published itinerary and schedule. Furthermore, the fixed and travel costs for ULCSs are high. Therefore, companies must simultaneously determine the most economical route of ULCSs between hubs as well as the routes of the feeder-containerships.

The case study problem presented in this paper involves a shipping service between eastern Asia and the western US through a hub-and-spoke network. The shipping company has signed a third-party logistics service contract with a clothes export union in China. This giant union has many factories within the origination region and many service areas in the US. Each month, the products of local factories are transported to the nearest ports by feeder-containerships with various capacities: less than 1000 TEU, 1000–2500 TEU, and 2500–5000 TEU (Panamax). The demand is then transported to hubs from which they are shipped to the US in large containerships: Post-Panamax with a capacity of 8000 TEU, Super Post-Panamax with a capacity of 8000–11 000 TEU, and ULCSs with capacity of more than 11 000 TEU. The shipping company controls the maritime transportation of all of these products. Because of contract specifications, the containerships used for the union clothing cannot be used to transport products from other companies; that is, the shipping company must provide a special service for the union. Hence, the mode of shipping operation is similar to that of tramp shipping without the ability to service others' goods.

Traditionally, those transporting bulk cargo, such as coal and iron, by tramp shipping did not need to take into consideration hub-and-spoke operations (Meng and Wang 2011). However, in the presented case study, featuring containers transported by tramp shipping, the company utilizes hub-and-spoke operations to maximize benefits from economies of scale. To successfully address the ship routing problem, we must simultaneously determine the selection of hubs, the assignment of spokes to hubs, and the deployment of feeder-containerships as well as the containership routing between spokes and spokes, hubs and spokes, and hubs and hubs. Here, the term 'routing' refers to the specification of sequences of ports of call to containerships; it does not refer to the path choice in the sea between two ports of call (Ronen 1983).

Ship routing and scheduling has been studied for many years. Ronen (1983) provided the first survey of ship routing and scheduling; in this work, various modes of cargo ships operation (e.g., liner shipping, tramp shipping, and industrial shipping) were described, and a classification scheme for ship routing and scheduling models was provided. Ronen (1993) published a second review on ship scheduling and related problems for the period between 1982 and 1992. Recently, a thorough survey of ship routing and scheduling research was conducted by Christiansen, Fagerholt, and Ronen (2004); they suggested that little work had been performed on tramp shipping or on the shift from industrial to tramp shipping.

More recently, as many companies focus on their core business and outsource transportation activities to third-party logistics companies (Christiansen et al. 2007), researchers see a corresponding increase of tramp shipping studies. Brønmo, Christiansen, Fagerholt, and Nygreen (2007) presented a maximum profit problem with pickups and deliveries of bulk cargoes in tramp shipping. A multi-start local search heuristic was proposed to solve the problem. The same classical ship scheduling problem was later solved by Korsvik, Fagerholt, and Laporte (2010) using a tabu search algorithm and by Malliappi, Bennell, and Potts (2011) using a variable neighborhood search heuristic. Considering flexible cargo sizes, Brønmo, Christiansen, and Nygreen (2007) described a ship routing and scheduling problem in the tramp shipping industry. Several small but real cases were solved by a set partitioning approach. For larger instances in which a priori column generation fails, a dynamic column generation scheme was proposed by Brønmo, Nygreen, and Lysgaard (2010). Korsvik, Fagerholt, and Laporte (2011) proposed a large neighborhood search heuristic for a ship routing and scheduling problem with split loads, in which each cargo can be transported by several ships. A new tramp ship routing and scheduling approach that considered speed occptimization was described by Norstad, Fagerholt, and Laporte (2011). However, no tramp ship routing problem that involves hub-and-spoke network designs is described in the literature.

Several authors have addressed the determination of hub-and-spoke networks in liner shipping, albeit using problems different than the one we present. Fagerholt (1999) examined the problem of determining the optimal fleet and the corresponding weekly routes for the selected ships in a real feeder system. The problem adopted the technique used for the multi-trip vehicle routing problem and was solved by a set partitioning approach. Although the feeder system was the focus of this study, transportation from the depot to the markets was not considered. Fagerholt (2004) solved a similar problem, based on a fixed heterogeneous ship fleet with a given cost structure, capacity, and sailing speed for each type of ship. Aversa et al. (2005) proposed a mixed integer programming (MIP) model for locating a hub port on the east coast of South America. Gelareh, Nickel, and Pisinger (2010) addressed the liner shipping hub network design problem in a competitive environment. Imai et al. (2006) studied container mega-ship viability using a game-theory model that was based on different service network configurations for various ship sizes: hub-and-spoke for mega-ships and multi-port calling for conventional ship sizes. Based on this model, Imai, Shintani, and Papadimitriou (2009) compared the efficiency of the two shipping network topological structures by looking at empty container repositioning. Meng and Wang (2011) proposed a liner shipping service network design problem with combined hub-and-spoke and multi-port-calling operations and empty container repositioning. On a strategic level, Hsu and Hsieh (2007) studied routing, ship size, and frequency decision-making for a maritime hub-and-spoke container network. They formulated a two-objective model of shipping and inventory costs and used the Pareto optimal solution. In their model, large containerships can visit any port and hub selection is not considered. Kjeldsen (2011) published a new review for ship routing and scheduling problems in liner shipping.

As one part of the complex situation, the fleet deployment problem (FDP) has been addressed in various papers. Meng and Wang (2010) investigated a short-term liner ship fleet planning (LSFP) method with demand uncertainty. They comprehensively reviewed relevant studies, and we refer to their paper as a survey. In addition, a general review of the network hub location problem appears in Alumur and Kara (2008). The first model for the simultaneous network design of a hub-and-spoke network and fleet deployment in liner shipping was presented by Gelareh and Pisinger (2011), who proposed a mixed integer linear programming formulation to determine a cyclic hub network and to find the capacity of the hub and feeder lines. In the model, the demand is elastic that the service provider can accept any fraction of the origin–destination demand. However, routing between two spokes was not considered in the feeder system, and the primal decomposition method was applied to small-sized problems. Thus, a survey of the existing literature shows that the combined containership routing problem presented in this paper has never been addressed.

This paper is organized as follows. In Section 2, the problem is defined, and the assumptions, notation, and a mathematical model are presented. In Section 3, we present a genetic algorithm (GA) with local search. In Section 4, a simple heuristic algorithm is proposed. Computational experiments are presented in Section 5. Finally, a number of concluding remarks are provided in Section 6.

2. Mathematical model

The problem considered involves a shipping company that provides a shipping service between an origin and a destination region through a hub-and-spoke network. The shipping company uses feeder-containerships to provide a short sea service and ULCSs to provide a transoceanic service. Feeder-containerships can visit both hubs and spokes but cannot be used to travel across oceans. Because of the limited berth depth and other infrastructure restrictions, ULCSs can visit only hubs. The entire transportation demand from the origination to the destination ports is provided. All of the origination ports are within the origination region (e.g., eastern Asia), and all of the destination ports are within the destination region (e.g., western US). The solution requires simultaneously deciding which origination and destination region hubs are to be called upon by ULCSs, determining the spokes to be assigned to them, and deciding on the deployment of feeder-containerships as well of the containership routing between spokes and spokes, hubs and spokes, and hubs and hubs.

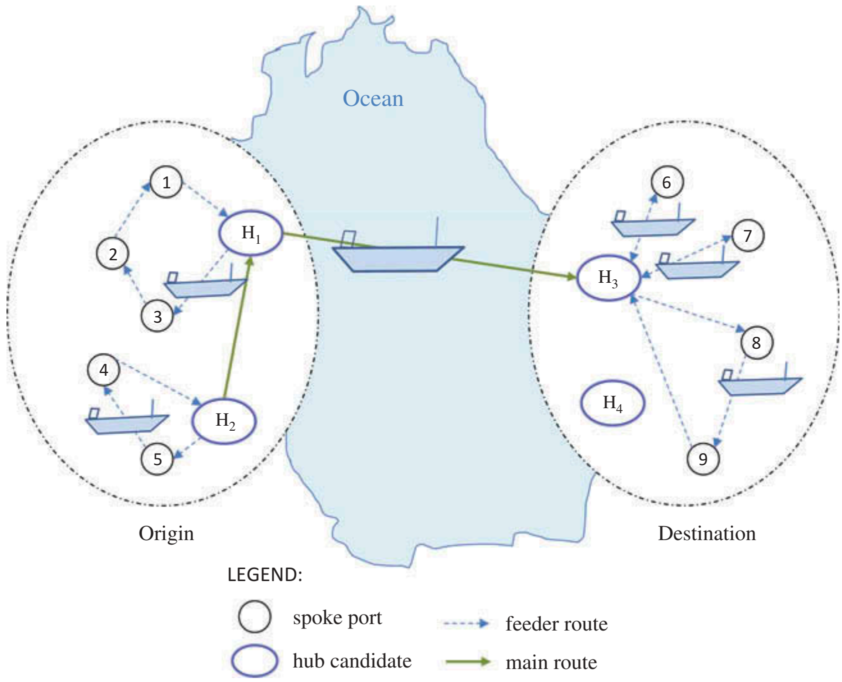

Figure 2. Example of the problem.

Figure 2 shows an example of the situation, with the predefined feasible routes indicated. In the origination region, ports 1, 2, 3, 4, and 5 are spokes, while ports H1 and H2 are hub candidates. In the destination region, ports 6, 7, 8, and 9 are spokes, while ports H3 and H4 are hub candidates. As shown in Figure 2, ports H_1, H_2, and H_3 are selected. One ULCS travels on the route H_2-H_1-H_3. Ports 1, 2, and 3 are assigned to hub H_1 and are visited by a feeder-containership in the sequence of H_1-3-2-1-H_1; the demands for this route are collected by the feeder-containership and are transported to hub H_1. At hub H_1, they will be loaded onto the ULCS and transshipped to hub H_3 and then distributed to their destinations by the other feeder-containerships on different routes within the destination region. Routes H_2-5-4-H_2, H_3-6-H_3, H_3-7-H_3, and H_3-8-9-H_3 are followed in a similar fashion.

The major assumptions of this study are as follows:

(1) All of the transportation demands are known and must be satisfied.
(2) There are no time-window constraints for container delivery.
(3) The locations of spokes and hub candidates are known.
(4) Each spoke is served by exactly one feeder-containership; thus, the demand of each spoke cannot be split in this model.
(5) The containerships are initially located at the hub candidate locations.
(6) The feeder-containership must return to the same hub from which it departs.

(7) The main route is operated by only one ULCS and the capacity of the ULCS is sufficient to accommodate all of the demand.
(8) The handling cost (the loading and unloading cost) depends on the ports and is independent of containership type.

The following notation is used in the model.

Indices

i – spoke index
j, k – hub candidate index
m, n – port (spoke or hub candidate) index
v – feeder-containership index

Sets

I_E – set of spokes in the origination region
I_W – set of spokes in the destination region
J_E – set of hub candidates in the origination region
J_W – set of hub candidates in the destination region
I – set of all spokes, $I = I_E \mathrm{U} I_W$
J – set of all hub candidates, $J = J_E \mathrm{U} J_W$
U – set of all ports, $U = I \mathrm{U} J$
R – subset of ports, $R \mathrm{U} U$
V – set of feeder-containerships

Parameters

L_{mn}–distance between port m and port n, m, $n \in U$ (mile)
D_{mn} – transportation demand for containers from port m to port n in the decision horizon, $m \in I_E$, $n \in I_W$ (TEUs)
$D_m{}^E$ – total demand from spoke m to all of its destinations, $D_m{}^E = \sum_{n \in I_W} D_{mn}, m \in I_E$ (TEUs)
$D_n{}^W$ – total demand to spoke j from all of its origins, $D_n W = \sum_{m \in I_E} D_{mn}, n \in I_W$ (TEUs)
Q_v – capacity of feeder-containership v, $v \in V$ (TEUs)
$C_v{}^F$ – fixed cost of using feeder-containership v, $v \in V$ ($/time)
$C_v{}^{SF}$ – sailing cost per mile for feeder-containership v, $v \in V$ ($)
C^{SU} – sailing cost per mile for ULCS ($)
$C_{nv}{}^{PF}$ – port charge for feeder-containership v at port n, $n \in U$, $v \in V$ ($)
$C_j{}^{PU}$ – port charge for ULCS at hub candidate j, and $j \in J$ ($)
$C_m{}^H$ – handling cost of per TEU of container at port m, $m \in U$ ($)
M– a sufficiently large number

Decision variables

$$x_{mnv} = \begin{cases} 1 & \text{if port } m \text{ immediately precedes } n \text{ on a route using containership } v \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{if spoke } i \text{ is assigned to hub } j \\ 0 & \text{otherwise} \end{cases}$$

$$z_j = \begin{cases} 1 & \text{if hub candidate } j \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

$$w_{jk} = \begin{cases} 1 & \text{if hub } j \text{ immediately precedes hub } k \text{ on a route using an ULCS} \\ 0 & \text{otherwise} \end{cases}$$

Minimize

$$\begin{aligned} &\sum_{v\in V}\sum_{m\in J_E}\sum_{n\in I_E} C_v^F x_{mnv} + \sum_{v\in V}\sum_{m\in J_W}\sum_{n\in I_W} C_v^F x_{mnv} + \sum_{v\in V}\sum_{m\in I_E\cup J_E}\sum_{n\in I_E\cup J_E} C_v^{SF} L_{mn} x_{mnv} \\ &+ \sum_{v\in V}\sum_{m\in I_W\cup J_W}\sum_{n\in I_E\cup J_E} C_v^{SF} L_{mn} x_{mnv} + \sum_{j\in J_E} C_j^H \left(\sum_{m\in J_E} D_i^E y_{ij}\right) \\ &+ \sum_{j\in J_W} C_j^H \left(\sum_{i\in I_W} D_i^W y_{ij}\right) + \sum_{v\in V}\sum_{n\in I_E\cup J_E} C_{nv}^{PF}\left(\sum_{m\in I_E\cup J_E} x_{mnv}\right) + \sum_{j\in J} C_j^{PU} z_j \\ &+ \sum_{j\in J}\sum_{k\in J} C^{SU} L_{jk} w_{jk} \end{aligned} \tag{1}$$

Subject to

$$\sum_{v\in V}\sum_{m\in I_E\cup J_E} x_{mnv} = 1, \forall\, n \in I_E, \tag{2a}$$

$$\sum_{v\in V}\sum_{m\in I_W\cup J_W} x_{mnv} = 1, \forall\, n \in I_W, \tag{2b}$$

$$\sum_{n\in I_E\cup J_E} x_{mnv} - \sum_{n\in I_E\cup J_E} x_{nmv} = 0, \forall\, m \in I_E\cup J_E, \forall\, v \in V, \tag{3a}$$

$$\sum_{n\in I_W\cup J_W} x_{mnv} - \sum_{n\in I_W\cup J_W} x_{nmv} = 0, \forall m \in I_W\cup J_W, \forall\, v \in V, \tag{3b}$$

$$\sum_{m\in R}\sum_{n\in R} x_{mnv} \le |R| - 1, \forall\, R \subseteq I_E, R\neq\emptyset, \forall\, v \in V, \tag{4a}$$

$$\sum_{m\in R}\sum_{n\in R} x_{mnv} \le |R| - 1, \forall\, R \subseteq I_W, R\neq\emptyset, \forall\, v \in V, \tag{4b}$$

$$\sum_{m\in J}\sum_{n\in I} x_{mnv} \le 1, \forall\, v \in V, \tag{5}$$

$$\sum_{m\in I_E}\sum_{n\in I_E\cup J_E} D_n^w x_{nmv} \le Q_v, \forall\, v \in V, \tag{6a}$$

$$\sum_{n\in I_w}\sum_{m\in I_w\cup J_w} D_n^w x_{nmv} \le Q_v, \forall\, v \in V, \tag{6b}$$

$$\sum_{i\in I_E} D_i^E y_{ij} \le M z_j, \forall\, j \in J_E, \tag{7a}$$

$$\sum_{i\in I_W} D_i^W y_{ij} \le M z_j, \forall\, j \in J_W, \tag{7b}$$

$$\sum_{n \in I_E \cup J_E} (x_{inv} + x_{jnv}) - y_{ij} \leq 1, \forall\, i \in I_E, \forall\, j \in J_E, \forall\, v \in V, \tag{8a}$$

$$\sum_{n \in I_W \cup J_W} (x_{inv} + x_{jnv}) - y_{ij} \leq 1, \forall\, i \in I_W, \forall\, j \in J_W, \forall\, v \in V, \tag{8b}$$

$$\sum_{J \in J} z_j \geq 2, \tag{9}$$

$$\sum_{k \in J} w_{jk} = z_j, \forall\, j \in J_E, \tag{10}$$

$$\sum_{k \in J_E} w_{jk} \leq z_j, \forall\, j \in J_E, \tag{11}$$

$$\sum_{k \in J} w_{kj} = z_j, \forall\, j \in J_W, \tag{12}$$

$$\sum_{k \in J_W} w_{jk} \leq z_j, \forall\, j \in J_W, \tag{13}$$

$$\sum_{k \in J_E} \sum_{k \in J_W} w_{jk} = 1, \tag{14}$$

$$\sum_{j \in R} \sum_{k \in R} w_{jk} \leq |R| - 1, \forall\, R \subseteq J_E, R \neq \emptyset, \tag{15a}$$

$$\sum_{j \in R} \sum_{k \in R} w_{jk} \leq |R| - 1, \forall\, R \subseteq J_W, R \neq \emptyset, \tag{15b}$$

$$x_{mnv} \in \{0, 1\}, \forall\, m \in U, \forall\, n \in U, \forall\, v \in V, \tag{16a}$$

$$w_{jk} \in \{0, 1\}, \forall\, j \in J, \forall\, k \in J, \tag{16b}$$

$$y_{ij} \in \{0, 1\}, \forall\, i \in I, \forall\, j \in J, \tag{16c}$$

$$z_j \in \{0, 1\}, \forall\, j \in J. \tag{16d}$$

The objective function minimizes the summation of the fixed costs of using feeder-containerships (the first and the second terms), the transportation costs of feeder-containerships in the origination region (the third term) and the destination region (the fourth term), the total handling costs in the hubs (the fifth and sixth terms), the port charges of the feeder-containerships (the seventh and eighth terms) and the ULCS (the ninth term), and the sailing costs of the ULCS (the tenth term). Because all of the demands must be satisfied, each spoke must be visited. Therefore, the total handling costs at the spokes are constant, and they are omitted in the objective function. Constraints (2a) and (2b) guarantee that a spoke belongs to one route only and that each spoke has only one predecessor. Constraints (3a) and (3b) ensure that each spoke entered by a feeder-containership should be left by the same feeder-containership. Constraints (4a) and (4b) are subtour elimination constraints for the feeder-containerships. Constraint (5) ensures that each route can be served no more than once and that a route cannot be operated from multiple hubs (i.e., a feeder-containership can only operate from one hub). Constraints (6a) and (6b) are the capacity constraints for the feeder-containerships. Constraints (7a) and (7b) ensure that the demand to or from the spokes can pass through a selected hub. Constraints (8a) and (8b) specify that a spoke can be assigned to a hub only a route operates from the hub and passes through that spoke.

Constraint (9) specifies that at least two hub candidates should be selected; there should be at least one hub in the origination region and at least one hub in the destination region to serve the spokes in each region. Constraint (9) can be eliminated because of Constraints (7a) and (7b). From constraint (7a), we know that at least one $z_j = 1$; otherwise, the spokes in the origination region have no hub assigned to them. Analogously, we could obtain at least one $z_j = 1$ from Constraint (7b). However, the computational time will decrease significantly by adding Constraint (9). Constraint (10) ensures that, after the ULCS visits a selected hub within the origination region, it must leave it for another selected hub. It may proceed to another hub within the origination region or within the destination region. Constraint (11) ensures that the selected hub will be visited no more than once by the ULCS. Constraint (12) guarantees that, if the hub in the destination region is selected, it should be visited exactly once. Constraint (13) ensures that the ULCS cannot visit two other selected hubs at the same time. Constraint (14) ensures that the ULCS proceeds across the ocean from the origination region to the destination region. Constraints (15a) and (15b) are sub-tour elimination constraints for ULCS; they are similar to Constraints (4a) and (4b). Finally, Constraints (16a)–(16d) are the binary requirements for the route design variables (x_{mnv}, w_{jk}), the allocation variables (y_{ij}), and the location variables (z_j), respectively.

3. GA with local search

The problem addressed here could be regarded as the sum of the following three NP-hard problems: (1) a location–allocation problem based a selection of hubs from candidates and the assignment of spokes to hubs, (2) a traveling salesman problem considering the routing between hubs and hubs, and (3) a multiple vehicle routing problem of container-ship routing between spokes and spokes, hubs and spokes, and the deployment of feeder-containerships on the routes. Each individual problem is proven to be an NP-hard problem based on its complexity and difficulty. The time that it takes to search for an optimal solution rapidly increases as the number of ports is increased. Therefore, some heuristic algorithms should be developed.

In recent years, the GA developed by John Holland in the 1960s has been applied successfully to hard optimization problems. Combined with some algorithms, GAs have been used in several location and transportation problems (Suzuki and Yamamoto 2012; Karaoglan and Altiparmak 2011). In this paper, a GA with local search is developed to solve the problem. A cut-off procedure to address fleet deployment is applied to the sub-route strategy, and two procedures embedded in the local search are used to improve the solutions obtained from the genetic operator. The flowchart of the proposed GA is shown in Figure 3.

3.1. *GA with local search*

To apply the algorithm to solve the proposed problem, the following steps are introduced.

3.1.1. *Gene representation.* The choice of representing the solution as a chromosome is crucial in implementing the GA. Sound representation will ensure that the algorithm converges to sound solutions. In this paper, we encode hub selection, the main route, and fleet deployment into a single genetic code by using three strings. The length of the chromosome is equal to the number of spokes. The first string of the gene shows the sequence of spokes. The second string shows the hub where the spoke is allocated. If the position under spoke i is occupied by hub j, then spoke i is assigned to hub j. A hub that is not in the chromosome is not selected. Therefore, the selection of

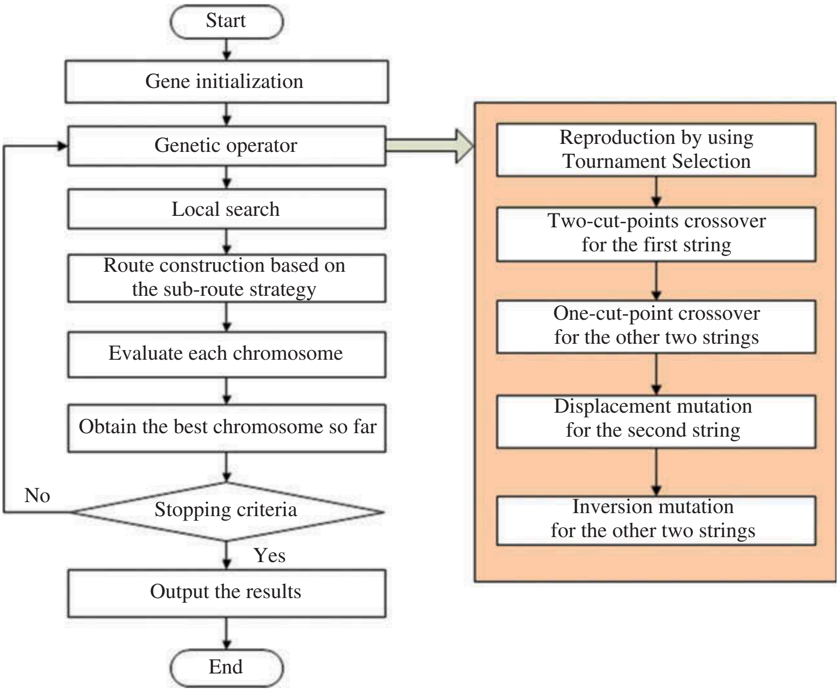

Figure 3. Flowchart of the genetic algorithm with local search.

Spoke No.	5	3	8	9	2	4	1	7	6
Hub No.	H_2	H_1	H_3	H_3	H_1	H_2	H_1	H_3	H_3
Ship No.	6	3	4	5	2	5	4	3	6

Figure 4. Possible chromosomes for a sample problem.

the hub is also implied by the chromosome. Meanwhile, the permutation of the distinct hub numbers in the chromosome represents a main route. The third string is occupied by the types of feeder-containerships and shows the deployed sequence of the feeder-containerships. For example, a chromosome for a nine-spoke and three-hub problem could be configured as shown in Figure 4.

From this chromosome, we can find that spokes {1, 2, 3} are assigned to hub H_1, spokes {4, 5} are assigned to hub H_2, and spokes {6, 7, 8, 9} are assigned to hub H_3. The dropping sequence of the spokes is 3-2-1, 5-4, and 8-9-7-6. The selected hubs are {H_1, H_2, H_3}. A main route, H_3-H_2-H_1, can also be obtained. Then, we use the sequence 6-3-4-5-2-5-4-3-6 to deploy the feeder-containerships using the sub-route strategy, which will be described.

3.1.2. *Gene initialization.* In the initialization process, we construct an initial solution by using a heuristic method for the hub selection and a random generation method for the sequence of spokes and feeder-containerships.

The initialization mechanisms are as follows:

Step 1. For each position of the gene, randomly generate a sequence of spokes.

Step 2. For each spoke within the origination region, select the nearest hub candidate within the origination region and place it in a position under the spoke.

Step 3. Similarly, for each spoke within the destination region, select the nearest hub candidate within the destination and place it in a position under the spoke.

Step 4. For each position of the gene, randomly generate a type of feeder-containership.

3.1.3. *Fitness function.* The fitness function of a GA plays a role similar to that of the environment in a natural evaluation. The quality of a chromosome (whether it is good or poor) is evaluated by the fitness function. Thus, a fitness function is computed for each chromosome within the population. Because the objective function in the proposed mathematical model is to minimize the total relevant cost, it is convenient and effective to use the reciprocal of this objective function as the fitness function:

$$Fitness\,(A) = \frac{1}{Minimize\,Z}$$

where A is the chromosome that is evaluated. To obtain the fitness value of each chromosome, the sub-route strategy is applied that provides the routing information.

3.1.4. *Reproduction.* Reproduction is a process in which individuals participate according to their fitness function values. Individuals with high fitness values will be reproduced with a higher probability, while those with lower fitness values will be discarded in the subsequent generation. In this paper, tournament selection was used as the primary selection technique, and the procedure is as follows:

Step 1. Calculate the fitness value of each individual in the population.

Step 2. Randomly pick two individuals from the population with replacement, and return the fitter individual to the new population.

Step 3. Repeat *Steps* 1 and 2 until the number of selected individuals is equal to the population size.

3.1.5. *Crossover.* The crossover operation is a diversification mechanism that enables a GA to examine unvisited space and generate solutions. In this paper, the two-cut-points method is applied to the sequence of spokes in the first string and the one-cut-point method is applied to the hub selection and the deployment of feeder-containerships in the other two strings. The procedure for the crossover operator is as follows:

Step 1. Select whether crossover is utilized by using the crossover probability.

Step 2. Randomly generate two different crossing points from the range of [1, length of the chromosome] and exchange the spokes between the two points.

Step 3. The spokes of parent 1 (2) that are not selected from parent 2 (1) are inherited by child 1(2) in the originating sequence.

Step 4. Sort the spokes in ascending order and apply the one-cut-point method to the second and third strings.
Step 5. Resort the spokes to the same sequence as the spokes in *Step* 3 because to apply the one-cut-point method to the other two strings, the spokes must be sorted in ascending order as in *Step* 4.

3.1.6. *Mutation.* In a GA, mutation is a background operator that produces a spontaneous random change in various chromosomes. Mutation plays the crucial role of replacing genes lost from the population during evolution so that the gene pool can maintain diversity, thereby preventing the population from generating immature convergence. Inversion and displacement are simple ways to achieve mutation. In this paper, an inversion operator is applied to the first and third strings and a displacement operator is applied to the second string.

Specifically, the mutation procedure in the proposed GA is as follows:

Step 1. Determine whether the mutation is utilized by using the mutation probability.

Step 2. Generate two different positions randomly from the range of [1, length of the chromosome] and reverse the spokes between the two points.

Step 3. Generate a position randomly from the range of [1, length of the chromosome], and check whether the hub at that position belongs to the origin or to the destination.

Step 4. Displace the hub with another hub that is randomly generated from the same set.

Step 5. For the third string, randomly generate two additional different positions from the range of [1, length of the chromosome], and reverse the feeder-containerships between the two points.

3.2. *Sub-route strategy*

As mentioned earlier, the gene used in the GA can reflect the hub selection, the main route, and the deployment of feeder-containerships using three strings; the sub-routes within one selected hub and the spokes belonging to it remain to be represented. Here, we explore a cut-off procedure to determine the sub-routes and the fleet deployment based on the chromosome obtained from the GA operator.

The procedure for the sub-route strategy is as follows:

Step 1. Select the first hub and the sequence of spokes belonging to it from each chromosome.

Step 2. Sweep the spokes belonging to the hub until the sum of demands is larger than the capacity of the feeder-containership. The spokes, excluding the last one, and the hub build a sub-route with the feeder-containership. Continue to sweep the remaining spokes using the next feeder-containership.

Step 3. If the last feeder-containership has arrived, deploy from the first one; if all of the feeder-containerships in the chromosome cannot be used, then generate new feeder-containerships randomly and change the third string of the chromosome.

Step 4. If the last spoke belonging to the hub has arrived, then select the next different hub and go to *Step* 2.

Step 5. If all of the selected hubs have arrived, then end the algorithm.

3.3. *Local search*

In recent years, a local search has often been embedded in a GA to compensate for the GA's poor local search ability. However, this strategy is very time consuming because the search heuristic is applied to every chromosome. In this paper, we apply the local search to the first 20% of the population after sorting by the fitness values. In the local search, two processes, a drop procedure and a feeder-containership displacement procedure, are used.

3.3.1. *Drop procedure.*

Although transportation costs could decrease if added, one more selected hub port results in a large, fixed, hub port charge cost. The drop procedure is used to make trade-offs between variable transportation costs and fixed hub port charges.

Step 1. Within each region, check whether only one hub port has been selected.

Step 2. If more than one hub port has been selected in the same region, try to drop a selected hub port and exchange it with another selected hub port. If there is a positive savings value with respect to the total cost, perform the process and go to *Step* 3; otherwise, repeat Step 2 by dropping another hub port.

Step 3. Repeat *Steps* 1 and 2 until no positive savings can be obtained.

3.3.2. *Feeder-containership displacement procedure.*

After the deployment of a feeder-containership in the sub-route strategy, a larger ship may be deployed to a route for which a smaller ship is also suitable. Therefore, a feeder-containership displacement procedure is utilized as follows:

Step 1. For every route, check the total demand of the spokes and the feeder-containership that is deployed.

Step 2. Check the smallest feeder-containership that has a capacity larger than the total demand. If it is not the same type as the deployed feeder-containership, then displace the deployed ship with the smaller-capacity feeder-containership.

4. Heuristic algorithm

The case study problem presented in this paper involves a company that provides a shipping service between eastern Asia and the western US. The company uses a decision support tool wherein a simple heuristic is embedded, to make the ship routing decisions.

In the algorithm, the combined problem is divided into three sub-problems: the hub location routing problems of the origination and destination region as well as the routing problem for ULCSs. The location problem is solved by the add algorithm, the feeder-containership routing problem is solved by the savings algorithm (Clarke and Wright 1964), and the ULCS routing algorithm is solved by the greedy algorithm. The flowchart of the heuristic algorithm is shown in Figure 5.

4.1. *Add algorithm*

The well-known add algorithm is a greedy heuristic that enlarges the set of open hubs one by one through the use of the hub that yields the largest decrease in the objective function value. The algorithm is ended when no positive decrease is obtained by using another hub or until all of the potential hubs are used. The algorithm follows:

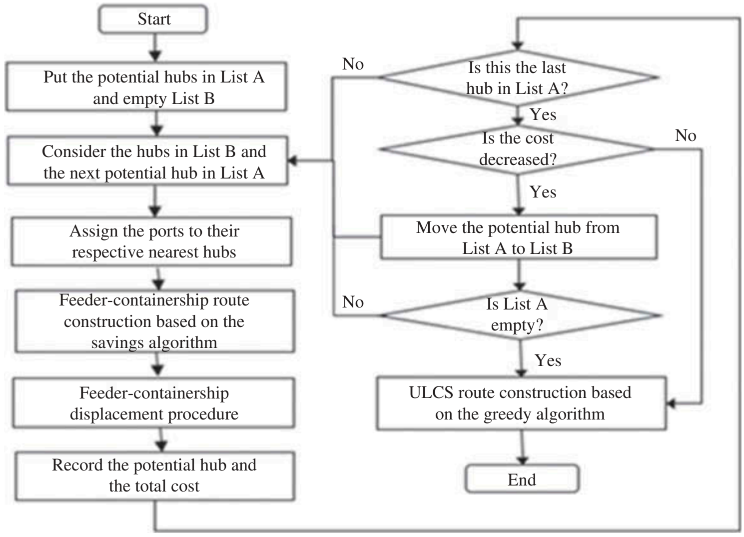

Figure 5. Flowchart of the realistic algorithm.

Step 1. Set the initial total cost to be a big number, put all the potential hubs in List A, and empty List B.

Step 2. Consider the hubs in List B and the next potential hub in List A, assign the ports to their respective nearest hubs.

Step 3. If this is the last potential hub in List A, obtain the potential hub that yields the largest decrease in the total cost and go to *Step* 4; otherwise, go to *Step* 2.

Step 4. If the largest decrease is positive, move the potential hub with minimum total cost from List A to List B and go to *Step* 5; otherwise, go to *Step* 6.

Step 5. If List A is empty, go to *Step* 6; otherwise, go to *Step* 2.

Step 6. Update the solution and the add algorithm stops.

4.2. *Savings algorithm*

The savings algorithm was developed by Clarke and Wright (1964) to solve a vehicle routing problem, which is a constructive and greedy heuristic algorithm. In the savings algorithm, only the largest feeder-containership is available to construct the routes. The procedures of the algorithm used in this paper are as follows:

Step 1. Assume each spoke is visited by one dedicated route (feeder-containership).

Step 2. Inspect each pair of routes in which the total load does not exceed the capacity of the feeder-containership and then evaluate the savings value that would result from this merger. For instance, if H denotes the unique hub node and i, j, k, and l, respectively, stand for the first and last spokes of route R and the first and last

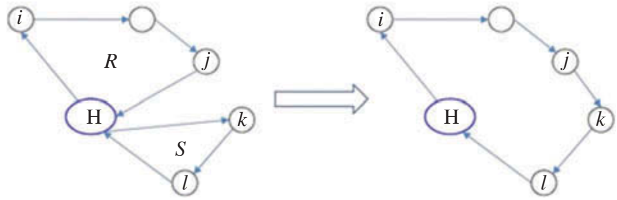

Figure 6. Merger in savings algorithm.

spokes of route S, the savings obtained by concatenating S with R is equal to $C_{jH} + C_{Hk} - C_{jk}$ + the fixed cost of using a feeder-containership on route S (Figure 6).

Step 3. Sort the savings values in descending order. Execute the merger that provides the largest positive savings.

Step 4. Repeat *Steps* 2 and 3 until no capacity-feasible merger with a positive saving can be found or until all the spokes are included in one route.

4.3. *Feeder-containership displacement procedure*

As mentioned earlier, only the largest feeder-containership is available in the savings algorithm. Therefore, the same feeder-containership displacement procedure as used in the GA procedure is applied to choosing the smallest feasible feeder-containerships.

4.4. *Greedy algorithm*

For the ULCS routing problem, a simple greedy algorithm is applied, shown as follows:

Step 1. Put all the selected hubs in List A.

Step 2. Randomly select a hub in List A as the current hub for the ULCS, and remove the hub from List A.

Step 3. The ULCS travels from the current hub to the next nearest hub among the hubs in List A. Set the next hub as the current hub and remove it from List A.

Step 4. Repeat *Step* 3 until List A is empty.

5. Computational experiments

The proposed GA features four parameters as follows: number of generations, population size, crossover probability, and mutation probability. With the aim of finding the proper values for each of these parameters, a large number of pilot tests were performed. The test problem is Experiment 3 (homogeneous feeder-containerships) in which there are 24 ports. Before embarking on rigorous testing, the initial values for each test must be empirically set. The population size, crossover probability, and mutation probability were fine-tuned as per the following method:

Step 1. The algorithm was run several times with large changes in the four parameters.

Step 2. After finding the range for each parameter that provides the best results in terms of the convergence rate and the objective function value, a number of detailed runs were performed. These runs were operated by increasing the current parameters being tested over a large range of values while fixing the other parameters to the

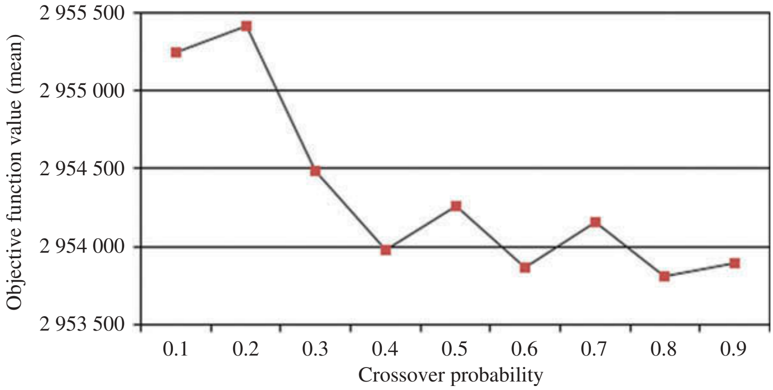

Figure 7. Test results for crossover probabilities.

previously found values. The algorithm was performed 10 times for each increased parameter value, so that it was run a total of several hundred times.

Populations of 50, 100, 150, and 200 were used for testing population sizes. The results obtained showed that the best population size is 100 with respect to the convergence rate, the final objective function value, and the computation time. The results for the effect of the crossover probability on the objective function value were obtained. The results showed that, with respect to the mean of the objective function value, 0.80 is the best value for the crossover probability, as shown in Figure 7. The results for the effect of the mutation probability on the objective function value are shown in Figure 8. These results indicate that the objective function value is higher when the mutation probability is too small or too large. If it is too small, many of the genes that would have been useful are not used; however, if the mutation probability is too large, there will be a significant random perturbation, the offspring will show less resemblance to their parents, and the algorithm will lose the ability to learn from the history of the search. In the algorithm, the mutation probability is set as 0.2.

Several numerical experiments were conducted. Both the small-sized problem (Experiments 1 and 2) and the medium-sized problem (Experiment 3) were applied to the evaluation of the performance and efficiency of the GA with local search. The mathematical model was

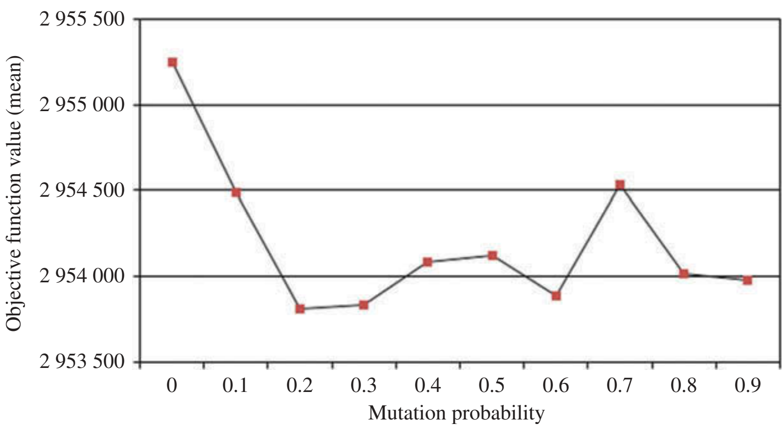

Figure 8. Test results for mutation probabilities.

Table 1. Feeder-containership-related data.

Feeder-containership type	Capacity (TEU)	Fixed cost of using ($)	Sailing cost ($/mile)	Port charge at each port ($/time)
Type 1	514	10 200	12.80	200
Type 2	582	13 250	13.00	220
Type 3	996	20 400	13.95	280
Type 4	1270	21 000	14.80	300
Type 5	1434	21 600	15.00	350
Type 6	1677	21 800	16.00	380
Type 7	2452	22 450	20.50	415

Note: For simplification, we assume that the port charges for the same type of feeder-containerships are the same at each port.

implemented and solved by CPLEX (Version 12.1; IBM Corporation, Armonk, NY, USA). All of the experiments were run on a personal computer with Intel (R) Core (TM) 2 CPU at 1.86 GHz (Intel, Santa Clara, CA, USA), 1 G of RAM and Windows XP (Microsoft, Redmond, Washington, DC, USA).

Common experimental data values for Experiments 1–4 include US$120 per mile for the travel costs of ULCSs. The feeder-containership related data are shown in Table 1.

In Experiment 1, a total of 11 ports with four hub candidates as well as the demand between the ports were generated randomly. The handling cost and port charge for ULCS vary for different hub candidates, as shown in Table 2. From Tables 3 and 4, we observe that, except for the computation time required by the mathematical model and the GA with local search, the results are the same. Because the GA with local search obtains the same results as the mathematical model in each of the 10 runs, the efficiency of the algorithm is proven.

Table 2. Handling costs and port charges for ULCSs at hub candidates.

Port number	Handling costs ($/TEU)	Port charges for ULCSs ($/time)
6	100	8000
7	90	6000
8	120	10 000
9	100	12 000

Table 3. Results of Experiment 1 from the mathematical model.

	Homogeneous		Heterogeneous	
	Results	Ship type	Results	Ship type
Objective function value	997 588.0	–	995 275.0	–
Computation time*	33 seconds	–	30 minutes	–
Selected hubs	7, 9	–	7, 9	–
Main route	7-9	–	7-9	–
Feeder route 1	7-5-3-2-1-7	6	7-5-3-2-1-7	5
Feeder route 2	9-10-11-9	6	9-10-11-9	5

Note: *Average of 10 evaluations.

Table 4. Results of Experiment 1 from the GA with local search.

	Homogeneous		Heterogeneous	
	Results	Ship type	Results	Ship type
Objective function value	997 588.0	–	995 275.0	–
Computation time*	0.05 s	–	0.55 s	–
Selected hubs	7, 9	–	7, 9	–
Main route	7-9	–	7-9	–
Feeder route 1	7-5-3-2-1-7	6	7-5-3-2-1-7	5
Feeder route 2	9-10-11-9	6	9-10-11-9	5

Note: *Average of 10 evaluations.

Table 5. Handling costs and port charges for ULCSs at hub candidates.

Port number	Handling costs ($/TEU)	Port charges for ULCSs ($/time)
8	100	8000
9	90	6000
10	120	10 000
11	100	12 000

Table 6. Results of Experiment 2 from the mathematical model.

	Homogeneous		Heterogeneous	
	Results	Ship type	Results	Ship type
Objective function value	969 025.0	–	967 247.4	–
Computation time*	100 s	–	10 h	–
Selected hubs	9, 11	–	9, 11	–
Main route	9-11	–	9-11	–
Feeder route 1	9-1-2-3-4-5-6-7-9	5	9-1-2-3-4-5-6-7-9	4
Feeder route 2	11-12-13-11	5	11-12-13-11	4

Note: *Average of 10 evaluations.

Table 7. Results of Experiment 2 from the GA with local search.

	Homogeneous		Heterogeneous	
	Results	Ship type	Results	Ship type
Objective function value	969 025.0	–	967 247.4	–
Computation time*	0.3 s	–	1.0 s	–
Selected hubs	9, 11	–	9, 11	–
Main route	9-11	–	9-11	–
Feeder route 1	9-1-2-3-4-5-6-7-9	5	9-1-2-3-4-5-6-7-9	4
Feeder route 2	11-12-13-11	5	11-12-13-11	4

Note: *Average of 10 evaluations.

Table 8. Comparison of the results obtained with the mathematical model and the GA with local search.

Problems		Mathematical model		GA with local search		
		Objective function value	Computation time*	Objective function value	Computation time* (second)	Gap (%)
Experiment 1	Homogeneous	997 588.0	33 seconds	997 588.0	0.05	0
	Heterogeneous	995 275.0	30 minutes	995 275.0	0.55	0
Experiment 2	Homogeneous	969 025.0	100 seconds	969 025.0	0.30	0
	Heterogeneous	967 247.4	10 hours	967 247.4	1.00	0

Note: *Average of 10 evaluations.

In Experiment 2, there is a transportation demand in hub candidates 8 and 9; therefore, two virtual spokes are added based on Experiment 1. The handling costs and port charges for ULCSs in different hub candidates are shown in Table 5. From Tables 6 and 7, we see that both the mathematical model and the GA with local search show the same solutions. However, the computation time of the mathematical model is significantly larger than that of the GA with local search.

Table 8 shows a comparison of the results obtained with the mathematical model and the GA with local search; this comparison demonstrates the efficiency of the GA with local search. We can also observe that the computation time increases significantly, even though only two ports are added in Experiment 2.

In Experiment 3, a total of 24 ports with six hub candidates as well as the demand between the ports were generated randomly. The handling costs and port charges for ULCSs are shown in Table 9. In this experiment, CPLEX is not capable of solving the mathematical model. Therefore, the upper and lower bounds obtained from CPLEX are compared with those obtained using the GA with local search. Ten datasets with randomly generated demand are used to conduct the experiments. Comparisons of the results are shown in Tables 10 and 11.

One can see that the proposed GA with local search reaches the upper bound for every case. The average gap between the algorithm and the lower bound is 0.93% for the homogeneous feeder-containerships and 2.21% for the heterogeneous feeder-containerships. The experiments with heterogeneous feeder-containerships yield lower total costs. This scenario is easily explained: Using more containership types provides more choices and thus lower costs.

Table 9. Handling costs and port charges for ULCSs at hub candidates.

Port number	Handling costs ($/TEU)	Port charges for ULCSs ($/time)
11	100	8000
12	90	6000
13	110	8500
14	100	9000
15	120	10 000
16	100	12 000

Table 10. Comparison of the results for homogeneous feeder-containerships.

	CPLEX (out of memory)			GA with local search		
Dataset	Upper bound	Lower bound	Computation time* (hour)	Objective function value*	Computation time* (second)	Gap (%)
1	2 953 418.0	2 949 140.0	8.0	2 953 418.0	0.8	0.14
2	3 343 303.5	3 339 388.0	8.8	3 343 303.5	0.9	0.12
3	3 294 725.2	3 272 865.9	5.7	3 294 725.5	0.8	0.66
4	3 277 593.0	3 246 179.3	9.0	3 277 593.0	0.9	0.96
5	3 372 793.5	3 360 407.4	9.3	3 372 793.5	0.9	0.37
6	3 198 724.5	3 159 234.0	6.7	3 198 724.5	0.9	1.23
7	2 846 520.0	2 806 531.0	7.0	2 846 520.0	0.8	1.40
8	2 674 022.5	2 623 658.6	9.8	2 674 022.5	0.8	1.88
9	3 056 536.5	3 019 556.8	8.3	3 056 536.5	0.9	1.21
10	2 925 668.5	2 808 821.5	6.3	2 925 668.5	0.9	1.31

Note: *Average of 10 evaluations.

Table 11. Comparison of the results for heterogeneous feeder-containerships.

	CPLEX (Out of memory)			GA with local search		
Dataset	Upper bound	Lower bound	Computation time* (hour)	Objective function value*	Computation time* (second)	Gap (%)
1	2 927 673.5	2 922 533.5	10.0	2 927 673.5	16	0.17
2	3 314 622.0	3 265 226.0	9.8	3 314 622.0	17	1.49
3	3 279 818.5	3 236 260.8	10.5	3 279 818.5	17	1.33
4	3 253 273.6	3 182 794.9	8.6	3 253 273.6	17	2.17
5	3 352 717.3	3 295 642.5	11.7	3 352 717.3	17	1.70
6	3 185 325.5	3 112 797.3	8.2	3 185 325.5	17	2.33
7	2 835 138.8	2 762 183.9	11.9	2 835 138.8	16	2.57
8	2 674 022.5	2 571 882.6	9.3	2 674 022.5	16	3.81
9	3 047 220.3	2 933 406.8	9.2	3 047 220.3	17	3.73
10	2 919 162.3	2 837 391.3	13.0	2 919 162.3	17	2.80

Note: *Average of 10 evaluations.

Experiment 4 illustrates a real case: the identities of 74 ports with 10 hub candidates were obtained from the company. The port numbers and their corresponding port names are provided in Table 12. Due to their length, the distances are not shown in this paper. The handling costs and port charges for ULCSs at hub candidates are shown in Table 13. The demand from each region port to each destination port was generated uniformly in the range of [1,50]. The data, which are related to the feeder-containerships, are the same as those in Experiment 3. The results of comparison of the GA with local search and the heuristic algorithm are shown in Tables 14 and 15. Using the GA with local search, we obtain better solutions than with the realistic algorithm in every instance.

Table 12. The port numbers and their corresponding port names in Experiment 4.

Port number	Port name	Port number	Port name	Port number	Port name
1	Dandong	26	Zhuhai	51	Gold River
2	Dalian	27	Shuidong	52	Port Alberni
3	Yingkou	28	Zhanjiang	53	Powell River
4	Jinzhou	29	Haikou	54	Nanaimo
5	Qinghuangdao	30	Yangpu	55	Chemainus
6	Longkou	31	Basuo	56	Cowichan Bay
7	Wei Hai	32	Sanya	57	Victoria BC
8	Rizhao	33	Beihai	58	Richmond CA
9	Lanshan	34	Fangcheng	59	Newport
10	Lianyungang	35	Xingang	60	Port Mellon
11	Tianjin	36	Qingdao	61	Squamish
12	Yantai	37	Shanghai	62	Bellingham
13	Zhangjiagang	38	Ningbo	63	Anacortes
14	Nantong	39	Yantian	64	Port Angeles
15	Zhoushan	40	Hong Kong	65	Port Townsend
16	Haimen	41	Long Beach	66	Everett
17	Wenzhou	42	Tacoma	67	Seattle
18	Fuzhou	43	Oakland	68	Olympia
19	Quanzhou	44	Vancouver	69	Astoria
20	Xiamen	45	Los Angeles	70	San Diego
21	Dongshan	46	Prince Rupert	71	Port Hueneme
22	Shantou	47	Kitimat	72	Port San Luis
23	Shanwei	48	Bella Coola	73	Redwood City
24	Huizhou	49	Port Hardy	74	San Francisco
25	Shekou	50	Port Alice		

Note: Ports 35 to 45 are hub candidates.

Table 13. Handling costs and port charges for ULCSs at hub candidates.

Port number	Handling costs ($/TEU)	Port charges for ULCSs ($/time)
35	124	10 300
36	111	7000
37	126	8500
38	120	12 500
39	120	9000
40	119	10 500
41	121	10 000
42	110	10 600
43	129	10 700
44	134	13 000
45	96	12 000

6. Conclusions

In this paper, a tramp ship routing problem combined with fleet deployment and a hub-and-spoke network design is investigated. To our knowledge, it shows the first study that addresses tramp shipping, using three NP-hard problems, in hub-and-spoke network design and involves the selection of hub candidates, the assignment of spokes to hubs, and containership routing between spokes and spokes, hubs and spokes, and hubs and

Table 14. Comparison of the results for homogeneous feeder-containerships.

	GA with local search		Heuristic algorithm		
Data set	Objective function value*	Computation time* (second)	Objective function value*	Computation time* (second)	Gap (%)
1	7 192 861.5	89.6	7 283 673.0	0.6	1.26
2	7 426 557.0	85.4	7 587 599.0	0.6	2.16
3	7 530 349.5	99.5	7 712 595.5	0.6	2.42
4	7 481 128.0	89.5	7 598 167.0	0.5	1.56
5	7 539 488.5	86.6	770 3961.5	0.6	2.18
6	7 460 629.5	98.3	7 552 895.0	0.6	1.23
7	7 459 311.5	102.3	7 572 497.0	0.8	1.51
8	7 347 315.5	94.6	7 465 208.5	0.7	1.60
9	7 641 939.0	97.6	7 788 788.0	0.6	1.92
10	7 347 958.0	92.6	7 462 524.0	0.6	1.56

Note: *Average of 10 evaluations.

Table 15. Comparison of the results for heterogeneous feeder-containerships.

	GA with local search		Heuristic algorithm		
Data set	Objective function value*	Computation time* (second)	Objective function value*	Computation time* (second)	Gap (%)
1	6 968 870.5	1915.6	7 188 224.5	1.0	3.15
2	7 401 221.0	1765.4	7 484 517.5	1.1	1.11
3	7 434 326.5	1556.2	7 617 122.0	1.0	2.46
4	7 377 876.0	2032.0	7 505 646.5	1.2	1.73
5	7 434 999.0	2512.5	7 595 736.0	1.1	2.16
6	7 395 682.5	1952.2	7 460 315.5	1.3	0.87
7	7 328 487.0	1755.3	7 456 301.0	0.9	1.74
8	7 237 080.5	2231.2	7 370 963.0	1.0	1.84
9	7 566 306.0	2012.5	7 692 379.0	1.5	1.67
10	7 284 965.0	1842.9	7 362 804.5	1.3	1.07

Note: *Average of 10 evaluations.

hubs. The presented model can be used as a decision support tool for making routing decisions for shipping companies that provide long-haul services under a hub-and-spoke network, especially those that use ULCSs to provide transoceanic shipping services. Because the problem is NP-hard, a GA with local search is proposed. A simple but realistic heuristic algorithm, typical of a decision support system, is also presented in detail. To evaluate the performance, the proposed GA with local search is tested on several sets of small- and medium-scale experiments. Through a comparison with the lower bound value from CPLEX, we demonstrated the efficiency of the algorithm. Through a comparison with the heuristic algorithm in the experiment of real-world data, we also observe that the GA with local search obtains better solutions in every instance.

In further research, both the delivery and pickup considerations could be included. Another study could take into account the time-window constraints. In addition, the inland transportation routing (e.g., vehicle routing from plants to spoke ports) could be integrated, rendering the problem somewhat complicated and difficult, yet still fairly interesting.

Acknowledgements

The authors are grateful for the useful comments from an associate editor and two anonymous referees. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0025714)

References

Alumur, S., and B. Y. Kara. 2008. "Network Hub Location Problems: The State of the Art." *European Journal of Operational Research* 190 (1): 1–21.

Aversa, R., H. Botter, R. Haralambides, and H. Yoshizaki. 2005. "A Mixed Integer Programming Model on the Location of a Hub Port in the East Coast of South America." *Maritime Economics & Logistics* 7: 1–18.

Brønmo, G., M. Christiansen, K. Fagerholt, and B. Nygreen. 2007. "A Multi-Start Local Search Heuristic for Ship Scheduling—A Computational Study." *Computers & Operations Research* 34: 900–917.

Brønmo, G., M. Christiansen, and B. Nygreen. 2007. "Ship Routing and Scheduling with Flexible Cargo Sizes." *Journal of the Operational Research Society* 58: 1167–1177.

Brønmo, G., B. Nygreen, and J. Lysgaard. 2010. "Column Generation Approaches to Ship Scheduling with Flexible Cargo Sizes." *European Journal of Operational Research* 200: 139–150.

BRS. 2008. *Shipping and Shipbuilding Markets*. Neuilly sur Seine: Barry Rogliano Salles.

Christiansen, M., K. Fagerholt, B. Nygreen, and D. Ronen. 2007. "Maritime Transportation." In *Handbooks in Operations Research and Management Science: Transportation*, vol. 14, edited by C. Barnhart and G. Laporte, 189–284. Elsevier: Amsterdam.

Christiansen, M., K. Fagerholt, and D. Ronen. 2004. "Ship Routing and Scheduling, Status and Perspectives." *Transportation Science* 38 (1): 1–18.

Clarke, G., and J. Wright. 1964. "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points." *Operations Research* 12: 568–581.

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 (4): 259–268.

Gelareh, S., S. Nickel, and D. Pisinger. 2010. "Liner Shipping Hub Network Design in a Competitive Environment." *Transportation Research Part E* 46: 991–1004.

Gelareh, S., and D. Pisinger. 2011. "Fleet Deployment, Network Design and Hub Location of Liner Shipping Companies." *Transportation Research Part E* 47: 947–964.

Hsu, C., and Y. Hsieh. 2007. "Routing, Ship Size, and Sailing Frequency Decision-Making for a Maritime Hub-and-Spoke Container Network." *Mathematical and Computer Modeling* 45: 899–916.

Imai, A., E. Nishimura, S. Papadimitriou, and M. Liu. 2006. "The Economic Viability of Container Mega-Ships." *Transportation Research Part E* 42 (1): 21–41.

Imai, A., K. Shintani, and S. Papadimitriou. 2009. "Multi-Port vs. Hub-and-Spoke Port Calls by Containerships." *Transportation Research Part E* 45: 740–757.

Karaoglan, I., and F. Altiparmak. 2011. "A Hybrid Genetic Algorithm for the Location-Routing Problem with Simultaneous Pickup and Delivery." *Industrial Engineering & Management Systems* 10 (1): 24–33.

Kjeldsen, K. H. 2011. "Classification of Ship Routing and Scheduling Problems in Liner Shipping." *Information Systems and Operational Research (INFOR)* 49 (2): 139–152.

Korsvik, J. E., K. Fagerholt, and G. Laporte. 2010. “A Tabu Search Heuristic for Ship Routing and Scheduling.” *Journal of the Operational Research Society* 61: 594–603.

Korsvik, J. E., K. Fagerholt, and G. Laporte. 2011. “A Large Neighborhood Search Heuristic for Ship Routing and Scheduling with Split Loads.” *Computers & Operations Research* 38: 474–483.

Malliappi, F., H. A. Bennell, and C. N. Potts. 2011. “A Variable Neighborhood Search Heuristic for Tramp Ship Scheduling.” *Lecture Notes in Computer Science* 6971: 273–285.

Meng, Q., and S. Wang. 2011. “Liner Shipping Service Network Design with Empty Container Repositioning.” *Transportation Research Part E* 47: 695–708.

Meng, Q., and T. S. Wang. 2010. “A Chance Constrained Programming Model for Short-Term Liner Ship Fleet Planning Problems.” *Maritime Policy & Management* 37 (4): 329–346.

Norstad, I., K. Fagerholt, and G. Laporte. 2011. “Tramp Ship Routing and Scheduling with Speed Optimization.” *Transportation Research Part C* 19: 835–865.

Ronen, D. 1983. “Cargo Ships Routing and Scheduling: Survey of Models and Problems.” *European Journal of Operational Research* 12 (2): 119–126.

Ronen, D. 1993. “Ship Scheduling: The Last Decade.” *European Journal of Operational Research* 71: 325–333.

Suzuki, A., and H. Yamamoto. 2012. “Solving Facility Rearrangement Problem Using a Genetic Algorithm and a Heuristic Local Search.” *Industrial Engineering & Management Systems* 11 (2): 170–175.

UNCTAD. 2010. *Review of Maritime Transport 2010*. Paper presented at United Nations Conference on Trade and Development (UNCTAD). New York: UNCTAD.