Network Design and Allocation Mechanisms for Carrier Alliances in Liner Shipping

Richa Agarwal, Özlem Ergun
School of Industrial and Systems Engineering
Georgia Institute of Technology, Atlanta, GA 30332-0205, USA
ragarwal@isye.gatech.edu, oergun@isye.gatech.edu

Many real world systems operate in a decentralized manner, where individual operators interact with varying degree of cooperation and self motives. In this paper, we study transportation networks that operate as an alliance among different carriers. In particular, we study alliance formation among carriers in liner shipping. We address tactical problems such as the design of large scale networks (which result from integrating the service networks of different carriers in an alliance) and operational problems such as the allocation of limited capacity on a transportation network among the carriers in the alliance. We utilize concepts from mathematical programming and game theory and design a mechanism to guide the carriers in an alliance to pursue an optimal collaborative strategy. The mechanism provides side payments to the carriers, as an added incentive, to motivate them to act in the best interest of the alliance while maximizing their own profits. Our computational results suggest that the mechanism can be used to help carriers form sustainable alliances.

Subject classifications: Mechanism design; Allocation of benefits; Liner shipping; Alliance formation.
Area of review: Transportation.

1. Introduction

In transportation and supply chain logistics (as in other wide variety of businesses such as telecommunications, health-care etc.), companies and business processes collaborate to improve performance. Sea-Land and Maersk began sharing ships in the Atlantic and Pacific oceans in 1990. Since then alliances have become increasingly common among sea cargo carriers. According to N. Hinggorani and Tornqvist (2005), in near future the top 10 carriers will control about 80% of the sea transportation market with the next 20 carriers controlling about 15% of the market. Similarly, airline carriers collaborate and share seat capacity through the use of code-sharing to increase asset utilization. In trucking, several web based collaborative networks save time and dollars for their customers by integrating the network of many shippers and carriers.

The system wide collaboration perspective provides opportunities for increased profitability that are impossible to achieve with an internal focus only. Collaborations have helped companies to reduce costs, decrease lead times, increase asset utilization and improve overall service levels. Since owning an asset (such as an airplane or a ship) involves large capital investment (usually in millions of US dollars) and the cost of idling an asset runs in tens of thousands of dollars per day, carriers collaborate and form alliances to share capacity on assets and infra-structural setup and capital costs.

In this paper, we study large scale transportation networks that operate as an alliance among carriers. Our study is motivated by the service network design problem in the liner shipping industry. Liner shipping is a form of sea cargo transportation and it involves carrying cargo stored in containers on regularly scheduled service routes. According to Barry Rogliano Salles- AlphaLiner (2006), the total capacity deployed on global liner routes increased by 77.4% between January 2000 and January 2006. Liner services involve high fixed costs and administrative overhead because they promise to depart on a predetermined schedule regardless of whether the ship is full. The number
of ships required for a given liner service route is determined principally by the frequency required on the service route. For example, a weekly liner service between New York and Hamburg may require four ships to maintain the necessary frequency. In liner shipping, given a set of ports, a fleet of ships and a set of cargo to be delivered the service network is designed by creating the ship routes, i.e. the sequence of port visits by the given fleet of ships. In general it is assumed that the ships move in cycles, referred to as service routes from one port to another following the same port rotation for the entire planning horizon. The service network is utilized to deliver the profit maximizing cargo. Carriers decide which cargo to accept or reject for servicing and which path(s) to use to deliver the selected cargo. The cargo is allowed to travel on ships on multiple service routes before reaching its final destination. A port where cargo is transferred from one ship to another, for further transportation, is referred to as a transshipment port.

Collaboration and alliance formation is a common phenomenon among liner shipping operators. Moreover, the shipping industry in many nations including the United States has enjoyed anti-trust impunity because of the widely accepted fact that this industry is highly capital intensive and collaboration among carriers helps to provide regular service between ports. Carriers used conferences, as a means for curbing competition and controlling tariff rates in the market, for over a century. The first conference was formed in 1875 on a route between the United Kingdom and Calcutta, India. More recently, carriers form strategic alliances by pooling their fleets and operating them together to share capacity on the ships. In such alliances, the carriers decide on a set of service routes, assign their ships for operating the chosen routes and allocate each ship’s capacity among the alliance members. As a result of these alliances and agreements, shipments arranged through one carrier may actually be moved by a ship operated by another carrier and alliance members can offer higher sailing frequencies than would be possible using only their own ships. Alliances are most common on deep sea routes such as the Asia-North America route that require a bigger commitment in terms of assets (ships) from carriers. In the mid 1990s an estimated 60% of the total global liner capacity was accounted by alliances.

Alliance formation among carriers poses various challenges as well. As carriers form alliances by pooling their ships and integrating their networks, a large scale optimization problem needs to be solved to design the overall service network. Furthermore, in an alliance, carriers work in collaboration with each other, however each carrier’s individual goal remains to be the maximization of his own benefit. Hence, for forming sustainable alliances, the task is not only to design an efficient service network but also to provide mechanisms to manage the interactions among the alliance members and share the benefits and costs of the alliance in such a way that all carriers are motivated to collaborate. This paper will focus on developing such mechanisms.

Problems where a number of selfish participants interact to obtain greater benefits occur in a wide variety of fields such as internet routing, auctions etc. The field of cooperative game theory provides a set of mathematical tools and concepts to study the “fair” distribution of costs and benefits among the participants in such collaborations. There exists literature that studies cooperative game theory for classical routing problems. For example, Sánchez-Soriano (2006) develops an allocation mechanism for a transportation game; Hamers (1997) and Granot et al. (1999) study delivery games associated with the Chinese postman problem; M. Gothe-Lundgren and Varbrand (1996) and Engevall et al. (2004) study the vehicle routing game with homogeneous and heterogeneous fleets respectively and allocate costs among the members of an alliance based on cooperative game theory concepts. Network related games such as the network design game Kubo and Kasugai (1992), the assignment game Shapley and Shubik (1972) and facility location game Goemans and Skutella (2000) have also been studied in the literature. Some studies on combinatorial games such as the bin packing game Dror (1990) and the knapsack game Dror (1990) are also available.

Qualitative studies regarding alliance formation among carriers in the liner shipping industry are also available in the literature. Midoro and Pittos (2000), Slack et al. (2002), Ryoo and Thanopoulou
Song and Panayides (2002) discuss the importance of strategic alliances in liner shipping. In particular, Midoro and Pitto (2000) studies the factors that led to the advent of strategic alliances among liner carriers 30 years ago, and the changes in the industry in the 1990s (for example, the increase in demand due to globalization) that made the previous alliances inadequate and called for a new generation of strategic partnerships. It suggests that differentiation in the contribution of each member, depending on their core competencies, can lead to alliances that deliver more than the sum of individual contributions and that alliances with fewer members or ones that are led by a dominant partner are more likely to succeed. Slack et al. (2002) provides industry data to support the claim that alliances lead to intensification in service frequency and an increase in ship sizes. It points out that as a result of alliances, carriers are becoming more similar (with similar service routes, serving the same markets and employing comparable ships) and although individual carriers who form alliances serve more ports than before, the total number of ports served by the overall industry remains remarkably constant. Ryoo and Thanopoulou (1999) studies the progression of collaborations from consortia, which are route-based forms of cooperation, to alliances, which cooperate on a global level, among Asian carriers. It argues that the reasons behind this trend are the flexibilities and synergies provided by alliances with a global perspective. Song and Panayides (2002) makes use of cooperative game theory and provides a quantitative study to analyze liner shipping alliances by considering two small examples (involving 3 ports and 2-3 carriers). It allocates costs and benefits among the alliance members in proportion to the shipping capacity provided by them. However, as we will see later, in many situations a proportional allocation of benefits is not guaranteed to provide a payoff to the alliances members to sustain the existence of the collaboration.

Thus in literature, though a few references are available regarding qualitative study of liner shipping alliances, a rigorous quantitative study is missing. In this paper, we provide a mathematical framework for such a quantitative analysis. As carriers form alliances by pooling their ships and integrating their networks, the maximum overall revenue that an alliance can generate can be obtained by replacing individual carriers with one large carrier, with a fleet equal to the combined fleet of the individual carriers’ and a demand structure equal to the combined demand of all carriers. We refer to this problem as the optimization problem for the grand coalition or the centralized problem and to its solution as the collaborative optimal solution. Though working in collaboration, carriers cannot be assumed to follow the collaborative optimal solution but their own self-interests. For example, on collaborative routes the “resource,” e.g. capacity on a ship, belongs to some carrier who does not allow other carriers to freely obtain “benefits” from using it. The fair distribution of costs and benefits among the alliance members is an intriguing question and in transportation networks, in particular liner shipping, sharing benefits and costs among the carriers generally translates into exchanging asset capacity on service routes. One way to regulate capacity exchanges among the carriers is to assign suitable capacity exchange costs so that the carrier who owns the capacity on a ship is motivated to sell the capacity to a carrier who can utilize it to deliver cargo.

We design membership mechanisms to allocate benefits and costs among the members of an alliance. To help the overall alliance achieve its maximum potential revenue, we provide incentives to the carriers to pursue the collaborative optimal solution. The revenue obtained from delivering cargo on the collaborative routes directly are often not enough to motivate individual carriers to behave in the best interest of the alliance. We compute capacity exchange costs on the edges of the network so that given these costs a carrier who has un-used capacity on his ships is motivated to sell that capacity to a carrier who can use it to transfer his own cargo. Thus capacity exchange costs provide side payments to the carriers in addition to the revenue generated by delivering cargo. These payments are made by the carriers who utilize a ship’s capacity, depending on the amount of utilization, to the carrier who owns the ship. To compute these costs we model the individual
behavior of a carrier in the alliance as a linear program. The mechanism drives each individual carrier’s linear program towards the collaborative optimal solution using inverse programming techniques. We test the performance of our mechanism on randomly generated instances simulating real life with up to 10 ports and 50 ships. We utilize notions from game theory to test the “fairness” of the allocations made by our mechanism. Our computational results suggest that the mechanism can be used to help carriers form sustainable alliances.

The rest of the paper is organized as follows. In Section 2 we review in detail the reasons that motivate carriers in liner shipping to collaborate and form alliances. In Section 3 we briefly review some relevant game theoretic and inverse programming concepts. In Section 4 we provide a formal description of our problem and in Section 5 we present the details of our solution approach. We present our computational results in Section 6 and conclusions in Section 7.

2. Why Collaborate?

In order to position themselves better against their competitors carriers usually rely on good customer service (shorter transit times, higher frequency of service) and competitive prices. However, many factors drive carriers to adopt solutions outside of their traditional business practices and collaborate with their competitors. Next, we briefly describe some of these factors. These have been studied by Yen (1994) in detail.

1. Liner shipping is a capital intensive industry and owning a ship alone involves large capital investment (typically $20,000/TEU and deep sea ships are up to 8000 TEU in capacity) for carriers. Further, there is low differentiation between the service provided by the carriers as all of them transfer cargo using ships. Thus, carriers collaborate and form alliances to reduce and share capital costs. In extreme cases of collaboration carriers form alliances with carriers who do not own any ships or “NVO”s (non-vessel operator). NVOs compensate ship owners for using capacity on their ships and for ship owning carriers they act like shippers with big demands.

2. Bigger ships provide economies of scale as the construction cost, the cost of operation and the cost of maintenance on a ship does not increase in proportion to the size of the ship. However, the capacity on a ship is perishable, as once the ship leaves the port the capacity becomes unusable until it reaches a loading port again. Alliances provide carriers with opportunities to deploy bigger ships, thus achieving reduction in cost via economies of scale and higher utilization of space, by catering to the demand of multiple carriers.

3. Good frequency of service is essential for a carrier to achieve higher market share. The “just-in-time” inventory management systems deployed by many businesses demands timely and frequent transportation service. Thus, most carriers have at least one departure each week from each port visited on a service route. However, this requires big commitment in terms of the number of ships from a carrier. Pooling ships together to form strategic alliances allows carriers to group ships with similar characteristics, independent of ownerships, to offer same service every week on jointly operated routes, by deploying compatible ships on a service route. For example according to 2005
data OOCL (2005), the Grand Alliance maintains its strong hold on the East-West trade lanes by offering faster transit times and high frequency service with the help of 112 ships that provide a total capacity of 640,000 TEUs on these trade lanes.

4. Alliances help carriers to explore new markets and enhance their global reach. Transshipments play a vital role in enhancing global markets. For example in Figure 1 let carrier A and carrier B operate cycle C₁ and C₂ respectively. To obtain benefit from an emerging market from port P₁ to port P₃ they can form an alliance by offering space on their respective cycles and by transshipping at port P₂. Thus both of them can expand their market reach without deploying any additional ships. This is well suited for shippers as well as they can avoid dealing with multiple carriers. In 2005 MISC, MOL, NYK, OOCL and PIL started two collaborative routes to cover China, Singapore and New Zealand. The strategic alliance allowed them to extend their services to new destinations thereby enhancing their role in those regions and generating new opportunities for mutual growth.

To summarize, though the notion of cooperation runs contrary to the concept of perfect competition, carriers form alliances to realize economies of scale, extend customer base, increase asset utilization, regulate traffic and fix prices while providing customers with more frequent sailings and faster transit times. Given that the decisions regarding the alliance routes have to be taken together and all partners have their own interests in mind, carriers need to identify suitable partners with good synergies. Since the advent of strategic alliances, many alliances have restructured and reorganized themselves. In many cases collaborators have parted ways because they could not align their interests properly. Figure 2 depicts some of the trends of consolidation in 2007 as compared to those in 1995.

3. Basic Concepts

In this section, we provide some basic game theory and inverse programming definitions and concepts relevant to this paper.
A cooperative game is defined by a set $\mathcal{N} = \{1, 2, \ldots, n\}$ of players and a characteristic function $\text{opt} : 2^{\mathcal{N}} \rightarrow \mathbb{R}$. $\text{opt}(S)$ maps a value to every subset/coalition $S \subset \mathcal{N}$, interpreted as the total gain the members of $S$ can achieve by cooperating. We assume that $\text{opt}(\emptyset) = 0$. The set $\mathcal{N}$ itself is referred to as the grand coalition. The central problem in cooperative game theory is how to allocate the total gain $\text{opt}(\mathcal{N})$ among the individual players $k \in \mathcal{N}$ in a fair way. We denote an allocation/payoff vector by $x = \{x^1, \ldots, x^n\} \in \mathbb{R}$, where $x^k$ refers to the payoff made to player $k$.

The notion of core is one of the most prominent and widely accepted notions of fair allocation of costs and benefits in cooperative game theory and it is similar to a Nash equilibrium in non-cooperative game theory. An allocation of benefits is said to be in the core if the sum of the payoffs over all players is their maximum attainable profit (budget balance property) and no subset of players can collude and obtain a better payoff for its members (stability property). Mathematically, a payoff vector $x$ is said to be in the core if:

\begin{align}
\sum_{k \in \mathcal{N}} x^k &= \text{opt}(\mathcal{N}) \quad (1) \\
\sum_{k \in S} x^k &\geq \text{opt}(S) \quad \forall S \subset \mathcal{N} \quad (2)
\end{align}

An allocation in the core, helps the grand coalition to be perceived as fair and not be threatened by its sub-coalitions. However, a payoff allocation in the core represents a very strong type of stability and does not exist in many settings. For a detailed discussion of core allocations in collaborative settings we refer the reader to Young (1985).

Another notion from game theory literature that is relevant to this paper is that of mechanism design. The traditional mechanism design problem concerns a set of players $\mathcal{N} = \{1, 2, \ldots, n\}$ who collaboratively choose an outcome $\bar{o}$ from a set $\mathcal{O}$ of possible outcomes. Each player $k$ has a preference relation for different outcomes $\bar{o}$ in the set $\mathcal{O}$, denoted by $v^k(\bar{o})$, to quantify his valuation of outcome $\bar{o}$. The goal is to design an algorithm that chooses an outcome, $\bar{o} \in \mathcal{O}$ and an n-tuple of side payments {$s^1, \ldots, s^n$} such that the total payment, $x^k$, to player $k$ is $x^k = v^k(\bar{o}) + s^k$. The total payment is what each individual player aims to optimize. Intuitively, a mechanism solves a given problem by providing side payments to players to assure that the required output occurs, when players choose their strategies so as to maximize their own selfish profits. Mechanism design has been used successfully for regulating the interactions among collaborative partners in many different applications. For a detailed discussion of mechanism design we refer the reader to Nisan and Ronen (2001).

We also make use of inverse optimization techniques in this paper. Next, we briefly describe a general inverse optimization problem. Let $P$ be an optimization problem with a given cost vector $c$ and a feasible solution $f$. An inverse optimization problem for the ordered pair $(P, f)$ is the problem of perturbing the cost vector $c$ to obtain a cost vector $d$ so that $f$ is an optimal solution of the problem $P$ with cost vector $d$. Thus, a typical optimization problem identifies the value of variables given the values of model parameters (cost coefficients, the right hand side vector etc.) whereas an inverse optimization problem identifies the value of model parameters that makes the given feasible solution optimal. Inverse programming techniques have been used in a variety of fields ranging from geophysical sciences Tarantola (1987), portfolio optimization Dembo and Rosen (1999) to traffic equilibrium Dial (2000). Ahuja and Orlin (2001) provides a unified framework for studying many inverse optimization problems on networks such as the shortest path, assignment and minimum cut problems.

4. Problem Definition

We now present a formal definition of our problem. Let $V$ denote the set of ports (also referred to as nodes) in the network and $\mathcal{N} = \{1, 2 \ldots n\}$ denote the set of carriers. We assume that all ships
are identical with $T$ units of capacity and for a carrier $k \in \mathbb{N}$, $N^k$ represents the number of ships in his fleet. For a carrier $k$, each demand is characterized by its origin $(o)$ - destination $(d)$ pair, the maximum demand that can arise between the given $(o,d)$ pair, $D^{(o,d,k)}$, and the revenue obtained by satisfying one unit of demand, $R^{(o,d,k)}$. $(o,d,k)$ is used to identify a demand from $o$ to $d$ for carrier $k$ and the demand set of carrier $k$ is identified by $\Theta^k$. Let $N = \sum_k N^k$ and $\Theta = \bigcup_k \Theta^k$, then in an alliance formed by pooling ships and consolidating demand, the carriers face the following set of problems:

1. Together the members of the alliance need to design the alliance’s service network. For this, they need to decide on a set of service routes (say $\mathcal{C} = \{C_1, \cdots, C_r\}$) to operate utilizing their ships. Also, they need to decide on the set of cargo (say $\Theta \subset \Theta$) to deliver and the paths to use to deliver the selected cargo.

2. The members of the alliance need to decide on how to realize the service routes in $\mathcal{C}$. That is, the number of ships that each carrier should assign to the service routes in $\mathcal{C}$ must be determined.

3. Each carrier $k$ needs to compute the valuation, $v^k$, of the solution given by $(\mathcal{C}, \Theta)$, by determining the costs incurred from operating his ships and the revenue generated from delivering his demands.

4. For a given collaborative solution $(\mathcal{C}, \Theta)$, the valuation $v^k$ alone is not enough to guarantee that carrier $k$ will route his cargo and share capacity as determined by the collaborative solution. Thus the alliance needs to put in place a mechanism that provides the right incentives to the carriers in order for them to act as prescribed by the collaborative solution. In this paper, we assume that the incentives are given in the form of side payments, $\{s^1, \cdots, s^n\}$, such that the total payment to carrier $k$ is $x^k = v^k + s^k$. Furthermore, the side payments are the net sum of the capacity exchange costs a carrier pays to and receives from other carriers.

5. **Solution Strategy**

In this section, we propose a set of algorithms for resolving the above problems faced by the alliance members. Before providing the details, we first present an outline of our solution strategy. The goal of an individual carrier is to design a service network which maximizes his profit. However, since he is working in collaboration with other carriers, a network that generates maximum overall revenue for all the carriers is selected. Clearly, such a network can be obtained by replacing the individual carriers’ fleet by $N$, the combined fleet of all the carriers, and the individual demand sets by $\Theta$, the combined demand of all the carriers, and then solving a network design problem on this input. We use the mathematical model and solution strategy described in Agarwal and Ergun (2007b) to solve this optimization problem and obtain a collaborative optimal solution $\text{opt}(N) = (\mathcal{C}, \Theta)$. For ease of understanding and completeness, we review the mathematical model of Agarwal and Ergun (2007b) briefly in Section 5.1.

It is non-trivial for the alliance to realize the solution given by the optimization algorithm. In a carrier alliance such as the one addressed in this paper, it is not reasonable to assume that there exists a fully centralized body that can operate the combined fleet, make the overall cargo accept-reject and routing decisions and hence gather the total profit and then allocate it among the members of the alliance in a fair manner. A more realistic model of operation is to assume that once the collaborative optimal service routes and the ships to operate on them are decided centrally, then the carriers individually operate their ships incurring the operational costs and make their own cargo accept-reject and routing decisions determining the revenue they earn. In such a partial decentralized system the challenges are (i) to design a mechanism that regulates the interactions among the carriers, that is exchange of capacity on each others’ ships; and (ii) to provide incentives so that the carriers’ individual accept-reject and routing decisions are as prescribed by the collaborative optimal solution.
One straightforward way to handle the exchange of capacity among the carriers is to use a proportional space allocation algorithm assigning each carrier a capacity on a network edge in proportion to the capacity provided by the carrier to that edge. However, such a simple allocation algorithm does not guarantee to provide outcomes that can achieve the generation of the maximum possible profit, $\text{opt}(\mathcal{N})$. To see this consider a simple network with two ports $P_1$ and $P_2$ such that it takes a ship one week to reach from one port to the other. Now let two carriers $A$ and $B$ with one ship each of 1000 TEU capacity form an alliance to operate a service route with weekly frequency between the two ports. Let $A$ have 700 TEUs of demand and $B$ have 300 TEUs of weekly demand from $P_1$ to $P_2$. Assume that a unit amount of revenue can be generated by satisfying any demand and that ship operation costs are negligible. Then a proportional space allocation algorithm would assign a capacity of 500 TEUs to each carrier and would be able to generate only 800 units of weekly revenue whereas the optimal solution to the problem is 1000 units of weekly revenue.

The valuation of solution $\text{opt}(\mathcal{N})$ is calculated for each carrier by calculating the revenue generated by him and the costs incurred by him. The valuation obtained from solution $\text{opt}(\mathcal{N})$ however is not guaranteed to provide enough motivation for a carrier to act according to the schedule $\text{opt}(\mathcal{N})$. To provide this guarantee incentives in the form of side payments must be provided to the carriers.

To handle both of the challenges described above, we suggest that the centralized authority also determines capacity exchange costs on each edge of the service network and then lets the carriers make cargo accept-reject and routing decisions, buying and selling capacity along the way at the given prices. That is, we determine side payment as the net sum of the capacity exchange costs a carrier receives from others for utilizing capacity on his ships and pays to others for utilizing their capacity.

Computation of capacity exchange costs is however non-trivial. In Section 5.2 we show that the assignment of different carriers’ ships to various selected cycles reduces to a generalized assignment problem in our setting. For a carrier, given an assignment of his ships to the cycles in $\overline{C}$, the computation of the fraction of capacity that the carrier owns on an edge is explained in Section 5.3. We denote by $\text{cost}_e$ the cost of using one TEU capacity on edge $e$. In other words, if a carrier provides capacity on an edge $e$ then for a unit utilization of capacity on edge $e$ he will charge other carriers $\text{cost}_e$ times the fraction of the capacity he owns on edge $e$. Note that the optimal solution $\text{opt}(\mathcal{N})$ provides a flow vector $\overline{f}$ on the edges of the network. We provide a linear program that models the behavior of each carrier $k$ for making cargo accept-reject and routing decisions, equivalently determining his optimal flow vector $\overline{f}_k$. Finally, we use inverse optimization techniques to determine the cost vector, $\text{cost}_e$, such that the optimal flow vector $\overline{f}_k$ for each carrier $k$ is as prescribed by the collaborative optimal flow vector $\overline{f}$. That is, the vector of side payments determined by the capacity exchange costs makes the solution $\text{opt}(\mathcal{N})$ attractive to all the carriers.

We now provide the details of our solution strategy.

### 5.1. Network Design

In liner shipping, the set of service routes determines which paths can be used to deliver the cargo. The delivered cargo and the paths chosen to deliver the cargo determine the revenue that can be generated and hence determine the profitability of the service network. Thus, these two problems are highly inter-dependent and it is important that they be studied in an integrated framework. In earlier research Agarwal and Ergun (2007b), we formulated the simultaneous ship scheduling and cargo routing problem as a mixed integer linear program. To facilitate further discussion, we briefly review the model and the solution strategies described in Agarwal and Ergun (2007b).

Let $G = (V, E)$ be a directed space-time network with vertex set $V$ and edge set $E$. Each vertex $v \in V$ represents a port on a day of the week. That is, for each port $p$ seven vertices are created in $V$. We also assume that when we refer to a commodity $(o, d, k)$, origin node $o$ also holds information regarding the day of the week the demand arises. The network $G = (V, E)$ contains three types
of edges: (i) Ground edges representing for a ship an over-night stay at a port and for cargo an overnight stay at a port either on ground or on a ship before continuing further; (ii) Voyage edges representing the movement of ships and cargo from one port to another at a given speed; (iii) Fictitious edges, \((d, o)\), for all demand triplets \((o, d, k)\) \(\in \Theta\) representing the fictitious flow of commodity \((o, d, k)\) from its destination to origin and enabling us to view the flow of commodity \((o, d, k)\) in the network as a circulation. Let us denote the set of all ground edges by \(E_g\), the set of all voyage edges by \(E_v\), and the set of all fictitious edges by \(E_f\). Let \(E = E_g \cup E_v \cup E_f\). We also use the following additional notation: \(\text{InEdges}(v)\) denotes the set of incoming edges into vertex \(v\) and \(\text{OutEdges}(v)\) denotes the set of outgoing edges from vertex \(v\); for an edge \(e = (u, v)\), \(\text{tail}(e)\) denotes vertex \(u\) and \(\text{head}(e)\) denotes vertex \(v\).

For a set of carriers and a given network, we now present the mixed integer programming model, as described in Agarwal and Ergun (2007b), for the centralized problem \(CP\). For the alliance, \(CP\) determines an optimal set of service routes to operate, the set of cargo to deliver and the paths to deliver the selected cargo on, simultaneously. Let \(C\) denote the set of all feasible service routes. For \(C \in \mathcal{C}\), \(L_C\) denotes the number of ships required to maintain the minimum required frequency on \(C\) and \(\text{Cost}_C\) denotes the cost of operating route \(C\). \(CP\) has two sets of variables: binary variables \(x_C\) for \(C \in \mathcal{C}\) to denote if route \(C\) is operated and non-negative continuous variables \(f^{(o,d,k)}_e\) to represent the flow of cargo from origin \(o\) to destination \(d\) due to carrier \(k\) on edge \(e\).

\[
CP: \quad r(\text{opt}(\mathcal{N})) = \max \sum_{(o,d,k) \in \Theta} f^{(o,d,k)}_{e(\theta,o)} R^{(o,d,k)} - \sum_{C \in \mathcal{C}} \text{Cost}_C x_C
\]

such that

\[
\sum_{e \in \text{InEdges}(v)} f_e^{(o,d,k)} - \sum_{e \in \text{OutEdges}(v)} f_e^{(o,d,k)} \leq 0 \quad \forall v \in V, \forall (o, d, k) \in \Theta
\]

\[
\sum_{(o,d,k) \in \Theta} f_e^{(o,d,k)} - \sum_{\{C \in \mathcal{C} \mid e \in C\}} T x_C \leq 0 \quad \forall e \in E
\]

\[
\sum_{(o,d,k) \in \Theta} f_e^{(o,d,k)} \leq D^{(o,d,k)} \quad \forall (o, d, k) \in \Theta
\]

\[
\sum_{C \in \mathcal{C}} L_C x_C \leq N
\]

\[
x_C \in \{0, 1\} \quad \forall C \in \mathcal{C}\) and \(f_e^{(o,d,k)} \geq 0 \quad \forall e \in E, \forall (o, d, k) \in \Theta.
\]

In the above formulation, the objective function (3) maximizes the net profit by subtracting the sum of operating costs from the revenue generated. Constraints (4) are flow balance constraints at each vertex of the network. They ensure that the total flow into vertex \(v\) of each demand \((o, d, k)\) is at most the total flow out of it for that demand. Note that if these inequalities hold for each vertex \(v \in V\), then in fact they must all hold with equality, there by implying flow conservation at each vertex. Constraints (5) and (6) are capacity constraints on the edges. Constraints (5) require that the flow on an edge must be less than the capacity of that edge. Capacity of an edge is determined by the sum of capacities of ships assigned to that edge. Constraints (6) model that the total flow from an origin node to a destination node must be less than the demand at the destination node. Constraint (7) requires that we do not use more ships than we have available. Finally we have integrality restriction on the \(x_C\) variables and non-negativity restriction on the flow variables. Integrality restriction on the \(x_C\) variables together with constraint (7) ensures that the minimum required frequency on an operated route will be maintained. Note that if a service route \(C\) is operated, i.e. \(x_C = 1\), then the model dedicates \(L_C\) ships to route \(C\), to maintain the required frequency.

The linear program given by (3)-(8) contains a large number of variables even for moderate size problems. The large size of the model is a direct result of the exponential number of possible feasible service routes. Furthermore, each demand adds a set of flow variables to the model. An useful observation, however, is that if we determine the set of service routes to be operated, i.e.
given non-negative values $\overline{\pi}$ satisfying fleet availability constraint (7), model (3)-(8) reduces to a multicommodity flow problem where each demand is considered as a different commodity.

In Agarwal and Ergun (2007b), we studied CP in detail and showed that the decision version of the problem is NP-hard by a reduction of the well known NP-complete problem, 0-1 Knapsack into CP. We developed and computationally tested three different heuristics and linear programming based algorithms to solve the problem. First, we designed a simple greedy heuristic that selects good service routes one by one and then picks the demand to be satisfied. Next, we developed a column generation based algorithm that generates a pool of good service routes and then selects the best service routes among these while routing the demand in the network. Finally, we developed a more sophisticated Benders decomposition based algorithm. The Benders procedure decomposes the network design problem into a master problem that generates service routes and a subproblem that solves the cargo routing problem. Our computational experience showed that the latter algorithm is very efficient and effective in solving large scale problems with up to 20 ports and 200 ships. See Agarwal and Ergun (2007b) for details of the model, algorithms, and computational results.

5.2. Valuation of the Schedule

For a carrier, the valuation of the collaborative optimal solution $\text{opt}(N)$ is determined by calculating the revenue generated by him and the costs incurred by him. Even when working in an alliance each carrier interacts with his own customers (shippers) and collects the revenue from them if their demand is satisfied. Thus the revenue generated by carrier $k$ is calculated by summing over the revenue generated by satisfying demand $(o,d,k)$ such that $(o,d,k) \in \Theta \cap \Theta_k$. Similarly, each carrier pays for maintaining and operating his ships on the collaborative routes. To compute the costs incurred, a carrier first needs to know the assignment of his ships to the selected routes. There exists a variety of ways to assign carriers’ ships to the selected routes. Given that these assignments directly impact the costs incurred by the individual carriers, the manner in which they are made can potentially have a fundamental impact on the decisions of the carriers and how they view the fairness of the collaborative solution. In the rest of this section, we present a model for assigning ships of various carriers to the routes selected in the collaborative optimal solution.

Let $y^k_{C_j}$ represent the number of ships carrier $k$ assigns to route $C_j$ and $u^k_{C_j}$ represent the utility he obtains by assigning one ship to route $C_j$. Then the problem of assigning ships over all the carriers to the set of selected service routes reduces to a generalized assignment problem, which we refer to as the ship assignment problem SAP.

$$\text{SAP : max } \sum_{k \in N, C_j \in C} u^k_{C_j} y^k_{C_j} \tag{9}$$

such that

\[ \sum_{k \in N} y^k_{C_j} = L_{C_j} \quad \forall C_j \in \overline{C} \tag{10} \]

\[ \sum_{C_j \in C} y^k_{C_j} \leq N^k \quad \forall k \in N \tag{11} \]

\[ y^k_{C_j} \text{ int} \quad \forall k \in N, \forall C_j \in \overline{C}. \tag{12} \]

The generalized assignment problem is a well studied problem in the literature and is shown to be NP-hard. Many algorithms (heuristic as well as exact) have been proposed to solve it effectively Savelsbergh (1997).

It is also non-trivial to decide how to assign values to the utility vector, $u$. Note that the cost of assigning a ship to a service route (cost of operating the ship, port visit cost, overnight stay cost at a port etc.) remains the same for all carriers. However, the value that a carrier obtains from a service route (that is the revenue generated by the flow of the carrier on the service route) is different for different carriers. Thus we use only the flow of a carrier on a service route to compute
his utility of assigning a ship to the service route. We use two heuristic methods - first, we take the
tility of assigning a ship to a service route for a carrier to be proportional to the flow of the carrier
on that service route and second, we take it to be proportional to the revenue generated by the
flow of the carrier from that service route. If a demand (say \((o,d,k) \in \Theta^k\)) is satisfied using multiple
routes (say \(\{C_1, \cdots, C_r\}\)) then the contribution of demand \((o,d,k)\) towards the utility of a ship for
carrier \(k\) on a service route (say \(C_r\)) in the later method is taken to be a fraction (proportional to
the fraction of the length of the path used to satisfy demand \((o,d,k)\) on the service route \(C_r\)) of
the revenue generated by the demand.

We solve the ship assignment problem exactly and heuristically. In Section 6.2 we report the effect
of different ship assignment algorithms and different utility functions on the overall mechanism.

Once an assignment of ships to service routes is computed, the cost incurred by carrier \(k\) in the
collaborative solution is computed as

\[
\text{Cost of operating routes} = \sum_{C_j \in \mathcal{C}} \frac{\text{Cost}_{C_j}}{L_{C_j}} y^k_{C_j}.
\]

Note that cost of operating a ship on cycle \(C_j\) (or the cost of cycle \(C_j\) per week) is given by \(\text{Cost}_{C_j}\). Thus if carrier \(k\) assigns \(y^k_{C_j}\) ships to cycle \(C_j\) then the cost per week incurred by him on this cycle is \(\frac{\text{Cost}_{C_j}}{L_{C_j}} y^k_{C_j}\). Finally, the valuation of solution \(\text{opt}(\mathcal{N})\) for carrier \(k\) is given by:

\[
v_k(\text{opt}(\mathcal{N})) = \sum_{(o,d,i) \in \Theta \cap \Theta^k} R^{(o,d,i)} \bar{f} - \text{Cost of operating routes}
\]

where \(\bar{f}\) denotes the optimal flow vector for \(CP\).

5.3. Computation of Side Payments

In an alliance, carriers work in collaboration with each other however, the primary objective of an
individual carrier remains to be the maximization of his own profits. We model the selfish behavior
of carriers by assuming that given the collaborative network carriers solve their cargo accept-reject
and routing problems individually. Given an assignment of ships, it is in the best interest of the
collaboration that the carriers make their cargo routing decisions as in \(\bar{f}\). Note that \(\bar{f}\) requires
carriers to share capacity on the ships. We facilitate this by allowing a carrier to charge other
carriers for using his ships’ capacity on an edge \(e\) whenever he has a ship assigned to that edge.
The rest of the times he will need to pay other carriers for using capacity on edge \(e\). We refer to this
payment as the capacity exchange cost on edge \(e\) and denote it by \(\text{cost}_e\). Note that the capacity
exchange costs provide side payments to the carriers, in addition to the valuation (13) obtained by
them.

As carriers pool their ships in an alliance to operate on service routes, usually multiple carriers
have their ships assigned to any given edge. In other words, carriers usually own a fraction of the
capacity on an edge. Given an assignment of ships of various carriers to the routes in \(\mathcal{C}\), let \(\gamma^k_e\) be
the fraction of capacity that carrier \(k\) owns on edge \(e\). Recall that we assume all ships are identical
with \(T\) units of capacity. Thus, in the special case, when an edge \(e\) is part of a single service route
\(C_j\) and the number of ships assigned by carrier \(k\) to \(C_j\) are \(y^k_{C_j}\),

\[
\gamma^k_e = \frac{y^k_{C_j}}{L_{C_j}}.
\]

However, an edge can be part of many operated service routes. Next, we compute \(\gamma^k_e\) for this case.
Consider an edge \(e\) that is part of multiple service routes, \(\hat{C} = \{C_1, C_2, \cdots, C_r\} \subset \mathcal{C}\). The total
capacity that carrier \( k \) owns on edge \( e \) due to his ships on service routes in \( \tilde{C} \) is \( \sum_{C_j \in \tilde{C}} \gamma_{C_j} \). As \( e \) is part of \( r \) cycles and thus total capacity on \( e \) is \( Tr \), \( \gamma_{e}^{k} \) is given by

\[
\gamma_{e}^{k} = \frac{1}{r} \sum_{C_j \in \tilde{C}} \frac{\gamma_{C_j}^k}{L_{C_j}}.
\]

Clearly for an edge \( e \), \( \sum_{k \in \mathcal{N}} \gamma_{e}^{k} = \frac{1}{r} \sum_{k \in \mathcal{N}} \sum_{C_j \in \tilde{C}} \gamma_{C_j}^k = \frac{1}{r} \sum_{C_j \in \tilde{C}} \sum_{k \in \mathcal{N}} \gamma_{C_j}^k = \frac{1}{r} \sum_{C_j \in \tilde{C}} 1 = 1. \)

Once we calculate the fraction of capacity each carrier owns on an edge, to determine side payments among the carriers we need to determine suitable capacity exchange costs on the network edges. To compute these costs we model the selfish cargo accept-reject and routing behavior of a carrier in an alliance. However, modelling this is not easy due to the fact that while computing his own profit in the alliance an individual carrier needs to account for other carriers’ flows in the network and the portion of their capacity that he can utilize for routing his own flow. We model the individual behavior of a carrier as a linear program and assume that if a carrier \( k \) owns \( \gamma_{e}^{k} \) fraction of capacity on edge \( e \) then he is allowed to collect \( \gamma_{e}^{k} \) fraction of the cost payed by other carriers for using capacity on edge \( e \). We assume that to make his routing and capacity allocation decisions an individual carrier solves a multicommodity flow problem where he routes all the carriers’ flow in the network to maximize his profit. Note that the capacity on the edges in the multicommodity flow network is given by the collaborative optimal solution \( \text{opt}(\mathcal{N}) \), since carriers operate same service routes as in \( \text{opt}(\mathcal{N}) \). On an edge \( e \), we denote this capacity by \( \overline{\text{Cap}}_e \). Clearly in practice, an individual carrier can only make decisions regarding his own flow and capacity. In this sense our approach is a conservative one, since the maximum revenue that carrier \( k \) can obtain will always be less than the optimal value of this model. In Agarwal and Ergun (2007a), it is proven that this behavioral model leads to final pay-offs (or allocations) to carriers with properties of the final allocations obtained under each model.

Let \( f^{k} = \{ f_{e}^{(o,d,i),k}, f^{(o,d,i),k} \geq 0 \forall e \in E, \forall (o,d,i) \in \Theta \} \), where \( f^{(o,d,i),k} \) represents the optimal flow of demand \( (o,d,i) \) on edge \( e \) when carrier \( k \) makes his cargo accept-reject and routing decisions. In mathematical terms we represent the optimization problem solved by carrier \( k \) in the alliance as:

\[
\begin{align*}
\text{SCP}^{k} & \quad \text{max} \sum_{(o,d,i) \in \Theta_{k}} f_{e}^{(o,d,i),k} R^{(o,d,i)} + \sum_{e \in E_{v}} \left( \sum_{(o,d,i) \in \Theta_{k}} \gamma_{e}^{k} f_{e}^{(o,d,i),k} - \sum_{(o,d,i) \in \Theta_{k}} (1 - \gamma_{e}^{k}) f_{e}^{(o,d,i),k} \right) \text{cost}_{e} \quad (14) \\
\text{such that} & \quad \sum_{e \in \text{InEdges}(v)} f_{e}^{(o,d,i),k} - \sum_{e \in \text{OutEdges}(v)} f_{e}^{(o,d,i),k} \leq 0 \quad \forall v \in V, \forall (o,d,i) \in \Theta \quad (15) \\
& \quad \sum_{(o,d,i) \in \Theta} f_{e}^{(o,d,i),k} \leq \overline{\text{Cap}}_{e} \quad \forall e \in E \quad (16) \\
& \quad f_{e}^{(o,d,i),k} \leq D^{(o,d,i)} \quad \forall (o,d,i) \in \Theta \quad (17) \\
& \quad f_{e}^{(o,d,i),k} \geq 0 \quad \forall e \in E, \forall (o,d,i) \in \Theta. \quad (18)
\end{align*}
\]

The objective function (14) consists of three terms. The first term denotes the revenue generated by satisfying demand corresponding to carrier \( k \). The second term computes the cost paid to carrier \( k \) by the other carriers for using capacity owned by him. Similarly, the third term represents the cost paid by carrier \( k \) to the other carriers in the alliance for using their capacity. Constraints (15)-(18) are the network flow constraints. Constraints (15) are the flow balance constraints at every
vertex of the network. Constraints (16) and constraints (17) are the capacity constraints on the edges of the network.

For the single carrier problem $SCP^k$ we wish to identify a cost vector, cost, such that the collaborative optimal flow, $\bar{f}$, is an optimal solution for $SCP^k$. This problem fits well in the inverse optimization framework where given a feasible solution (flow vector $f$) to a linear program ($SCP^k$), we wish to identify the parameters of the problem (vector cost) that will make the given solution (flow vector $\bar{f}$) optimal for the problem ($SCP^k$). Next, we demonstrate the use of inverse optimization to compute the vector cost.

Let us denote by $\pi^k = (\pi^k (o,d,i), k) : \pi^k (o,d,i) > 0 \forall (o,d,i) \in \Theta$, $\lambda^k = \{\lambda^k : \lambda^k \geq 0 \forall e \in E\}$ and $\omega^k = \{\omega^k (o,d,i), k : \omega^k (o,d,i) \geq 0 \forall (o,d,i) \in \Theta\}$ the dual variables associated with constraints (15), (16) and (17) respectively. Note the use of super-script $k$ to denote that the dual is considered for the single carrier problem corresponding to carrier $k$. For carrier $k$, the dual of the $SCP^k$, denoted by $DSCP^k$, can be written as:

$$DSCP^k \min \sum_{e \in E} C_{ap_e} \lambda^k_e + \sum_{(o,d,i) \in \Theta} \omega^k (o,d,i)$$

such that

$$\pi^k_{head(e)} - \pi^k_{tail(e)} + \lambda^k_e \geq (\gamma^k_e - 1)cost_e \forall e \in E, \forall (o,d,i) \in \Theta \setminus \Theta_k \tag{20}$$
$$\pi^k_{head(e)} - \pi^k_{tail(e)} + \lambda^k_e \geq \gamma^k_e cost_e \forall e \in E, \forall (o,d,i) \in \Theta_k \tag{21}$$
$$\pi^k_{o,d,i,k} - \pi^k_d (o,d,i), k + \omega^k_{o,d,i}, k \geq 0 \forall (o,d,i) \in \Theta \setminus \Theta_k \tag{22}$$
$$\pi^k_{o,d,i,k} - \pi^k_d (o,d,i), k + \omega^k_{o,d,i}, k \geq R^k (o,d,i) \forall (o,d,i) \in \Theta_k \tag{23}$$
$$\pi^k, \lambda^k, \omega^k \geq 0. \tag{24}$$

One form of the linear programming optimality conditions states that the primal solution $f^k$ and dual solution $(\pi^k, \lambda^k, \omega^k)$ are optimal for their respective problems if $f^k$ is feasible for the constraints (15) - (18) and $(\pi^k, \lambda^k, \omega^k)$ is feasible for the constraints (20) - (24), and together they satisfy the following complementary slackness conditions:

- For all edges $e$ with flows below their capacities, i.e. $\sum_{(o,d,i) \in \Theta} f^k_{e (o,d,i)} < C_{ap_e}, \lambda^k = 0$ for all carriers $k$.
- For all demand triplets $(o,d,i)$ that are not fully satisfied, i.e. $f^k_{e (o,d,i)} < D (o,d,i), \omega^k_{o,d,i}, k = 0$ for all carriers $k$.
- For all edges $e$ with non zero flow, i.e. $f^k_e > 0$, the corresponding dual constraints (20) - (23) must be tight.

We say that a cost vector cost is inverse feasible with respect to $f^k$ if $f^k$ is an optimal solution to $SCP^k$ with cost vector cost. Let us denote by INVP the inverse problem that finds this cost vector.

Our aim is to determine the values for the vector cost such that the flow $\bar{f}$ as given by opt($\mathcal{N}$) is optimal for all individual carrier problems i.e. $SCP^k \forall k \in \mathcal{N}$. From above, $\bar{f}$ is an optimal solution for every $SCP^k$ if for every $DSCP^k$ there exists a dual feasible solution $(\pi^k, \lambda^k, \omega^k)$ and a common vector cost that satisfies the primal-dual complementary slackness conditions. This gives us the following characterization of the inverse optimization problem we should solve to determine the vector cost:

$$INVP: \bigcup_{k \in \mathcal{N}} \text{INVP}^k.$$ 

If cost is a feasible solution to $INVP$, then the overall payoff to carrier $k$ is given by:

$$x^k = v^k (\text{opt(\mathcal{N})}) + s^k$$
where the vector of side payments \( \{s_1, s_2, \ldots, s_n\} \) is calculated as,

\[
s^k = \sum_{e \in E} \left( \sum_{(o,d,i) \in \Theta_k} \gamma_{e_{(o,d,i)}} \frac{f_{e_{(o,d,i)}}}{\text{cost}_e} - \sum_{(o,d,i) \in \Theta_k} (1 - \gamma_{e_{(o,d,i)}}) \frac{f_{e_{(o,d,i)}}}{\text{cost}_e} \right).
\]

(27)

Note that the vector of payoffs to carriers, \( \{x^1, x^2, \ldots, x^n\} \), is such that \( \sum_{k \in N} x^k = \text{opt}(N) \). This is easy to see since once a feasible solution is found for \( \text{INVP} \), the flow generated by each carriers' individual decisions in the network will result in the same revenue as in \( \text{opt}(N) \) and that \( \sum_{k \in N} s^k = 0 \).

The following theorem guarantees that an inverse feasible vector \( \text{cost} \) for \( \text{INVP} \) can always be found.

**Theorem 1.** The inverse problem \( \text{INVP} \) is feasible.

Individual carrier problem \( \text{SCP}^k \) for carrier \( k \) is identical to the problem each player is assumed to solve in the simple multicommodity flow game considered in Agarwal and Ergun (2007a). Hence the result follows from Theorem 1 from Agarwal and Ergun (2007a).

The existence of a feasible vector \( \text{cost} \) guarantees that each individual carrier will make decisions in line with the collaborative optimal solution. However, it is possible that the carriers will collude to form sub-coalitions and these sub-coalitions may find it more profitable to make different routing decisions. To prohibit such an outcome, we want to find a cost vector such that for any subset \( S \subset N \), \( \bar{f} \) is an optimal solution for the corresponding problem \( \text{SCP}^S \). However, there are exponential number of such subsets and including an inverse problem corresponding to each of them in \( \text{INVP} \) will cause \( \text{INVP} \) to become exponential in size. The following theorem guarantees that it is sufficient to consider only single carrier problems in \( \text{INVP} \).

**Theorem 2.** The inverse problem \( \text{INVP} \) identifies a cost vector such that \( \bar{f} \) is optimal for any \( \text{SCP}^S \) such that \( S \subset N \).

As before, individual carrier problem \( \text{SCP}^k \) for carrier \( k \) is identical to the problem each player is assumed to solve in the simple multicommodity flow game considered in Agarwal and Ergun (2007a). Hence, Theorem 2 from Agarwal and Ergun (2007a) guarantees that it is sufficient to consider only single carrier problems in \( \text{INVP} \) to determine a cost vector that makes \( \bar{f} \) optimal for any subset \( S \subset N \) of carriers.

Finally, it is reasonable to assume that an individual carrier would seek a higher payoff in the alliance as compared to the revenue that he can generate on his own. Furthermore, as stated before the carriers can collude and form sub-coalitions if they can obtain a higher payoff. The mechanism described above guarantees that this will not happen, that is the final allocations made to the carriers are in the core, for the simple multicommodity flow game considered in Agarwal and Ergun (2007a). However, this cannot be guaranteed in the much more complicated setting of liner shipping alliances. To this end, we can enhance our model \( \text{INVP} \) by adding a set of limited rationality constraints to guarantee that no sub-coalition of a given size can be profitable when capacity exchange costs as determined by \( \text{INVP} \) are used. In our computational study we add the following single and two carrier rationality constraints to \( \text{INVP} \):

\[
\text{opt}([k]) \geq x^k \text{ for each } k \in N
\]

(28)

\[
\text{opt}([k,i]) \geq x^k + x^i \text{ for } k, i \in N
\]

(29)

where, \( \text{opt}(S) \) for \( S \subset N \) is the maximum revenue that the carriers in set \( S \) can obtain when working on their own.

More over, the core of a sea cargo network design game can in fact be empty. Consider a network with three ports (at the vertices of an equilateral triangle) as represented in Figure 3 and three
carriers A, B and C with fleet size and demand data as represented in Table 1. Let all ships be identical with capacity 100 TEU and the operational costs be negligible. Furthermore, assume that a ship takes one week to go from one port to another. The maximum revenue generated by the set \{A, B, C\} is obtained by operating the service route \(P_1 - P_2 - P_3 - P_1\) and satisfying 100 TEUs of demand from \(P_2\) to \(P_3\) and 100 TEUs of demand from \(P_1\) to \(P_2\) with \(\text{opt}(\{A, B, C\}) = 200\). Note that the same revenue is obtained by operating on service routes \(P_2 - P_3 - P_2\) and \(P_1 - P_2 - P_1\).

Similarly, it is easy to see that:

\[
\begin{align*}
\text{opt}(\{A\}) &= 100, \\
\text{opt}(\{B\}) &= \text{opt}(\{C\}) = 0, \\
\text{opt}(\{A, B\}) &= \text{opt}(\{A, C\}) = 200, \\
\text{opt}(\{B, C\}) &= 100
\end{align*}
\]

However the core of this game is empty as the budget balance condition requires

\[
x_A + x_B + x_C = 200
\]

whereas the sum of two carrier rationality constraints requires

\[
x_A + x_B + x_C \geq 250.
\]

### 6. Computational Study

Next, we analyze liner shipping alliances from a quantitative as well as a qualitative point of view. The focus of our computations is to study the performance of the mechanism designed in this paper in the context of liner shipping. Note that the core of a collaborative game concerns only with an overall payoff to the players. However, the mechanism designed in this paper not only provides an overall payoff to the members of a partially decentralized alliance but also sets the rules and provides the incentives for regulating their interactions and guiding their autonomous behaviors. The aim of our computational experiments is to study the solution quality generated by our mechanism in the light of the standard game theoretic concepts.
6.1. Data Generation and Computational Set-up

We performed our experiments on the data simulating real life data from the liner shipping industry. We assume that all ships are identical with 4000 TEU of capacity. Sailing distances between ports are chosen randomly from the intervals [2, 30] or [14, 42] days to simulate intra-region and inter-region distances between ports in Asia and North America. Origin-destination pairs are chosen randomly from the pairs of ports. Demands are randomly generated to be [0.1, 1.0] fraction of the capacity of the ships. Finally, revenue generated by satisfying a unit demand is chosen to be in direct proportion to the distance between the origin port and the destination port of the demand. Different classes of random instances are generated to test the robustness of the mechanism. Classes are characterized by specifying the number of ports \((P)\), the number of ships \((S)\) and the number of demand triplets \((D)\) for the collaborative problem. For example, an instance with 6 ports, 30 ships and 18 demand triplets is represented as \(P6S30D18\). We consider networks with up to 10 ports and fleet sizes with up to 50 ships. This data generation scheme is the same as the one used by Agarwal and Ergun (2007b) and we refer to Agarwal and Ergun (2007b) for further details and justification of various parameters regarding data generation.

As reflected in Figure 2, most global alliances among liner shipping carriers have 2 to 4 members. Hence in our computational study, we consider alliances with two, three and four carriers. We assign each demand and ship to one of the carriers with equal probability, unless specified otherwise.

All of our algorithms are implemented in C++ in an Unix environment. We also made extensive use of the callable libraries in CPLEX 9.0. All computational experiments were performed on a Sun280R system with UltraSparc-III processor. Results are reported on 50 randomly generated instances in each test class.

6.2. Effect of Asset Assignment and Rationality Constraints

Recall that the problem of assigning different carriers’ ships on the service routes is formulated as the generalized assignment problem \(SAP\) in Section 5.2. As noted earlier, this problem is NP-hard. However, in our case since the number of selected service routes are between 3 and 10 (depending on the problem size), an explicit enumeration scheme can also be used to determine the exact assignment of ships to the service routes. Next, we analyze how the mechanism is effected by different assignment of ships to the service routes. We obtain different assignments by considering different algorithms to solve \(SAP\). In particular, we consider an exact assignment, a greedy assignment and a random assignment of the ships. We also consider two different utility functions where the utility of assigning a ship to a service route for a carrier is taken to be proportional to the sum of his flow on the edges of the service route (represented by columns \(f\) in the following tables) and the utility of assigning a ship to a service route for a carrier is taken to be proportional to the sum of his profit generated from his flow on the edges of the service route (represented by columns \(f.R\) in the following tables).

We also study the effect of enhancing the inverse problem \(INVP\) by adding rationality constraints. As mentioned at the end of Section 5.3, rationality constraint for a subset \(S\) of carriers states that carriers in \(S\) seek higher payoff in an alliance as compared to the payoff they can generate on their own. We divide the rationality constraints into different sets, depending on the number of carriers considered. For example, two carrier rationality constraints is the set of rationality constraints for all subsets with two carriers and is represented by columns \(\{2\}\) in the following tables. To study the effect of rationality constraints we introduce one set of rationality constraints at a time to the inverse problem \(INVP\). Inverse program together with all the single carrier rationality constraints is referred as \(INVP + \{1\}\) and inverse program together with all the single carrier and two carrier rationality constraints is referred as \(INVP + \{1\} + \{2\}\) in the following discussion. Recall that \(INVP\) is a feasibility problem and the final payoff made by our algorithms is always
budget balanced. Thus for a three carrier alliance if \( INVP + \{1\} + \{2\} \) is feasible than it means that a capacity exchange cost structure that yields a payoff allocation in the core can be identified.

Table 2 reports the effect of different assignment of ships on the service routes and the effect of rationality constraints on the solution quality for 3 carrier alliances and different test classes. In this table, columns \( \{0\} \) present the results when \( INVP \) is solved, columns \( \{1\} \) present the results when \( INVP + \{1\} \) is solved and columns \( \{1\} + \{2\} \) presents the results when \( INVP + \{1\} + \{2\} \) is solved. The first column identifies the different problem classes solved based on the total number of ports, ships and demand pairs considered. For these experiments, each demand and ship is assigned to one of the three carriers with equal probability. The next three columns report the number of instances (out of 50) for which an allocation in the core is found by our mechanism when the \( SAP \) is solved exactly and the utility function is taken to be proportional to the profit generated by the flow. The next three triplets of columns report the same numbers when \( SAP \) is solved exactly, greedily and randomly, respectively. In these cases the utility function is taken to be proportional to the flow. The two triplet of columns for the exact assignment of ships to the service routes suggest that the choice of utility function does not impact the overall payoffs significantly. Thus for the next two triplet of columns we performed our experiments with only one type of utility function.

Different assignment algorithms and utility functions result in different number of ships being assigned by a carrier to each of the selected service routes. This in turn influences a carrier’s valuation of the optimal solution and the way the optimal solution is realized by the alliance. Note that the inverse problem computes the cost structure for a given assignment of ships to the service routes. Table 2 suggests that the number of cases for which the mechanism successfully finds a cost structure which result in a final core allocation does not depend significantly on the way the ships are assigned to the service routes. If we consider all the rationality constraints, even for a random assignment of ships to the service routes, in most of the instances the mechanism is able to determine a cost structure that yields an allocation in the core.

From Theorem 1, the inverse program \( INVP \) is feasible. We found in our computational study that \( INVP + \{1\} \) is also feasible in all the instances. However, in some cases a feasible solution for \( INVP + \{1\} + \{2\} \) could not be found. Note that for three carrier alliances a solution to \( INVP + \{1\} + \{2\} \) means that a cost structure that yields payoffs in the core can be identified. For \( INVP \) and \( INVP + \{1\} \) we report if the feasible solution provided by CPLEX is in the core. For these cases there might be alternate solutions and only some of these might provide an allocation in the core.

Table (2) suggests that a feasible solution to \( INVP \) in 10-25% of the instances directly yields an allocation in the core. As the inverse problem is constrained by adding single carrier rationality constraints in 25-45% of the instances the inverse feasible solution obtained by CPLEX yields an allocation in the core. Furthermore \( INVP + \{1\} + \{2\} \) is feasible in 70-95% of the cases, depending on the test class. Thus in 70-95% of the cases our mechanism leads to a profit allocation in the core.

### 6.3. Analysis of Different Test Classes

We analyze the effect of different test classes in Table 3. We consider alliances with three carriers. Ships and demand pairs are distributed uniformly among the carriers. We solve the ship assignment problem exactly and the utility of assigning a ship to a service route for a carrier is taken to be proportional to the sum of his flow times the revenue on the edges of the service route. First column in Table 3 identifies the different problem classes. For each test class, the second column reports the average CPU time taken (averaged over 50 instances) in minutes to solve a problem instance. This includes the time taken to solve the service network design problem for all the subsets of carriers and the time taken to solve the inverse problem. The third column represents the number of
<table>
<thead>
<tr>
<th>Test</th>
<th>Class</th>
<th>Greedy</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Greedy</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Random</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2: Effect of asset assignment and rationality constraints.
instances, out of a total of 50 random instances generated for each test class, for which an allocation in the core exists. To test if the core of a problem is non-empty, a linear program consisting of all the core inequalities is constructed and its feasibility is tested. The next three columns report the average percentage of unsatisfied demand, the average number of un-utilized ships and the average utilization of capacity on the edges of the network respectively, for the instances with a non-empty core. The next three columns report same statistics for the cases with empty core.

The second column in Table 3 suggests that as the problem size (number of ports, ships or demand pairs) increases the time taken to solve the problem increases. Also, more than 95% of the time reported here is taken in solving the network design problem for various subset of carriers. The increase in time taken to solve the network design problem with the increase in problem size is similar to the trend reported in Agarwal and Ergun (2007b).

Note that among the test classes, P6S30D6 and P10S50D10 have the highest number of instances with an empty core. A closer look at Table 3 reveals that these test classes have the highest number of un-used ships and the lowest percentage of unsatisfied demand. Also instances in these classes have lower utilization of capacity on the edges of the network. More specifically, these networks have over capacity. This leads us to the conclusion that instances with over-capacity are more likely to have an empty core. The primary motivation for carriers to collaborate in liner shipping is that they do not have enough ships to maintain weekly frequency on the routes. For instances other than in P6S30D6 and P10S50D10 test classes, since carriers and subset of carriers have few ships (as compared to the available demand), in most of the cases the grand alliance offers the best possibility for maintaining the required frequency on the best service routes and thus most of the instances have a non-empty core. Instances in P6S30D6 and P10S50D10 test classes are however more likely to have many profitable sub-coalitions. Our experiments yield that subsets of carriers that have good synergy in the origin-destination port of their demand triplets are more likely to form sub-coalitions.

For 6 port instances with 18 ships, the percentage of unsatisfied demand is quite high. Further the average unsatisfied demand for the instances with empty core is even higher than the average unsatisfied demand for the instances with non-empty core. Thus instances with small fleet sizes in which carriers find synergies among themselves to satisfy higher demand are more likely to have a stable grand alliance. Also we note from Table 3 that as the size of the network increases from 6 to 10 ports all the instances in all the test classes (except P10S50D10) have non-empty cores. Instances with 10 ports that have a very high demand as compared to the available fleet (un-satisfied demand is 40-60% of the total demand) are very likely to form stable grand alliance. In these instances, as there is a shortage of ships and only the grand alliance provides a global optimal schedule for the available fleet. Table 3 reflects that in fact the grand alliance schedules

### Table 3: Analysis of test classes.

<table>
<thead>
<tr>
<th>Test Class</th>
<th>Time</th>
<th># Non-empty core</th>
<th>Non-empty core</th>
<th>Empty core</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>%Unmet demand</td>
<td>Unused ships</td>
</tr>
<tr>
<td>P6S18D6</td>
<td>1.92</td>
<td>48</td>
<td>21.9 0.5 0.70</td>
<td>41.6 1 0.59</td>
</tr>
<tr>
<td>P6S18D9</td>
<td>3.08</td>
<td>49</td>
<td>36.93 0.25 0.64</td>
<td>45.07 0 0.96</td>
</tr>
<tr>
<td>P6S30D6</td>
<td>5.75</td>
<td>36</td>
<td>3.61 2.22 0.77</td>
<td>0.25 3.29 0.56</td>
</tr>
<tr>
<td>P6S30D9</td>
<td>8.60</td>
<td>46</td>
<td>11.60 0.76 0.83</td>
<td>5.30 0.75 0.72</td>
</tr>
<tr>
<td>P10S30D18</td>
<td>98.01</td>
<td>50</td>
<td>43.83 0.04 0.83</td>
<td>N/A N/A N/A</td>
</tr>
<tr>
<td>P10S30D27</td>
<td>181.46</td>
<td>50</td>
<td>58.91 0 0.86</td>
<td>N/A N/A N/A</td>
</tr>
<tr>
<td>P10S50D10</td>
<td>48.70</td>
<td>35</td>
<td>0.52 4.39 0.60</td>
<td>0.63 6.43 0.65</td>
</tr>
<tr>
<td>P10S50D18</td>
<td>306.12</td>
<td>50</td>
<td>16.90 0.45 0.88</td>
<td>N/A N/A N/A</td>
</tr>
<tr>
<td>P10S50D27</td>
<td>514.61</td>
<td>50</td>
<td>28.19 0 0.91</td>
<td>N/A N/A N/A</td>
</tr>
</tbody>
</table>
almost all the ships in the fleet and provides very high utilization of capacity (85-95%) on the operated routes.

Note that from Table 2 and Table 3, if the core of a problem instance is non-empty, our mechanism succeeds in finding capacity exchange costs that lead to allocations in the core in almost all (95-100%) of the instances, when the inverse problem is considered with all the subset rationality constraints.

6.4. Size and Number of Carriers

Next, we study the effect of the number of carriers in an alliance. Table 4 reports results for alliances with two, three and four carriers. The first column represents the test class. To generate instances with $i$-carriers the number of demand pairs and ships are distributed uniformly randomly among $i$ carriers. Thus, Table 4 reports results for alliances among carriers with similar characteristics. The second and the third columns report results for alliances with two carriers. The second column reports the number of instances (out of 50) for which an allocation in the core exists. The third column reports the average (average taken over 50 instances) percentage improvement in the total revenue generated by the alliance as compared to the sum of the revenue generated by individual carriers working independently. The next two sets of columns report similar statistics for alliances with three and four carriers, respectively. For a particular instance, the percentage improvement in the revenue generated as a result of the alliance is computed by:

\[
\frac{\text{opt}(N) - \sum_{i \in N} \text{opt}(\{i\})}{\sum_{i \in N} \text{opt}(\{i\})}.
\]

Table 4 suggests that as the number of carriers increases in an alliance, the alliance tends to become more un-stable in the sense that the number of instances with a non-empty core decreases. Note that as the number of carriers increases the number of constraints that needs to be satisfied to obtain an allocation in the core increases exponentially and with a higher number of carriers it is more likely that some subset of carriers will have better synergies than the grand alliance. Qualitatively, as the number of carriers increases in an alliance the organizational complexity of the alliance increases and the decision making process becomes time consuming. One of the most successful alliances among liner carriers have been the Maersk-Sealand alliance which consists of only two carriers Song and Panayides (2002). Some of the bigger alliances have organized and re-organized themselves a number of times within a short period. For example, the Global Alliance which was formed in 1995 among four carriers (APL, OOCL, MOL and Nedlloyd) reorganized in 1998 to form the New World Alliance (NOL/APL, MOL, HMM) after the merger of APL and NOL in 1997.
The third, fifth and the seventh column of Table 4 clearly shows that alliances can generate higher revenues as compared to the carriers operating on their own. For a particular instance, i.e. for a given set of ports, fleet size and demand pairs, as we increase the number of carriers (that is distribute the fleet and demand pairs among a larger number of carriers), the revenue that an individual carrier can generate reduces. However, the optimal solution of the grand alliance remains the same, independent of the number of carriers in the alliance. Thus, as reflected by Table 4 the percentage increase in the revenue generated as a result of the alliance increases as the number of carriers increases.

6.5. Role and Contribution of Carriers in an Alliance

In this section, we analyze how participants with complementary or similar characteristics influence an alliance. Specifically, we study the alliance between a ship owner and a group of shippers. That is one carrier has all the ships and the other carriers have all the demand. First, we study instances (drawn from different test classes P6S18D6 - P10S50D27) with one ship owner and one shipper. This is a perfect situation for collaboration and all these instances have a non-empty core. Further, in all such instances our mechanism provides a cost structure such that the resulting payoffs to both participants are in the core when the inverse problem is solved with all the subset rationality constraints. Thus a stable alliance can be formed in all these instances. Next, we study problem instances with three shippers and a single ship owner. Depending on the problem instance we found that the core is non-empty in 90%-100% of the cases. Among the instances with non-empty core, in 95%-100% of the instances our mechanism provides a cost structure such that the resulting payoffs to the participants are in the core. Comparing one ship owner and one shipper case with the one ship owner and three shippers case we conclude that in the latter case, shippers give rise to competition and instability in the grand alliance.

An interesting observation is that for $P6S30D6$ problem instances with one ship owner and three shippers 98% of the instances have non-empty core. Whereas, for this test class when the ships and demand pairs are distributed uniformly among four carriers only 58% of the instances have non-empty core. For the $P6S30D6$ class the number of ships are enough to satisfy most of the demand, thus even if there are three competing shippers the alliance is stable. As the ship owner has sufficient number of ships, he has an incentive to collaborate with as many shippers as possible to increase his revenue. Similarly, though the shippers compete for capacity on the ships, in the case when the system has over-capacity they can all form a sustainable alliance with the ship owner. However this is not the case when ships and demand pairs are uniformly distributed among four carriers as many subset of carriers find synergies to form sub-coalitions. In general, for other test classes also, instances with ships and demand pairs distributed among one ship owner and three shippers are more likely to have a non-empty core as compared to four equi-sized carriers. This is simply because in the former scenario the carriers have a higher degree of complementarity in their roles. Carriers have used conferences and alliances to fix price and moderate the buying power of shippers. As a result of our experiments we conclude that it is a good strategy for shippers also to form alliances and consolidate cargo before negotiating with the carriers and ship operators. This practice is observed in the industry as giant shippers and freight-forwarders consolidate the cargo of small shippers.

7. Conclusions

In this paper we addressed various problems posed by alliance formation among carriers in the transportation industry. We designed allocation mechanism for a partially decentralized alliance to share the benefits of the collaboration in such a way that all the members are motivated to act in the best interest of the alliance. Considering the preliminary results obtained by us, we believe that the suggested solution approach has the potential to help carriers form sustainable alliances.
Our experiments suggest that across all test classes, in most (more than 95%) of the instances that have non-empty core our mechanism successfully finds a cost structure such that the resulting payoff to the carriers is in the core when the inverse problem is solved with all the rationality constraints. Assignment of ships to the service routes influences the cost incurred by a carrier (thus his payoff) and the ownership of capacity on the edges by the carriers. However our results indicate that independent of the assignment of ships to the service routes in most of the cases our mechanism successfully finds a cost structure such that the overall payoff to the carriers is in the core. Analysis of different test classes suggests that the core is empty for a very high number of instances (more than 72%) drawn from the classes in which carriers have sufficient number of ships to satisfy the available demand. Further our experiments yield that, as the number of carriers increase in an alliance, the percentage improvement in the total revenue generated by the alliance as compared to the sum of the revenue generated by individual carrier independently increases. However, it becomes harder to find a solution in the core as the number of constraints to obtain a core allocation increases exponentially with the number of carriers. Further we conclude that carriers who have complementarity in their roles, for example ship owners and freight forwarders, are more likely to form stable alliances. Note that many other factors (such as compatibility in mission, strategy, governance, culture, organization and management etc. of different partners of an alliance) also contribute significantly to the success of an alliance.

In this paper we considered alliance formation among 3-4 carriers. For these alliances we considered all subset rationality constraints to find an allocation in the core. In many other transportation and other logistics settings it is necessary to consider alliances with a higher number of participants. Also in liner shipping, smaller alliances are collaborating to from even bigger alliances, for example Grand Alliance and The New World Alliance laid down foundations for cooperation in 2006. Considering all rationality constraints becomes prohibitively expensive as the number of carriers increases in an alliance. To extend the mechanism developed in this paper for alliances with higher number of participants, further research is needed in order develop how to add subset rationality constraints in the inverse problem in a constraint generation setting.

The liner shipping industry is deploying bigger and bigger ships. Further research is required to quantitatively study the viability of bigger ships and their impact on alliances.

Acknowledgments
Ozlem Ergun was supported in part under NSF CAREER grant DMI-0238815 and NSF ITR grant DMI-0427446.

References


